Lab #4. Challenges

Prof. Jaeseung Choi

Dept. of Computer Science and Engineering

Sogang University

General Information

- **Check "Lab #4" in *Assignment* tab of *Cyber Campus***
  - Skeleton code (**`Lab4.tgz`**) is attached in the post
  - Deadline: **12/04 Monday 23:59**
  - Submission will be accepted in that post, too
  - **No late submission for this lab**
- **Please read the instructions in this slide carefully**
  - New constraints are added for this lab
  - It also contains important submission guidelines
    - If you do not follow the guidelines, you will get penalty

Remind: Course Policy

- **Cheating (code copy) is strictly forbidden in this course**
  - Read the orientation slide once more
- **Don’t ask for solutions in the online community**
  - TA will regularly monitor the communities
- **Sharing your code with others is as bad as copying**
  - Your cooperation is needed to manage this course successfully
- **Even after the end of the course, please do not upload your code at GitHub or share it with your friends**
  - This makes it hard to manage the course in the following years

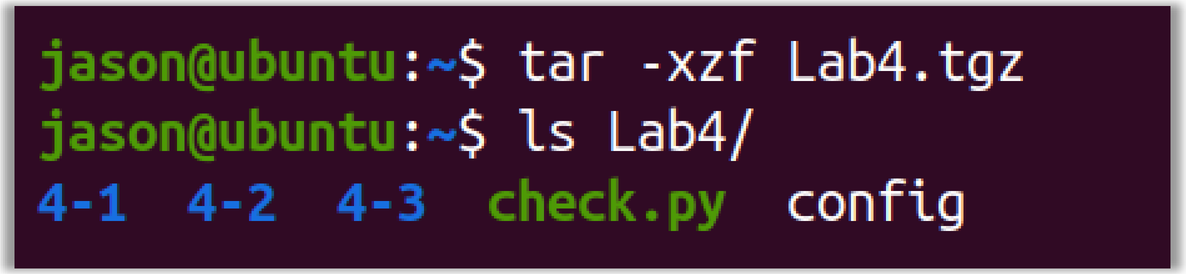
Again, any discussion about the problem is forbidden

Remind: Grading Components

- **As I announced after the midterm exam, there has been updates to the weight of grading components**
  - Note that the total score is 90% this year (not 100%)
- **Until the midterm: 45%**
  - Lab #1: Warm-up **(5%)**
  - Lab #2: BOF **(10%)**
  - Midterm exam **(30%)**
- **After the midterm: 45%**
  - Lab #3. ROP **(12%)**
  - Lab #4. Challenges **(13%)**
  - Final quiz in **12/12 Tuesday** **(20%)**

Skeleton Code Structure

- Copy `Lab4.tgz` into CSPRO server and decompress it
  - Now you can use cspro5.sogang.ac.kr or cspro.sogang.ac.kr
- Overall structure is same as before:
  - `4-1/ ... 4-3/`: Problems you have to solve
  - `check.py`: Self-grading script
  - `config`: Used internally by the self-grading script



