

System Programming Lab #7

2019-04-23

sp-tas

Lab Assignment #4 : Kernel Lab

- Download skeleton code & pdf from eTL
 - kernellab-handout.tar, kernellab-handout.pdf
- Hand In #1 – Setup
 - capture your development environment
 - Upload your capture image eTL
 - filename for part #1 : [학번]_[이름]_kernellab_ptree.jpg (or .png, etc)
 - filename for part #2: [학번]_[이름]_kernellab_paddr.jpg (or .png, etc)
- Hand In #2 – Your Implementation
 - Upload your files eTL
 - 압축파일 양식 : [학번]_[이름]_kernellab.tar (or .zip, etc)
 - Ex) 2017-12345_홍길동_kernellab.tar
 - A zip file should include
 - (1) a tarball of your implementation directory (2) report
 - tarball 양식 : kernellab-[학번].tar.gz eg) kernellab-2019-12345.tar.gz
 - Report 양식 : [학번]_[이름]_kernellab_report.pdf (or .hwp, .txt etc)
- Please, **READ** the Hand-out and Lab material thoroughly!

Lab Assignment #4 : Kernel Lab

- **Step 1.** Setup
 - (part #0) Load my own kernel module
- **Step 2.** Implementation
 - (part #1) Tracing process tree from process id
 - (part #2) Finding physical address using virtual address
- Assigned : April 23
- Deadline for **Step 1.** Setup : April 29, 23:59:59 (Delay NOT allowed)
- Deadline for **Step 2.** Implementation : May 13, 23:59:59
- Delay policy : Same as before
- Lab sessions will be
 - 4/23: Kernel lab part #0, #1
 - 4/30: Kernel lab part #2
 - 5/7 : Kernel lab Q&A session

Linux kernel programming

- Two Programmatic ways to access kernel space
 - 1. Adding system call to system(Kernel code).
 - Need to recompile whole kernel.
 - 2. Adding Loadable Kernel Module
 - Load&unload new interface to the system.

What is a kernel module?

- Module
 - Pieces of code that can be loaded & unloaded to kernel
- How to Compile
 - Kernel module is not compiled with general gcc
 - It needs kernel specific compile tools
- How to load & unload my code
 - Load `root # insmod <module_name.ko>`
 - Check `root # lsmod`
 - Unload `root # rmmod <module_name>`
- # All implemented in Makefile!

What is debug file system?

- **Debug File System(debugfs)** is Special file system available in the Linux Kernel.
- Provides simple way for kernel developers to make information available to user space.
- User space developers can access Linux Kernel information easily using debugfs.

Debug file system APIs

- Description of debugfs API

- <https://www.kernel.org/doc/Documentation/filesystems/debugfs.txt>

```
struct dentry *debugfs_create_dir(const char *name, struct dentry *parent)
struct dentry *debugfs_create_file(const char *name, umode_t mode,
                                   struct dentry *parent, void *data,
                                   const struct file_operations *fops)

struct dentry *debugfs_create_u32(const char *name, umode_t mode,
                                  struct dentry *parent, u32 *value)
struct dentry *debugfs_create_u64(const char *name, umode_t mode,
                                  struct dentry *parent, u64 *value)

struct dentry *debugfs_create_x32(const char *name, umode_t mode,
                                  struct dentry *parent, u32 *value)
struct dentry *debugfs_create_x64(const char *name, umode_t mode,
                                  struct dentry *parent, u64 *value)

struct debugfs_blob_wrapper {
    void *data,
    unsigned long size;
};
struct dentry *debugfs_create_blob(const char *name, umode_t mode,
                                   struct dentry *parent,
                                   struct debugfs_blob_wrapper *blob)
```

