



دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

گزارش کار آزمایشگاه آزمایشگاه سیستم‌های عامل

گزارش آزمایش شماره ۳
(مشاهده رفتار هسته و سیستم عامل)

شماره گروه:	۲۰
گروه:	ارشیا یوسف‌نیا (۴۰۱۱۱۰۴۱۵)
استاد درس:	محمد عارف زارع زاده (۴۰۱۱۰۶۰۱۷)
تاریخ:	دکتر بیگی
تاریخ:	تابستان ۱۴۰۴

فهرست مطالب

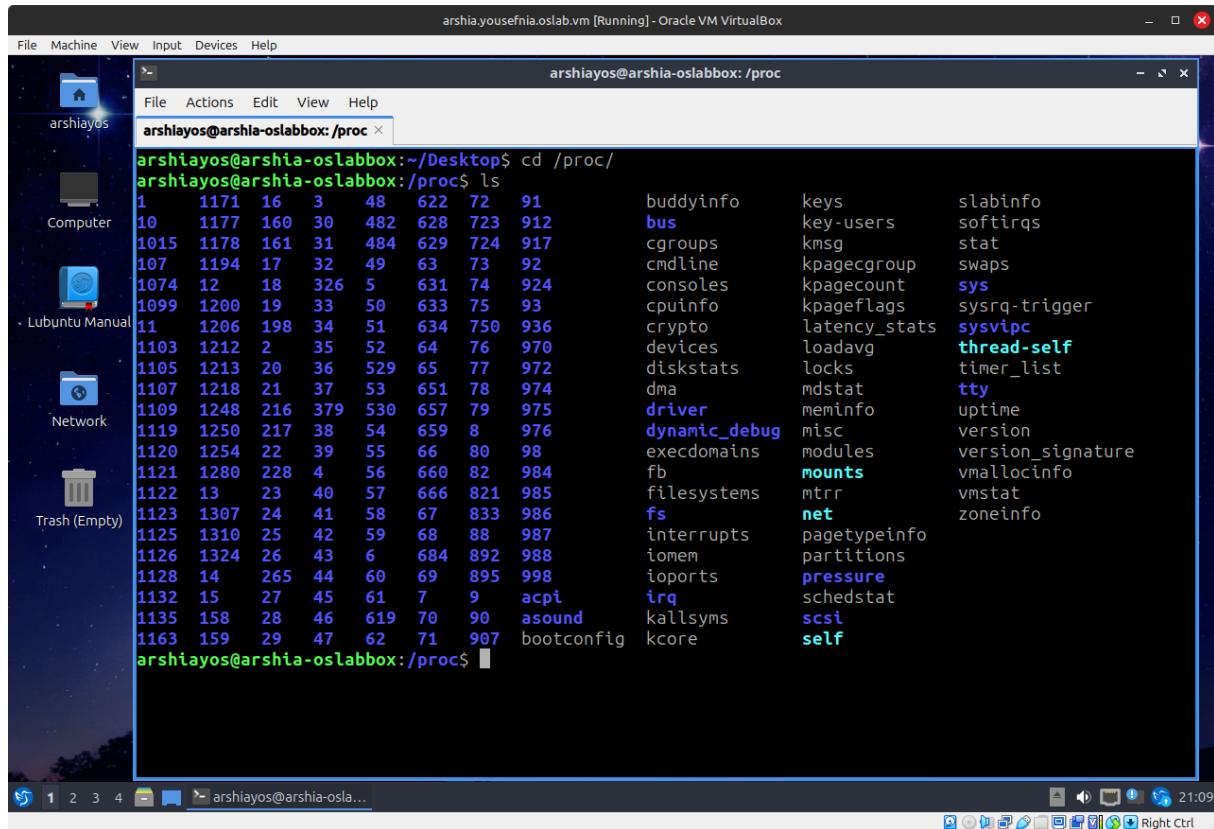
۱	۱ مشاهده فایل سیستم /proc
۲	۲ مشاهده محتویات یک فایل در شاخه /proc
۶	۳ مشاهده وضعیت پردازه ها
۱۲	۱.۳ تمرین
۱۶	۴ مشاهده اطلاعات مربوط به هسته
۲۴	۱.۴ تمرین

لیست تصاویر

۱	/proc	۱
۲	/proc/version	۲
۲	/proc/devices	۳
۳	/proc/filesystems	۴
۴	/proc/meminfo	۵
۴	/proc/swaps, /proc/partitions	۶
۵	یک برنامه cpp برای خواندن محتوای /proc/version و انتقال آن به فایل LinuxVersion.txt		۷
۵	/proc	۸
۷	یک پوشه مربوط به یک پردازه	۹
۸	محتوای فایل های مختلف در پوشه پردازه	۱۰
۹	محتوای فایل های مختلف در پوشه پردازه	۱۱
۱۰	محتوای فایل های مختلف در پوشه پردازه	۱۲
۱۱	یک script برای خروجی دادن لیست شماره پردازه ها و نام آنها	۱۳
۱۲	اجرای اسکریپت شکل ۱۳	۱۴
۱۳	کد برنامه دریافت شماره پردازه و خروجی اطلاعات مربوط به پردازه	۱۵
۱۴	ادامه کد برنامه دریافت شماره پردازه و خروجی اطلاعات مربوط به پردازه	۱۶
۱۵	کامپایل و اجرای یک نمونه برای برنامه شکل ۱۵	۱۷
۱۷	چاپ محتوای مسیر /proc و /proc/cpuinfo	۱۸
۱۷	چاپ محتوای /proc/cmdline و /proc/ioports	۱۹
۱۸	چاپ محتوای /proc/meminfo	۲۰
۱۹	چاپ محتوای /proc/stat و /proc/uptime و /proc/loadavg و /proc/version	۲۱
۱۹	چاپ محتوای /proc/interrupts	۲۲
۲۰	چاپ محتوای /proc/filesystems و /proc/net	۲۳
۲۱	برنامه چاپ مدل و مشخصات پردازنده	۲۴
۲۲	کامپایل و اجرای برنامه شکل ۲۴	۲۵
۲۳	برنامه چاپ وضعیت حافظه	۲۶
۲۴	کامپایل و اجرای برنامه شکل ۲۶	۲۷

