Operating Systems–II: CS3523
Spring 2020
Lab Assignment 2: Linux Kernel Debugger and Process Management
Last date for submission: N/A

Setup:

a. Install virt-manager on your laptop.
b. Import or Install CentOS 7:
   Import:
   i. Download pre-installed CentOS 7 with kernel debugger setup from : http://jenkins.cse.iith.ac.in:8080/userContent/cs3523-1.qcow2
   ii. Import the image into a VM

```
$ virt-install --import --memory 4096  --vcpus 2 --name cs3523-1 --disk ./cs3523-1.qcow2 --os-variant rhel7
```

   Install
   i. Download CentOS 7 ISO image from: http://intranet.iith.ac.in/files/os/CentOS-7-x86_64-DVD-1908.iso
   ii. Create a virtual machine with 4 GiB RAM, 10GB HDD (dynamic vdi), 2 CPU, 1 network cards (NAT)
   iii. Install Centos 7 64-bit on it (minimal centos / no GUI, enable kdump) . Check the kernel version.
   iv. Yum install crash, gcc, perl, strace rpms.

```
# yum install -y strace crash vim gcc perl
```

   v. Update all packages

```
# yum update
```

   vi. Reboot
   vii. Install kernel-debuginfo and kernel-debuginfo-common packages

```
# yum --enablerepo=base-debuginfo install -y kernel-debuginfo-$(uname -r)
```

c. You should be able to login via ssh to the IP address of the VM.
   User: root, Passwd: r123
d. Do your experiments in the VM.

NOTE: If you don't use VM setup, your laptop may hang, crash, and go bad. Use centos in VM, as Ubuntu is not suitable for kernel programmers.

Problem Statement:

Background:

- Linux maintains separate process descriptor (struct task) for each process and thread.

Goal: The goal of this assignment is to task structures of processes using kernel debugger.

Details: On the virtual machine running centos, run the following experiments.

Experiment #1:

a. Collect kernel dump of the running system using “echo c > /proc/sysrq-trigger”
b. The system reboots and dump is captured in a time-stamped directory under /var/crash/
c. Launch kernel debugger “crash” on the kdump collected using

```
# cd /var/crash/<dumpdirectory>
# crash /usr/lib/debug/usr/lib/modules/`uname -r`/vmlinux ./vmcore
```

```
crash> help

*               extend          log             rd              task
alias           files           mach            repeat          timer
ascii           foreach         mod             runq            tree
bpf             fuser           mount           search          union
bt              gdb             net             set             vm
btop            help            p               sig             vtop
dev             ipcs            ps              struct          waitq
dis             irq             pte             swap            whatis
eval            kmem            ptob            sym             wr
exit            list            ptov            sys             q

crash version: 7.2.3-10.el7   gdb version: 7.6
For help on any command above, enter "help <command>".
For help on input options, enter "help input".
For help on output options, enter "help output".
```

d. Try out a few commands like “log”, “bt”, “dev”, “p”, “ipcs”

e. Quit from kernel debugger

```
crash> quit
```

Experiment #2:

a. Run some processes in background

```
# cat &
# cat &
# top &
# vi abc.txt &
```

b. Collect kernel dump of the running system using “echo c > /proc/sysrq-trigger”
c. The system reboots and dump is captured in a time-stamped directory under /var/crash/
d. Launch kernel debugger “crash” on the kdump collected using

```
# cd /var/crash/<dumpdirectory>
# crash /usr/lib/debug/usr/lib/modules/`uname -r`/vmlinux ./vmcore
```