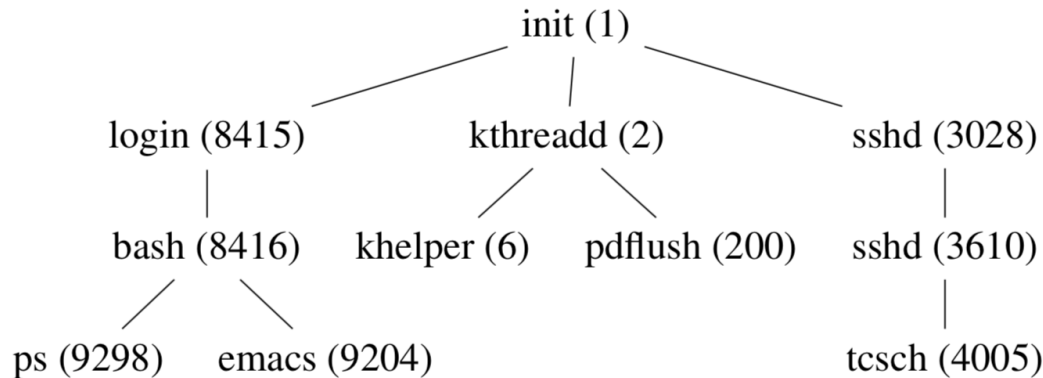
File Operations

- File Operations
 - The file function pointer structure
 - File Operations are used to communicate with files in Device Driver and Debug File System.

```
struct file_operations Fops = {  
    .read = file_read,  
    .write = file_write,  
    .open = file_open,  
    .release = file_close,  
};
```


Kernel lab part #1 – tracing parent process tree

- Trace process from leaf to init process



- Spec

- Input : [input process id]
- Output : list of [process name] [process id]

Ex) input : 9204 output : init(1)
 login (8415)
 bash (8416)
 emacs (9204)

Kernel lab part #1 – tracing parent process tree

- Testing

- Get root access `user# sudo su`
- Go to ptree dir `root# cd /sys/kernel/debug/ptree`
- Show current process `root# ps`
- Write input PID to file `root# echo [input process id] >> input`
- Read ptree file `root# cat ptree`

- Example output

```
unix> cat ptree
init (1)
xfce4-panel (2306)
xfce4-terminal (2408)
bash (2413)
sudo (2881)
```

With Skeleton Code

```
static int __init dbfs_module_init(void)
```

<- executed when module is inserted

```
{
```

```
    // Implement init module code
```

```
    dir = debugfs_create_dir("ptree", NULL);
```

```
    if (!dir) {
```

```
        printk("Cannot create ptree dir\n");
```

```
        return -1;
```

```
    }
```

```
    inputdir = debugfs_create_file("input", , , , );
```

<- file to read input

```
    ptreedir = debugfs_create_dir("ptree", , , );
```

<- file to write output

```
    printk("dbfs_ptree module initialize done\n");
```

```
    return 0;
```

```
}
```

```
static void __exit dbfs_module_exit(void)
```

<- executed when module is deleted

```
{
```

```
    // Implement exit module code
```

```
    printk("dbfs_ptree module exit\n");
```

```
}
```

```
module_init(dbfs_module_init);
```

```
module_exit(dbfs_module_exit);
```

With Skeleton Code

- ```
static ssize_t write_pid_to_input(struct file *fp,
 const char __user *user_buffer,
 size_t length,
 loff_t *position)
{
 pid_t input_pid;

 sscanf(user_buffer, "%u", &input_pid); <- read input pid
 //curr = // Find task_struct using input_pid. Hint: pid_task

 // Tracing process tree from input_pid to init(1) process

 // Make Output Format string: process_command (process_id)

 return length;
}

static const struct file_operations dbfs_fops = { Begin of code <- file write operation
 .write = write_pid_to_input,
};
```

# Hints. Helpful kernel functions & data structures

- struct dentry
- struct task\_struct
- struct list\_head
- INIT\_LIST\_HEAD()
  - list\_add()
  - list\_for\_each\_entry()

# Kernel programming 101 (utilities)

---

- dmesg
  - dmesg -w
- printk
- insmod / rmmod / lsmod

# Prepare your own development environment

- Oracle VirtualBox + Ubuntu 16.04
- we will check your preparation status
  - capture kernel module load/exit image as follow
  - deadline: 4/29 (Mon) 23:59:59
  - No delay allowed
  - upload your capture image file to eTL
    - filename for part#1 : [학번]\_[이름]\_kernellab\_ptree.jpg (or .png, etc)
    - filename for part #2: [학번]\_[이름]\_kernellab\_paddr.jpg (or .png, etc)
    - you will **lose 5 points** if you missing upload your capture image until 4/29