۱ مشاهده فایل سیستم /proc

در شکل ۱ به پوشه گفته شده رفتیم و لیستی از فایل ها و پوشه های داخل آن را دیدیم.

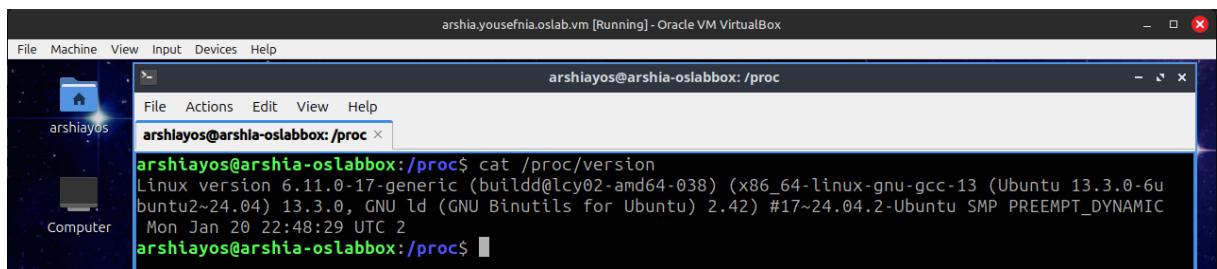


The screenshot shows a terminal window titled "arshiyos@arshia-oslabbox: /proc". The terminal is running on a Lubuntu desktop environment. The command "ls" is being run to list the contents of the /proc directory. The output shows numerous files and sub-directories, including "1", "10", "1015", "107", "1074", "1099", "11", "1103", "1105", "1107", "1109", "1119", "1120", "1121", "1122", "1123", "1125", "1126", "1128", "1132", "1135", "1163", "buddyinfo", "bus", "cgroups", "cmdline", "consoles", "cpuinfo", "crypto", "devices", "diskstats", "dma", "driver", "dynamic_debug", "execdomains", "fb", "filesystems", "fs", "interrupts", "iomem", "ioports", "irq", "kallsyms", "bootconfig", "kcore", "keys", "key-users", "kmsg", "kpagegroup", "kpagecount", "kpageflags", "latency_stats", "loadavg", "locks", "mdstat", "meminfo", "misc", "modules", "mounts", "mtrr", "net", "pagetypeinfo", "partitions", "pressure", "schedstat", "scsi", and "self". The terminal prompt is "arshiyos@arshia-oslabbox: /proc\$".