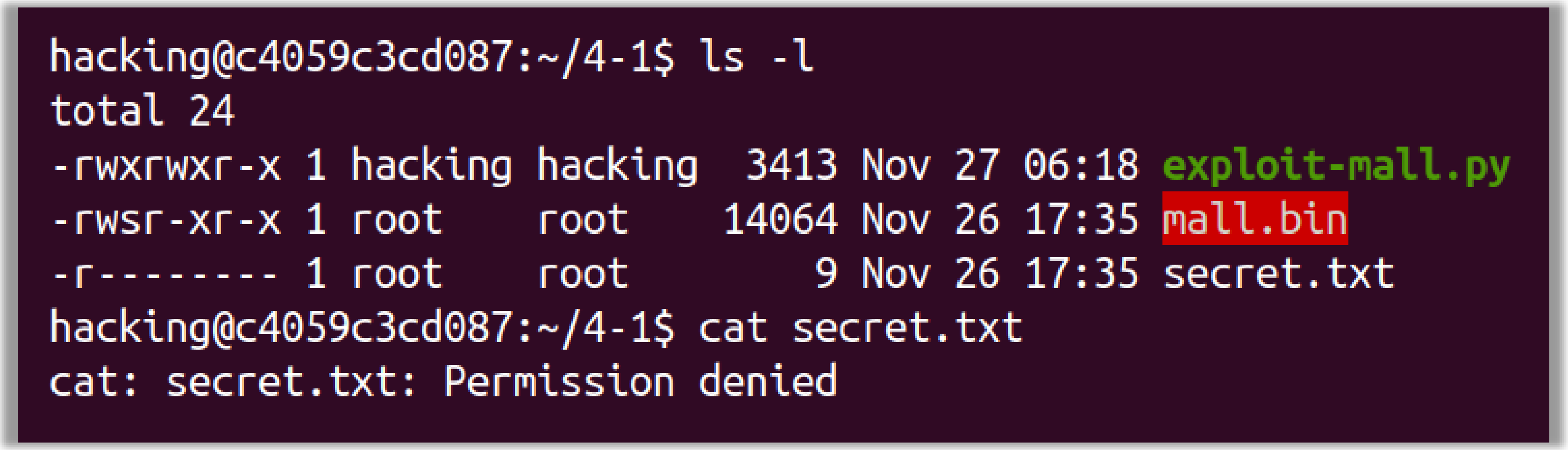
```
jason@ubuntu:~$ tar -xzf Lab4.tgz
jason@ubuntu:~$ ls Lab4/
4-1   4-2   4-3   check.py   config
```

Difficulty

- **The problems in this lab will be hard**
  - You will have to make use of all the things you've learned so far
  - Don't be frustrated even if you cannot solve any of the problem
- **Also, no question will be accepted for this lab**
  - Except for the questions about the specification or constraints
  - So it will be a real challenge that you have to solve on your own
- **Solution will be discussed in the make-up class**
  - **12/05 Tuesday 19:00 (K202)**
  - No attendance check

Grading Environment: SUID

- **During the grading, the target program will have SUID**
  - Review the ***Side Note*** of Chapter 2
- **Assume that the target program and `secret.txt` file are owned by the `root` user**
- **Meanwhile, your exploit is executed as a normal user**
  - Therefore, your exploit script cannot read **`secret.txt`** directly



```
hacking@c4059c3cd087:~/4-1$ ls -l
total 24
-rwxrwxr-x 1 hacking hacking  3413 Nov 27 06:18 exploit-mall.py
-rwsr-xr-x 1 root    root    14064 Nov 26 17:35 mall.bin
-r-------- 1 root    root        9 Nov 26 17:35 secret.txt
hacking@c4059c3cd087:~/4-1$ cat secret.txt
cat: secret.txt: Permission denied
```