# Prepare your own development environment

- Part #1: ptree module compile preparation

sptest [실경 중] - Oracle VM VirtualBox

파일 마신 보기 입력 장치 도움말

root@spta-VirtualBox: /home/spta

```
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree# lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 16.04.1 LTS
Release: 16.04
Codename: xenial

root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree# uname -ar
Linux spta-VirtualBox: 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016 x86_64 x86_64 GNU/Linux

root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree# make
make -C /lib/modules/4.4.0-31-generic/build M=/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree module
es;
make[1]: Entering directory '/usr/src/linux-headers-4.4.0-31-generic'
CC [M] /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.o
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c: In function 'write_pid_to_input':
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c:16:15: warning: unused variable 'input
_pid' [-Wunused-variable]
 pid_t input_pid;
 ^
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c: At top level:
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c:8:23: warning: 'dir' defined but not u
sed [-Wunused-variable]
 static struct dentry *dir, *inputdir, *ptreedir;
 ^
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c:8:29: warning: 'inputdir' defined but
not used [-Wunused-variable]
 static struct dentry *dir, *inputdir, *ptreedir;
 ^
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c:8:40: warning: 'ptreedir' defined but
not used [-Wunused-variable]
 static struct dentry *dir, *inputdir, *ptreedir;
 ^
/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.c:9:28: warning: 'curr' defined but not
used [-Wunused-variable]
 static struct task_struct *curr;
 ^
Building modules, stage 2.
MODPOST 1 modules
CC /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.mod.o
LD [M] /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/dbfs_ptree.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.4.0-31-generic'
sudo insmod dbfs_ptree.ko
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree# make clean
make -C /lib/modules/4.4.0-31-generic/build M=/home/spta/Downloads/CSAP_KernelLab/skeleton/ptree clean
;
make[1]: Entering directory '/usr/src/linux-headers-4.4.0-31-generic'
CLEAN /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/.tmp_versions
CLEAN /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree/Module.symvers
make[1]: Leaving directory '/usr/src/linux-headers-4.4.0-31-generic'
sudo rmmod dbfs_ptree.ko
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/ptree#
```

```
[5.265557] systemd-journald[222]: Received request to flush runtime journal from PID 1
[5.441991] audit: type=1400 audit(1555842325.756:2): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/lightdm/lightdm-guest-session" pid=413 comm="apparmor_parser"
[5.441995] audit: type=1400 audit(1555842325.756:3): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/lightdm/lightdm-guest-session/chromium" pid=413 comm="apparmor_pa
rser"
[5.445794] audit: type=1400 audit(1555842325.760:4): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/sbin/dhclient" pid=415 comm="apparmor_parser"
[5.445797] audit: type=1400 audit(1555842325.760:5): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/NetworkManager/nm-dhcp-client.action" pid=415 comm="apparmor_parse
r"
[5.445799] audit: type=1400 audit(1555842325.760:6): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/NetworkManager/nm-dhcp-helper" pid=415 comm="apparmor_parser"
[5.445801] audit: type=1400 audit(1555842325.760:7): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/connman/scripts/dhclient-script" pid=415 comm="apparmor_parser"
[5.463102] audit: type=1400 audit(1555842325.776:8): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/bin/evince" pid=417 comm="apparmor_parser"
[5.463106] audit: type=1400 audit(1555842325.776:9): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/bin/evince/sanitized helper" pid=417 comm="apparmor_parser"
[5.463108] audit: type=1400 audit(1555842325.776:10): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/bin/evince-previewer" pid=417 comm="apparmor_parser"
[5.743474] vgdvrHeartbeatInit: Setting up heartbeat to trigger every 2000 milliseconds
[5.743547] input: Unspecified device as /devices/pci0000:00/0000:00:04:00/input/input7
[5.769195] vboxguest: misc device minor 55, IRQ 20, I/O port d020, MMIO at 00000000f0400000 (size
0x400000)
[5.769197] vboxguest: Successfully loaded version 5.0.18_Ubuntu (interface 0x00010004)
[5.788740] random: nonblocking pool is initialized
[5.891797] [drm] Initialized drm 1.1.0 20060810
[5.901461] pitx4 smbus 0000:00:07:0: SMBus Host Controller at 0x4100, revision 0
[5.954858] AVX2 version of gcm enc/dec engaged.
[5.954860] AES CTR mode by8 optimization enabled
[5.957749] [drm] VRAM 01000000
[5.971386] [ITM] Zone kernel: Available graphics memory: 508136 kiB
[5.971389] [ITM] Initializing pool allocator
[5.971392] [ITM] Initializing DMA pool allocator
[5.976199] fbcon: vboxdrmfb (fb0) is primary device
[6.043385] Console: switching to colour frame buffer device 100x37
[6.048086] vboxvideo 0000:00:02:0: fb0: vboxdrmfb frame buffer device
[6.048625] [drm] Initialized vboxvideo 1.0.0 20130823 for 0000:00:02:0 on minor 0
[6.049011] snd_intel8x0 0000:00:05:0: disable (unknown or VT-d) VM optimization
[6.061349] intel_rapl: no valid rapl domains found in package 0
[6.429932] snd_intel8x0 0000:00:05:0: white list rate for 1028:0177 is 48000
[6.775637] Adding 1046524k swap on /dev/sda5. Priority:-1 extents:1 across:1046524k FS
[8.085821] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[8.087517] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[8.088602] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[8.088867] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[8.758092] floppy0: no floppy controllers found
[8.758125] work still pending
[271.472653] dbfs_ptree: module verification failed: signature and/or required key missing - tainti
ng kernel
[271.472785] dbfs_ptree module initialize done
[275.029860] dbfs_ptree module exit
```