شکل ۱: محتوای مسیر /proc

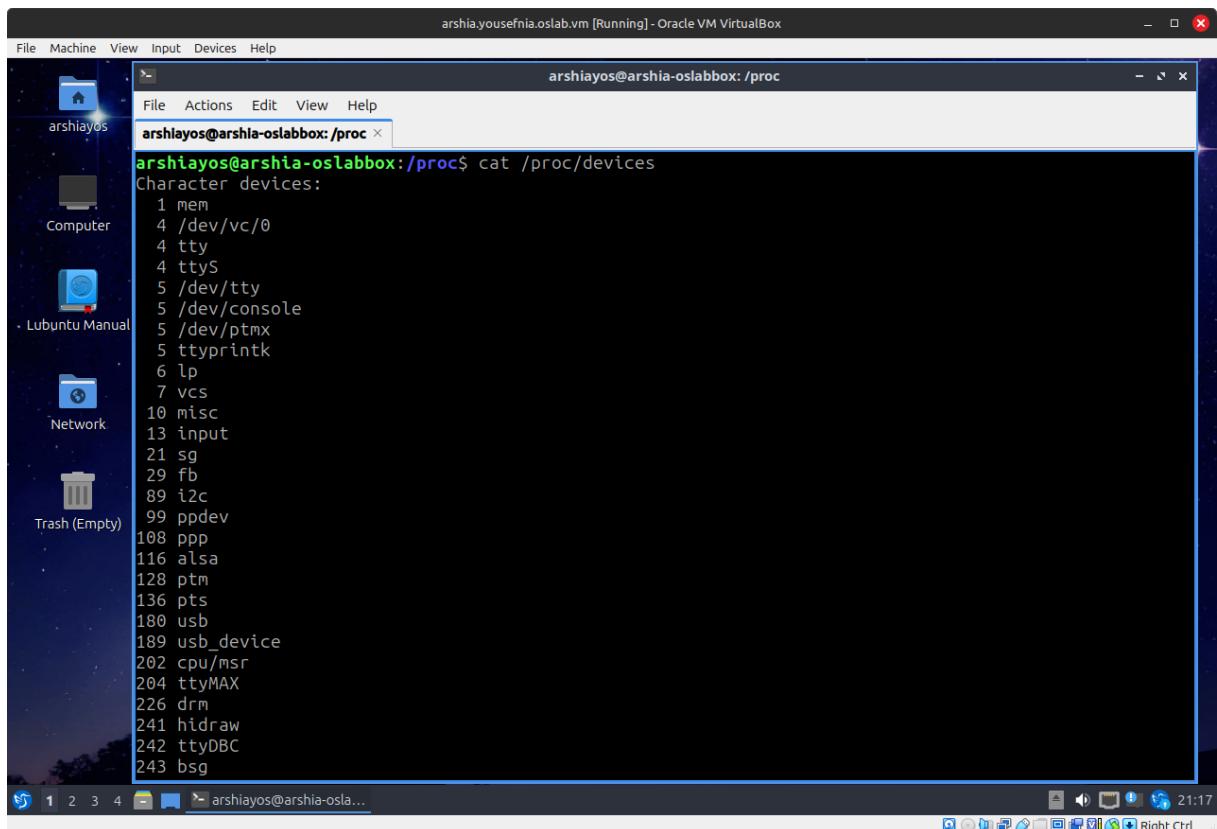
۲ مشاهده محتويات یک فايل در شاخه /proc

در شکل ۲ محتوای فایل `version` چاپ شده است. همچنین در شکل های ۳، ۴، ۵، و ۶ هم محتوای بعضی دیگر از فایل های اين مسیر آمده است. فایل `version` اطلاعات مربوط به نسخه هسته سیستم عامل و دیگر مشخصات آن است. فایل های دیگری هم که در شکل های بعدی آمده اند در بخش آخر این گزارش بیشتر بررسی شده اند اما به طور کلی جنبه های مختلف سیستم عامل از جمله حافظه، فایل سیستم ها، دستگاه های متصل، و موارد دیگر را نشان می دهند.



```
arshiyos@arshia-oslabbox: /proc$ cat /proc/version
Linux version 6.11.0-17-generic (buildd@lcy02-amd64-038) (x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6u
buntu2~24.04) 13.3.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #17~24.04.2-Ubuntu SMP PREEMPT_DYNAMIC
Mon Jan 20 22:48:29 UTC 2
arshiyos@arshia-oslabbox: /proc$
```

شکل ۲: چاپ محتوای /proc/version



```
arshiyos@arshia-oslabbox: /proc$ cat /proc/devices
Character devices:
  1 mem
  4 /dev/vc/0
  4 tty
  4 ttys
  5 /dev/tty
  5 /dev/console
  5 /dev/ptmx
  5 ttyprintk
  6 lp
  7 vcs
 10 misc
13 input
21 sg
29 fb
89 i2c
99 ppdev
108 ppp
116 alsa
128 ptm
136 pts
180 usb
189 usb_device
202 cpu/msr
204 ttyMAX
226 drm
241 hidraw
242 ttyDBC
243 bsg
```

شکل ۳: چاپ محتوای /proc/devices

The screenshot shows a terminal window titled "arshiyos@arshia-oslabbox: /proc". The window is part of a desktop environment with a dark blue background. On the left, there's a file manager sidebar showing icons for Home, Computer, Lubuntu Manual, Network, and Trash (Empty). The terminal itself displays the command "cat /proc/filesystems" followed by its output, which lists various file systems including sysfs, tmpfs, bdev, proc, cgroup, cgroup2, cpuset, devtmpfs, configfs, debugfs, tracefs, securityfs, sockfs, bpf, pipes, ramfs, hugepaged, devpts, ext3, ext2, ext4, squashfs, vfat, encryptfs, fuseblk, fuse, fusectl, efivarfs, and nqueue.

```
arshiyos@arshia-oslabbox: /proc$ cat /proc/filesystems
nodev    sysfs
nodev    tmpfs
nodev    bdev
nodev    proc
nodev    cgroup
nodev    cgroup2
nodev    cpuset
nodev    devtmpfs
nodev    configfs
nodev    debugfs
nodev    tracefs
nodev    securityfs
nodev    sockfs
nodev    bpf
nodev    pipes
nodev    ramfs
nodev    hugepaged
nodev    devpts
ext3
ext2
ext4
squashfs
vfat
nodev  encryptfs
fuseblk
nodev  fuse
nodev  fusectl
nodev  efivarfs
nodev  nqueue
```

شکل ۴: چاپ محتوای /proc/filesystems

```

arshiyos@arshia-oslabbox: /proc$ cat /proc/meminfo
MemTotal:       8131900 kB
MemFree:        7192808 kB
MemAvailable:   7482304 kB
Buffers:         50388 kB
Cached:          454088 kB
SwapCached:      0 kB
Active:          665536 kB
Inactive:        119860 kB
Active(anon):    289668 kB
Inactive(anon):  0 kB
Active(file):   375868 kB
Inactive(file): 119860 kB
Unevictable:     0 kB
Mlocked:         0 kB
SwapTotal:      524284 kB
SwapFree:        524284 kB
Zswap:           0 kB
Zswapped:        0 kB
Dirty:            0 kB
Writeback:        0 kB
AnonPages:      280932 kB
Mapped:          198804 kB
Shmem:           8748 kB
KReclaimable:   25516 kB
Slab:            84600 kB
SReclaimable:   25516 kB
SUnreclaim:     59684 kB
KernelStack:     4592 kB
PageTables:     10416 kB

```

شکل ۵: چاپ محتوای /proc/meminfo

```

arshiyos@arshia-oslabbox: /proc$ cat /proc/swaps
Filename      Type      Size      Used      Priority
/swappfile    file      524284   0         -2

arshiyos@arshia-oslabbox: /proc$ cat /proc/partitions
major minor #blocks name
11       0    1048575 sr0
8        0    57700288 sda
8        1    57696423 sda1

```

شکل ۶: چاپ محتوای /proc/swaps, /proc/partitions

در شکل ۷ یک برنامه برای خواندن فایل مربوط به نسخه و ریختن آن در فایل دیگر به همراه مراحل کامپایل، اجرا، و نتیجه نهایی آورده‌ایم.