e. Try out commands “ps”, “runq”, “task”, “set”

```
crash> help ps
crash> ps
   PID  PPID  CPU       TASK           ST  %MEM       VSZ     RSS  COMM
       0      0   0  ffffffffb0e18480  RU   0.0    0       0  [swapper/0]
>      0      0   1  ffff9b0cb94941c0  RU   0.0    0       0  [swapper/1]
```

```
      1       0   0  ffff9b0cb9450000  IN   0.1  127940   6568  systemd
      2       0   1  ffff9b0cb9451070  IN   0.0    0       0  [kthreadd]
      4       2   0  ffff9b0cb9453150  IN   0.0    0       0  [kworker/0:0H]
      ...
>   1642      811   0  ffff9b0bb60141c0  RU   0.0  115448   2100  bash
    1828      1   1  ffff9b0c3778b150  IN   0.0   39084   1540  systemd-journal
    1833      2   1  ffff9b0cb5c641c0  IN   0.0    0       0  [kworker/u4:0]
    1844      1   1  ffff9b0bb66362a0  IN   0.0   26380   1772  systemd-logind
    1946   1469   1  ffff9b0c376fc1c0  IN   0.1   89804   4068  pickup
    1968      2   1  ffff9b0c376f9070  IN   0.0    0       0  [kworker/1:0]
    1989      2   0  ffff9b0cb720a0e0  RU   0.0    0       0  [kworker/0:1]
    1992      2   1  ffff9b0cb5d041c0  IN   0.0    0       0  [kworker/1:2]
    1993      2   1  ffff9b0cb5d062a0  IN   0.0    0       0  [kworker/1:1]
    2015   1642   1  ffff9b0c376fe2a0  ST   0.0  107976   600  cat
    2016   1642   1  ffff9b0c376fa0e0  ST   0.0  107976   600  cat
    2017   1642   0  ffff9b0c376fb150  ST   0.0  159364   1784  top
    2018   1642   0  ffff9b0c376f8000  ST   0.1  149004   4956  vim
    2021      2   1  ffff9b0cb94541c0  IN   0.0    0       0  [kworker/1:3]
```

```
crash> help runq
crash> runq
CPU 0 RUNQUEUE: ffff9b0cbfc1ac80
  CURRENT: PID: 1642   TASK: ffff9b0bb60141c0  COMMAND: "bash"
  RT PRIO_ARRAY: ffff9b0cbfc1ae20
       [no tasks queued]
  CFS RB_ROOT: ffff9b0cbfc1ad28
       [120] PID: 1989   TASK: ffff9b0cb720a0e0  COMMAND: "kworker/0:1"

CPU 1 RUNQUEUE: ffff9b0cbfd1ac80
  CURRENT: PID: 0   TASK: ffff9b0cb94941c0  COMMAND: "swapper/1"
  RT PRIO_ARRAY: ffff9b0cbfd1ae20
       [no tasks queued]
  CFS RB_ROOT: ffff9b0cbfd1ad28
       [no tasks queued]

crash> runq -m
 CPU 0: [0 00:00:00.001]  PID: 1642   TASK: ffff9b0bb60141c0  COMMAND: "bash"
 CPU 1: [0 05:46:06.069]  PID: 0        TASK:    ffff9b0cb94941c0       COMMAND:
"swapper/1"

crash> runq -t
 CPU 0: 20766064292679
       20766063199929  PID: 1642   TASK: ffff9b0bb60141c0  COMMAND: "bash"
 CPU 1: 20766069904525
       00000000000000  PID: 0     TASK: ffff9b0cb94941c0  COMMAND: "swapper/1"

crash> runq -g
CPU 0
  CURRENT: PID: 1642   TASK: ffff9b0bb60141c0  COMMAND: "bash"
  ROOT_TASK_GROUP: ffffffffb1307c00  RT_RQ: ffff9b0cbfc1ae20
       [no tasks queued]
```

```
  ROOT_TASK_GROUP: ffffffffb1307c00  CFS_RQ: ffff9b0cbfc1ad00
       [120] PID: 1642   TASK: ffff9b0bb60141c0  COMMAND: "bash" [CURRENT]
       [120] PID: 1989   TASK: ffff9b0cb720a0e0  COMMAND: "kworker/0:1"

CPU 1
  CURRENT: PID: 0   TASK: ffff9b0cb94941c0  COMMAND: "swapper/1"
  ROOT_TASK_GROUP: ffffffffb1307c00  RT_RQ: ffff9b0cbfd1ae20
       [no tasks queued]
  ROOT_TASK_GROUP: ffffffffb1307c00  CFS_RQ: ffff9b0cbfd1ad00
       [no tasks queued]
```

```
crash> help task

crash> whatis task_struct

crash> whatis wait_queue_head_t
typedef struct __wait_queue_head {
       spinlock_t lock;
       struct list_head task_list;
} wait_queue_head_t;
SIZE: 24

crash> waitq <pointer>
```

f. Quit from kernel debugger

```
crash> quit
```