# Prepare your own development environment

- Part #2: paddr module compile preparation

sptest [실행 중] - Oracle VM VirtualBox

파일 편집 보기 입력 장치 도움말

```
Terminal File Edit View Search Terminal Help
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr # ls -ls_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 16.04.1 LTS
Release: 16.04
Codename: xenial
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr # uname -ar
Linux spta-VirtualBox 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016 x86_64 x86_64 GNU/Linux
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr # make
make -C /lib/modules/4.4.0-31-generic/build M=/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr module
es;
make[1]: Entering directory '/usr/src/linux-headers-4.4.0-31-generic'
CC [M] /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.o
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c: In function 'read_output':
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c:18:1: warning: no return statement in
function returning non-void [-Wreturn-type]
}
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c: At top level:
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c:9:23: warning: 'dir' defined but not u
sed [-Wunused-variable]
static struct dentry *dir, *output;
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c:9:29: warning: 'output' defined but no
t used [-Wunused-variable]
static struct dentry *dir, *output;
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c:10:28: warning: 'task' defined but not
used [-Wunused-variable]
static struct task_struct *task;
/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.c:12:16: warning: 'read_output' defined
but not used [-Wunused-function]
static ssize_t read_output(struct file *fp,
Building modules, stage 2.
MODPOST 1 modules
CC /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.mod.o
LD [M] /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/dbfs_paddr.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.4.0-31-generic'
gcc -o app app.c;
sudo insmod dbfs_paddr.ko
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr # make clean
make -C /lib/modules/4.4.0-31-generic/build M=/home/spta/Downloads/CSAP_KernelLab/skeleton/paddr clean
;
make[1]: Entering directory '/usr/src/linux-headers-4.4.0-31-generic'
CLEAN /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/.tmp_versions
CLEAN /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr/Module.symvers
make[1]: Leaving directory '/usr/src/linux-headers-4.4.0-31-generic'
rm app;
sudo rmmod dbfs_paddr.ko
root@spta-VirtualBox: /home/spta/Downloads/CSAP_KernelLab/skeleton/paddr #
```

```
rofile="unconfined" name="/usr/lib/lightdm/lightdm-guest-session" pid=413 comm="apparmor_parser"
[5.441995] audit: type=1400 audit(1555842325.756:3): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/lightdm/lightdm-guest-session//chromium" pid=413 comm="apparmor_pa
rser"
[5.445794] audit: type=1400 audit(1555842325.760:4): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/sbin/dhclient" pid=415 comm="apparmor_parser"
[5.445797] audit: type=1400 audit(1555842325.760:5): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/NetworkManager/nm-dhcp-client.action" pid=415 comm="apparmor_parse
r"
[5.445799] audit: type=1400 audit(1555842325.760:6): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/NetworkManager/nm-dhcp-helper" pid=415 comm="apparmor_parser"