The screenshot shows a terminal window titled "arshia.yousefnia.oslab.vm [Running] - Oracle VM VirtualBox". The terminal session starts with the command `cat versionreader.cpp` which contains code to read from /proc/version and write to LinuxVersion.txt. It then lists files in the current directory, compiles the program with g++, runs it, and finally displays the contents of LinuxVersion.txt.

```

arshiyos@arshia-oslabbox:~$ cat versionreader.cpp
#include <iostream>
#include <fstream>
#include <string>

int main() {
    std::ifstream input("/proc/version");
    std::ofstream output("LinuxVersion.txt");
    std::string line;
    while (std::getline(input, line)) {
        output << line << '\n';
    }
    input.close();
    output.close();
    return 0;
}
arshiyos@arshia-oslabbox:~$ ls
Desktop  Music  syscalls      testdir      testsyscall.cpp
Documents Pictures Templates  testhellosyscall.cpp  versionreader.cpp
Downloads Public  testaddsyscall.cpp  testsyscall2.cpp  Videos
arshiyos@arshia-oslabbox:~$ g++ versionreader.cpp
arshiyos@arshia-oslabbox:~$ ./a.out
arshiyos@arshia-oslabbox:~$ cat LinuxVersion.txt
Linux version 6.11.0-17-generic (buildd@lcy02-amd64-038) (x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6u
buntu2+24.04) 13.3.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #17~24.04.2-Ubuntu SMP PREEMPT_DYNAMIC
Mon Jan 20 22:48:29 UTC 2
arshiyos@arshia-oslabbox:~$ 

```

شکل ۷: یک برنامه cpp برای خواندن محتوای /proc/version و انتقال آن به فایل LinuxVersion.txt

همانطور که در شکل ۸ نشان داده شده امکان نوشتن و تغییر در این فایل‌ها و محتوای در مسیر ویژه /proc نداریم.

The screenshot shows a terminal window titled "arshia.yousefnia.oslab.vm [Running] - Oracle VM VirtualBox". The user tries to echo "salam doostan!" to /proc/version using both regular echo and sudo echo, but receives "Permission denied" errors.

```

arshiyos@arshia-oslabbox:~$ echo "salam doostan!" > /proc/version
bash: /proc/version: Permission denied
arshiyos@arshia-oslabbox:~$ sudo echo "salam doostan!" > /proc/version
bash: /proc/version: Permission denied
arshiyos@arshia-oslabbox:~$ 

```

شکل ۸: نبود دسترسی تغییر در فایل‌های مسیر /proc

۳ مشاهده وضعیت پردازه‌ها

شکل ۹ محتوای پوشه یک پردازه را نشان می‌دهد. شکل‌های ۱۰، ۱۱، و ۱۲ به جزییات فایل‌های موجود برای هر پردازه می‌پردازد.

cmdline

آرگومان‌های خط فرمان که پردازه با آن اجرا شده است.

environ

متغیرهای محیطی مربوط به پردازه.

stat

اطلاعات کلی پردازه مانند، PID وضعیت اجرا، مصرف CPU و غیره.

status

اطلاعات خلاصه‌شده و قابل خواندن درباره وضعیت پردازه.

statm

اطلاعات حافظه مورد استفاده پردازه (بر حسب صفحات حافظه).

cwd

.(Current Working Directory) لینک به مسیر کاری فعلی پردازه

exe

لینک به فایل اجرایی پردازه.

root

ریشه مجازی پردازه (برای chroot شده‌ها مفید است).

arshiyas@arshia:yousefnia.oslab.vm [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

arshiyas@arshia-oslabbox: /proc/1088

File Actions Edit View Help

arshiyas@arshia-oslabbox: /proc/1088 x

101	1133	201	37	53	642	79	881	cpuinfo	kpageflags	sysrq-trigger	
1013	1135	21	38	535	645	792	889	crypto	latency_stats	sysvipc	
1014	1144	22	39	537	646	795	9	devices	loadavg	thread-self	
1015	1173	220	4	54	65	8	90	diskstats	locks	timer_list	
1017	1176	221	40	55	651	80	905	dma	mdstat	tty	
1018	1197	23	406	56	669	809	92	driver	meminfo	uptime	
1019	12	24	41	57	67	81	93	dynamic_debug	misc	version	
1020	1200	25	411	58	68	818	94	execdomains	modules	version_signature	
1022	1212	26	42	59	686	82	95	fb	mounts	vmallocinfo	
1027	13	268	429	6	687	821	964	filesystems	mtrr	vmstat	
1029	136	27	43	60	69	824	965	fs	net	zoneinfo	
1060	14	28	44	61	7	825	993	interrupts	pagetypeinfo		
1062	15	29	45	611	70	829	997	iomem	partitions		
1071	16	3	46	613	71	84	999	ioports	pressure		
1072	162	30	47	614	711	863	acpi	irq	schedstat		
1088	163	31	48	616	72	865	asound	kallsyms	scsi		
1094	164	32	489	618	73	868	bootconfig	kcore	self		
arshiyas@arshia-oslabbox: /proc\$ cd 1088											
arshiyas@arshia-oslabbox: /proc/1088\$ ls											
arch_status	cpuset	latency	ns	sched	syscall						
attr	cwd	limits	numa_maps	schedstat	task						
autogroup	environ	loginuid	oom_adj	sessionid	timens_offsets						
auxv	exe	map_files	oom_score	setgroups	timers						
cgroup	fd	maps	oom_score_adj	smaps	timerslack_ns						
clear_refs	fdinfo	mem	pagemap	smaps_rollup	uid_map						
cmdline	gid_map	mountinfo	patch_state	stack	wchan						
comm	io	mountstats	personality	stat							
coredump_filter	ksm_merging_pages	net	projid_map	statm							
cpu_resctrl_groups	ksm_stat	root	status								
arshiyas@arshia-oslabbox: /proc/1088\$											