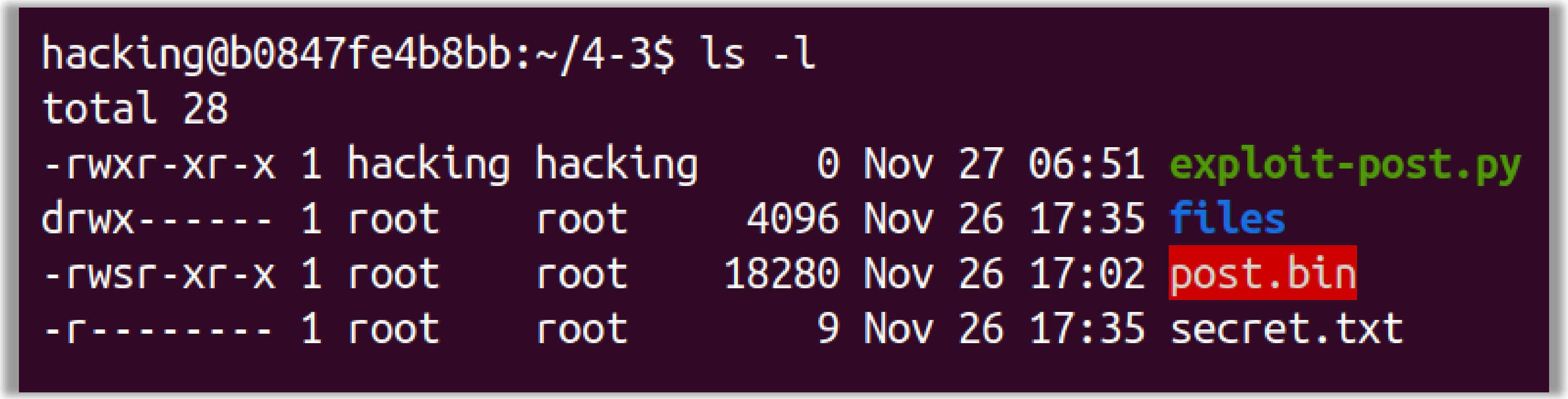
Grading Environment: Shell

- **Okay, but why should I care about the existence of SUID?**
- **It affects the behavior of several functions and programs**
  - **`system()`** drops the SUID privilege before executing command
  - **`/bin/sh`** program also drops the SUID privilege when launched
  - In other words, the shell will **not be able to read** **`secret.txt`**
- **Therefore, you must spawn a shell in the following way**
  - Use **`exec*`** functions, not **`system()`**
  - Pass **`"-p"`** option to **`/bin/sh`**: this prevents the shell from dropping the SUID privilege

```
char *argv[3];
argv[0] = "/bin/sh"; argv[1] = "-p"; argv[2] = NULL;
execv(argv[0], argv); // This is what you have to do
```

Grading Environment: Permission

- **In problem 4-3, the target program will read and write files in a directory name `files/`**
  - Again, you must assume that this directory is **not directly readable or writable** by your exploit script
  - **`root`** permission is required to read and write files here
  - Your exploit can read and write files only by ***interacting with*** the target program (since the target program has **`root`** SUID)



```
hacking@b0847fe4b8bb:~/4-3$ ls -l
total 28
-rwxr-xr-x 1 hacking hacking     0 Nov 27 06:51 exploit-post.py
drwx------ 1 root    root     4096 Nov 26 17:35 files
-rwsr-xr-x 1 root    root    18280 Nov 26 17:02 post.bin
-r-------- 1 root    root        9 Nov 26 17:35 secret.txt
```

Tips

- **Carefully read the Linux manual page for the dynamic memory allocation (type "`man malloc`" in terminal)**
- **Test various `malloc()`/`free()` sequences and examine which addresses are used**
  - Since it is hard to analyze and understand the internal algorithm of **`malloc()`** and **`free()`**, this would be a better approach
- **Be careful: `printf()` internally allocates heap memory, so it will have side effects on the result of `malloc()`**

```
void *p1 = malloc(32);
free(p1);
void *p2 = malloc(32);
printf("%p, %p\n", p1, p2); // Call printf() at the end
```

Problem Information

- **Four problems, 100pt in total**
  - **4-1 (30pt): `mall.bin`**
  - **4-2 (35pt): `login.bin`**
  - **4-3 (35pt): `post.bin`**
- **You'll get the point for each problem if the exploit works**
  - If your exploit works in the CSPRO server, but does not work in the environment with SUID, I will consider partial point
- **For each problem, your report must clearly explain the vulnerability you found and your approach for exploit**
  - You may **lose points** if the report does not clearly describe it
  - No template for the report this time

Submission Guideline

- **You should submit four exploit scripts and report**
  - Problem 4-1: **`exploit-mall.py`**
  - Problem 4-2: **`exploit-login.py`**
  - Problem 4-3: **`exploit-post.py`**
  - **Don’t forget the report**: **report.pdf**
- **Submission format**
  - Upload these files directly to *Cyber Campus* (**do not zip them**)
  - **Do not change the file name** (e.g., adding any prefix or suffix)
  - If your submission format is wrong, you will get **-20% penalty**