[5.445801] audit: type=1400 audit(1555842325.760:7): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/lib/connman/scripts/dhclient-script" pid=415 comm="apparmor_parser"
[5.463102] audit: type=1400 audit(1555842325.776:8): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/bin/evince" pid=417 comm="apparmor_parser"
[5.463106] audit: type=1400 audit(1555842325.776:9): apparmor="STATUS" operation="profile_load" p
rofile="unconfined" name="/usr/bin/evince//sanitized_helper" pid=417 comm="apparmor_parser"
[5.463108] audit: type=1400 audit(1555842325.776:10): apparmor="STATUS" operation="profile_load"
profile="unconfined" name="/usr/bin/evince-previewer" pid=417 comm="apparmor_parser"
[5.743474] vboxdrvHeartBeatInit: Setting up heartbeat to trigger every 2000 milliseconds
[5.743547] Input: Unspecified device as /devices/pci0000:00/0000:00:04.0/input/input7
[5.769195] vboxguest: msc device minor 55, IRQ 20, I/O port d020, MMIO at 00000000f0400000 (size
0x400000)
[5.769197] vboxguest: Successfully loaded version 5.0.18_Ubuntu (interface 0x00010004)
[5.788740] random: nonblocking pool is initialized
[5.891797] [drm] Initialized drm 1.1.0 20060810
[5.901461] piix4_smbus 0000:00:07.0: SMBus Host Controller at 0x4100, revision 0
[5.954858] AVX2 version of gcm_enc/dec engaged.
[5.954860] AES CTR mode by8 optimization enabled
[5.957749] [drm] VRAM 01000000
[5.971386] [TTM] Zone kernel: Available graphics memory: 508136 kiB
[5.971389] [TTM] Initializing pool allocator
[5.971392] [TTM] Initializing DMA pool allocator
[5.976199] fbcon: vboxdrmfb (fb0) is primary device
[6.043385] Console: switching to colour frame buffer device 100x37
[6.048086] vboxvideo 0000:00:02.0: fb0: vboxdrmfb frame buffer device
[6.048625] [drm] Initialized vboxvideo 1.0.0 20130823 for 0000:00:02.0 on minor 0
[6.049011] snd_intel8x0 0000:00:05.0: disable (unknown or VT-d) VM optimization
[6.061349] intel_rapl: no valid rapl domains found in package 0
[6.062932] snd_intel8x0 0000:00:05.0: white list rate for 1028:0177 is 48000
[6.775637] Adding 1046524k swap on /dev/sda5. Priority: -1 extents:1 across:1046524k FS
[8.085821] IPV6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[8.087517] IPV6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[8.088602] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[8.088667] IPV6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[8.758092] floppy0: no floppy controllers found
[8.758125] work still pending
[271.472653] dbfs_ptree: module verification failed: signature and/or required key missing - tainti
ng kernel
[271.472785] dbfs_ptree module initialize done
[344.645272] dbfs_paddr module initialize done
[348.375903] dbfs_paddr module exit
```

# Demo

---

# etc

- sudo apt-get install error

```
yschoi@yschoi-VirtualBox:~$ sudo apt-get install tmux
E: Could not get lock /var/lib/dpkg/lock-frontent - open (11: Resource temporarily unavailable)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontent), is another process using it?
```

- Fix

```
root@yschoi-VirtualBox:/home/yschoi# sudo rm /var/lib/apt/lists/lock
root@yschoi-VirtualBox:/home/yschoi# sudo rm /var/cache/apt/archives/lock
root@yschoi-VirtualBox:/home/yschoi# sudo rm /var/lib/dpkg/lock
root@yschoi-VirtualBox:/home/yschoi# dpkg --configure -a
```

# References

---

- Linux Kernel Module Programming Guide
  - <http://www.tldp.org/LDP/lkmpg/2.6/html/>
- Debugfs APIs
  - <https://www.kernel.org/doc/Documentation/filesystems/debugfs.txt>
- Makefile Guide
  - <https://www.cs.duke.edu/~ola/courses/programming/Makefiles/Makefiles.html>