شکل ۹: محتوای داخل یک پوشه مربوط به یک پردازه

```
arshiyos@arshia-oslabbox: /proc/1088
File Actions Edit View Help
arshiyos@arshia-oslabbox: /proc/1088
coredump_filter      ksm_merging_pages   mountstats projid_map      statm
cpu_resctrl_groups  ksm_stat           net          root       status
arshiyos@arshia-oslabbox: /proc/1088$ cat cmdline
/usr/libexec/gvfs-udisks2-volume-monitorarshiyos@arshia-oslabbox: /proc/1088$ ^C
arshiyos@arshia-oslabbox: /proc/1088$ cat environ
HOME=/home/arshiyosLANG=en_US.UTF-8LC_ADDRESS=en_US.UTF-8LC_IDENTIFICATION=en_US.UTF-8LC_MEASUREME
NT=en_US.UTF-8LC_MONETARY=en_US.UTF-8LC_NAME=en_US.UTF-8LC_NUMERIC=en_US.UTF-8LC_PAPER=en_US.UTF-8L
C_TELEPHONE=en_US.UTF-8LC_TIME=en_US.UTF-8LOGNAME=arshiyosPATH=/usr/local/sbin:/usr/local/bin:/usr
:/sbin:/usr/bin:/bin:/usr/games:/usr/local/games:/snap/binSHELL=/bin/bashUSER=arshiyosXDG_RUN
TIME_DIR=/run/user/1000QT_ACCESSIBILITY=1QTWEBENGINE_DICTIONARIES_PATH=/usr/share/hunspell-bdic/BRO
WSER=firefoxDBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/busDESKTOP_SESSION=LubuntuDISPLAY=:0G
PG_AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1GSM_SKIP_SSH_AGENT_WORKAROUND=trueGTK_CSD=0GTK_OV
ERLAY_SCROLLING=0LXQT_SESSION_CONFIG=sessionPWD=/home/arshiyosQT_PLATFORM_PLUGIN=lxqtQT_QPA_PLATFO
RMTHEME=lxqtSAL_USE_VCLPLUGIN=qtSSAL_VCL_QT5_USE_CAIRO=trueSHLVL=0SSH_AGENT_PID=965SSH_AUTH_SOCK=t
mp/ssh-928EucX1ml45/agent.881TERM=qterminalXAUTHORITY=/tmp/xauth_VJzwWzXDG_CACHE_HOME=/home/arshiy
os/.cacheXDG_CONFIG_DIRS=/etc/xdg/xdg-Lubuntu:/etc/xdg:/etc:/usr/shareXDG_CONFIG_HOME=/home/arshiy
os/.configXDG_CURRENT_DESKTOP=LXQtXDG_DATA_DIRS=/usr/share/Lubuntu:/usr/local/share:/usr/share/:/u
sr/shareXDG_DATA_HOME=/home/arshiyos/.local/shareXDG_MENU_PREFIX=lxqt-XDG_SEAT=seat0XDG_SEAT_PATH=
/org/freedesktop/DisplayManager/Seat0XDG_SESSION_CLASS=userXDG_SESSION_DESKTOP=LXQtXDG_SESSION_ID=1
XDG_SESSION_PATH=/org/freedesktop/DisplayManager/Session0XDG_SESSION_TYPE=x11XDG_VTNR=2,_/usr/bin/d
bus-update-activation-environmentMANAGERPID=868INVOCATION_ID=33c98deee2a646d79e88ef3026a14af9JOURNA
L_STREAM=9:12145SYSTEMD_EXEC_PID=1088MEMORY_PRESSURE_WATCH=/sys/fs/cgroup/user.slice/user-1000.slic
e/user@1000.service/session.slice/gvfs-udisks2-volume-monitor.service/memory.pressureMEMORY_PRESSUR
E_WRITE=c29tZSayMDAwMDAwMAA=arshiyos@arshia-oslabbox: /proc/1088$ ^C
arshiyos@arshia-oslabbox: /proc/1088$ cat stat
1088 (gvfs-udisks2-vo) S 868 1088 0 -1 4194304 1067 0 2 0 12 9 0 0 20 0 5 0 8480 398467072 248
4 18446744073709551615 105540686155776 105540686246193 140735400017408 0 0 0 4096 0 0 0 0 17 0 0
0 0 0 0 105540686280528 105540686295664 105541407334400 140735400020049 140735400020090 14073540002
0090 140735400021967 0
arshiyos@arshia-oslabbox: /proc/1088$
```

شکل ۱۰: محتوای فایل‌های مختلف در پوشه پردازه

arshlayos@arshla-oslabbox: /proc/1088 \$ cat status

```
Name: gvfs-udisks2-vo
Umask: 0002
State: S (sleeping)
Tgid: 1088
Ngid: 0
Pid: 1088
PPid: 868
TracerPid: 0
Uid: 1088 1088 1088 1088
Gid: 1088 1088 1088 1088
FDSize: 64
Groups: 4 24 27 38 40 113 988 1088
NSgid: 1088
NSpid: 1088
NSpgid: 1088
NSuid: 1088
Ktread: 0
VmPeak: 454604 kB
VmSize: 389128 kB
VmLck: 0 kB
VmPin: 0 kB
VmNM: 9930 kB
VmRSS: 9930 kB
RssAnon: 1288 kB
RssFile: 8050 kB
RssShmem: 0 kB
VmData: 42576 kB
VmStk: 132 kB
VmExe: 92 kB
VmLib: 8888 kB
VmPTE: 124 kB
VmSwap: 0 kB
HugetlbPages: 0 kB
CoreDumping: 0
THP_enabled: 1
untag_mask: exfffffffffffff
Threads: 5
SigQ: 0/31434
SigPnd: 0000000000000000
ShdPnd: 0000000000000000
SigBlk: 0000000000000000
SigIgn: 0000000000000000
SigCgt: 0000000100000000
CapInh: 0000000000000000
CapPrm: 0000000000000000
CapEff: 0000000000000000
CapBnd: 000001ffffffff
CapAnb: 0000000000000000
NoNewPrivs: 0
Seccomp: 0
Seccomp_filters: 0
Speculation_Store_Bypass: vulnerable
SpeculationIndirectBranch: always enabled
Cpus_allowed: 7
Cpus_allowed_list: 0-2
Mems_allowed: 00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00
000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00
000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,00
000001
Mems_allowed_list: 0
voluntary_ctxt_switches: 28
nonvoluntary_ctxt_switches: 32
x86_Thread_features:
x86_Thread_features_locked:
```

arshlayos@arshla-oslabbox: /proc/1088 \$

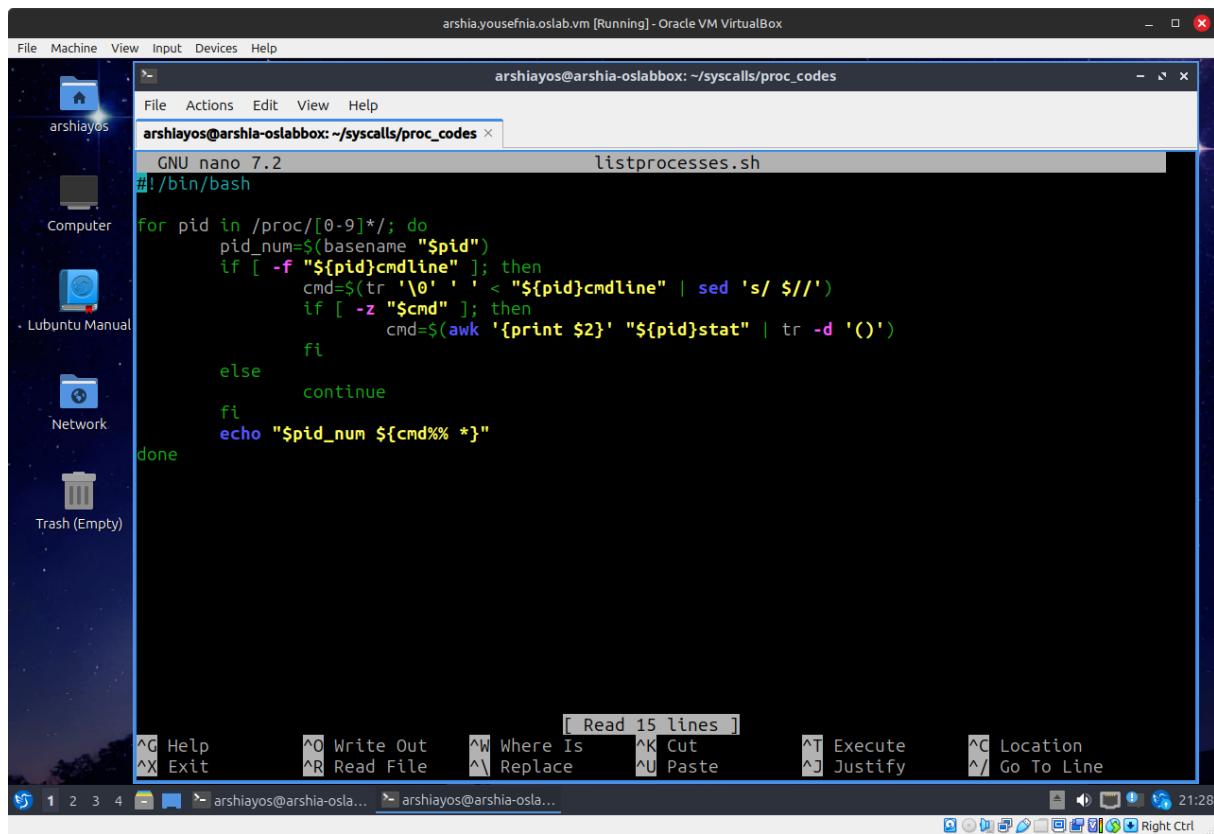
شکل ۱۱: محتوای فایل‌های مختلف در پوشه پردازه

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "arshiyos@arshia-oslabbox: /proc/1088". The terminal content displays the output of several commands:

```
arshiyos@arshia-oslabbox:/proc/1088$ cat statm
97282 2484 2164 23 0 10677 0
arshiyos@arshia-oslabbox:/proc/1088$ ls cwd
a.out    Downloads    Pictures    Templates    testhellosyscall.cpp    versionreader.cpp
Desktop   LinuxVersion.txt  Public    testaddsyscall.cpp  testsyscall2.cpp    Videos
Documents  Music    syscalls  testdir    testsyscall.cpp
arshiyos@arshia-oslabbox:/proc/1088$ ls root
bin      dev    lib    lost+found    opt    run        srv    tmp
bin usr-is-merged etc    lib64    media    proc    sbin    swapfile  usr
boot     home  lib usr-is-merged  mnt    root    sbin usr-is-merged  sys    var
arshiyos@arshia-oslabbox:/proc/1088$ file exe
exe: symbolic link to /usr/libexec/gvfs-udisks2-volume-monitor
arshiyos@arshia-oslabbox:/proc/1088$ file /usr/libexec/gvfs-udisks2-volume-monitor
/usr/libexec/gvfs-udisks2-volume-monitor: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=6b6fb3c912d1fc089d7051514efe777ae00e674d, fo
r GNU/Linux 3.2.0, stripped
arshiyos@arshia-oslabbox:/proc/1088$
```

شکل ۱۲: محتوای فایل‌های مختلف در پوشه پردازه

شکل ۱۳ یک اسکریپت برای خواندن از مسیر `/proc` و خروجی دادن لیست پردازه‌ها به همراه نام آنهاست.
در ادامه شکل ۱۴ یک نمونه اجرای آن است.



```
GNU nano 7.2 arshiyos@arshia-oslabbox: ~/syscalls/proc_codes
```

```
#!/bin/bash

for pid in /proc/[0-9]*; do
    pid_num=$(basename "$pid")
    if [ -f "${pid} cmdline" ]; then
        cmd=$(tr '\0' ' ' < "${pid} cmdline" | sed 's/ $//')
        if [ -z "$cmd" ]; then
            cmd=$(awk '{print $2}' "${pid} stat" | tr -d '(')
        fi
    else
        continue
    fi
    echo "$pid_num ${cmd% *}"
done
```

[Read 15 lines]

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

شکل ۱۳: یک script برای خروجی دادن لیست شماره پردازه‌ها و نام آن‌ها

The screenshot shows a terminal window titled "arshiyos@arshia-oslabbbox: ~/syscalls/proc_codes". The terminal displays the following command-line session:

```

arshiyos@arshia-oslabbbox:~/syscalls/proc_codes$ mkdir proc_codes
arshiyos@arshia-oslabbbox:~/syscalls/proc_codes$ cd proc_codes/
arshiyos@arshia-oslabbbox:~/syscalls/proc_codes$ nano listprocesses.sh
arshiyos@arshia-oslabbbox:~/syscalls/proc_codes$ ls
listprocesses.sh
arshiyos@arshia-oslabbbox:~/syscalls/proc_codes$ sh listprocesses.sh
1 /sbin/init
1000 /usr/bin/openbox
1004 /usr/libexec/at-spi-bus-launcher
1006 /usr/bin/python3
1008 /usr/libexec/geoclue-2.0/demos/agent
1010 /usr/bin/lubuntu-update
1016 /usr/bin/dbus-daemon
1021 /usr/bin/pcmanfm-qt
1022 /usr/bin/lxqt-globalkeysd
1023 /usr/bin/lxqt-notificationd
1024 /usr/bin/lxqt-panel
1025 /usr/bin/lxqt-policykit-agent
1027 /usr/bin/lxqt-runner
1029 /usr/bin/xscreensaver
1031 /usr/bin/python3
1037 xscreensaver-systemd
1059 /usr/libexec/at-spi2-registryd
107 kworker/R-charger_manager
1075 /usr/libexec/gvfsd
1081 /usr/libexec/gvfsd-fuse
1082 /usr/libexec/gvfsd-trash
1099 /usr/libexec/gvfs-udisks2-volume-monitor
11 kworker/u12:0
1105 /usr/libexec/gvfs-afc-volume-monitor

```

The terminal window is part of a desktop environment with icons for Home, Computer, Lubuntu Manual, Network, and Trash (Empty) visible on the left. The bottom status bar shows the date and time as 21:27.

شکل ۱۴: اجرای اسکریپت شکل ۱۳

۱.۳ تمرین

در این تمرین یک برنامه نوشته‌ایم که شماره یک پردازه را ورودی میگیرد و اطلاعاتی اعم از نام فایل اجرایی آن، مقدار حافظه مصرفی به بایت، پارامترهای اجرا و متغیرهای محیطی مربوط به آن در خروجی چاپ می‌کند. شکل‌های ۱۵ و ۱۶ کد منبع این برنامه را نشان می‌دهند. در نهایت نتیجه کامپایل و یک نمونه اجرا در شکل ۱۷ آمده است.

```
File Actions Edit View Help
arshiyos@arshia-oslabbox: ~/syscalls/proc_codes ×
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ ls
listprocesses.sh process_info.c
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ cat process_info.c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <dirent.h>
#include <limits.h>
#include <sys/types.h>

#define MAX_LINE 4096

void print_process_info(pid_t pid) {
    char path[PATH_MAX];
    FILE *file;
    char line[MAX_LINE];

    snprintf(path, sizeof(path), "/proc/%d/exe", pid);
    char exe[PATH_MAX] = {0};
    ssize_t len = readlink(path, exe, sizeof(exe)-1);
    if (len == -1) {
        perror("readlink");
    } else {
        exe[len] = '\0';
        printf("Executable: %s\n", exe);
    }

    snprintf(path, sizeof(path), "/proc/%d/statm", pid);
    file = fopen(path, "r");
    if (file) {
        unsigned long size, resident, shared, text, lib, data, dt;
        fscanf(file, "%lu %lu %lu %lu %lu %lu", &size, &resident, &shared, &text, &lib, &data, &dt);
        fclose(file);
        long page_size = sysconf(_SC_PAGESIZE);
        printf("Memory Usage:\n");
        printf(" Resident Set Size: %lu bytes\n", resident * page_size);
        printf(" Virtual Memory Size: %lu bytes\n", size * page_size);
    } else {
        perror("fopen statm");
    }

    snprintf(path, sizeof(path), "/proc/%d/cmdline", pid);
    file = fopen(path, "r");
}
```

شکل ۱۵: کد برنامه دریافت شماره پردازه و خروجی اطلاعات مربوط به پردازه

```
File Actions Edit View Help
arshiyos@arshia-oslabbox: ~/syscalls/proc_codes ×
if (file) {
    printf("Command Line Parameters:\n");
    while (fgets(line, sizeof(line), file)) {
        char *param = line;
        while (*param) {
            printf(" %s\n", param);
            param += strlen(param) + 1;
        }
    }
    fclose(file);
} else {
    perror("fopen cmdline");
}

snprintf(path, sizeof(path), "/proc/%d/environ", pid);
file = fopen(path, "r");
if (file) {
    printf("Environment Variables:\n");
    while (fgets(line, sizeof(line), file)) {
        char *var = line;
        while (*var) {
            printf(" %s\n", var);
            var += strlen(var) + 1;
        }
    }
    fclose(file);
} else {
    perror("fopen environ");
}
}

int main(int argc, char *argv[]) {
if (argc != 2) {
    fprintf(stderr, "Usage: %s <pid>\n", argv[0]);
    return 1;
}

pid_t pid = atoi(argv[1]);
if (pid <= 0) {
    fprintf(stderr, "Invalid PID\n");
    return 1;
}

print_process_info(pid);
return 0;
}
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$
```

شکل ۱۶: ادامه کد برنامه دریافت شماره پردازه و خروجی اطلاعات مربوط به پردازه

```
File Actions Edit View Help
arshiyos@arshia-oslabbox: ~/syscalls/proc_codes ×

arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ ls
listprocesses.sh process_info.c
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ gcc process_info.c -o process_info
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ sudo ./process_info 1
Executable: /usr/lib/systemd/systemd
Memory Usage:
  Resident Set Size: 14016512 bytes
  Virtual Memory Size: 23371776 bytes
Command Line Parameters:
  /sbin/init
  splash
Environment Variables:
  HOME=/
  init=/sbin/init
  NETWORK_SKIP_ENSLAVED=
  TERM=linux
  BOOT_IMAGE=/boot/vmlinuz-6.14.0-24-generic
  drop_caps=
  PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
  PWD=/
  rootmnt=/root
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$
```

شکل ۱۷: کامپایل و اجرای یک نمونه برای برنامه شکل ۱۵

۴ مشاهده اطلاعات مربوط به هسته

بار دیگر به به مسیر `/proc`/ می‌رویم و محتوای آن را نگاه می‌کنیم، همچنین مسیرهای داخلی گفته شده را هم بررسی می‌کنیم تا محتوای آن‌ها را ببینیم. این موارد در شکل‌های ۱۸، ۱۹، ۲۰، ۲۱، ۲۲، ۲۳، و به تفکیک آمده است.

`/proc/meminfo`

اطلاعات مربوط به حافظه (RAM) سیستم، مانند مقدار کل، آزاد، و کش شده.

`/proc/version`

نسخه هسته (Kernel) و اطلاعات مربوط به سیستم عامل.

`/proc/uptime`

مدت زمانی که سیستم روشن بوده و زمان بیکار بودن CPU.

`/proc/stat`

آمارهای کلی پردازنده و فعالیت سیستم.

`/proc/mounts`

لیست فایل سیستم‌های mount شده فعلی.

`/proc/net/`

اطلاعات شبکه مانند اتصالات و آمار رابطهای شبکه.

`/proc/loadavg`

میانگین بار سیستم در بازه‌های زمانی مختلف.

`/proc/interrupts`

لیست وقفه‌های سخت‌افزاری و شمارش فراخوانی‌ها.

`/proc/ioports`

آدرس‌های ورودی/خروجی استفاده شده توسط سخت‌افزار.

`/proc/filesystems`

فایل سیستم‌های پشتیانی شده توسط هسته.

`/proc/cpuinfo`

اطلاعات دقیق درباره پردازنده‌ها (تعداد، مدل، سرعت و غیره).

`/proc/cmdline`

پارامترهای استفاده شده برای راهاندازی هسته هنگام بوت.

```

arshiyos@arshia-oslabbox:~$ cd /proc
arshiyos@arshia-oslabbox:/proc$ ls
1 1160 1707 31 41 47 55 67 887 97      iomem      partitions
1028 1166 1712 3136 42 4724 56 68 888 99    ioports    pressure
1055 1172 18 3138 4208 48 57 693 889 acpi    irq       schedstat
1059 1178 19 3141 4210 487 58 694 890 asound   kallsyms scsi
1061 1184 2 3142 4214 49 59 7 891 bootconfig  kcore     self
1063 12 20 32 4215 490 593 70 892 buddyinfo  keys      slabinfo
1070 1210 2039 3272 4217 4932 6 71 896 bus     key-users softirqs
1071 1212 21 3273 4247 5 60 713 9 cgroups  kmsg      stat
1076 1224 219 33 4252 50 602 72 90 cmdline  kpagemgroup swaps
1078 1235 22 332 4264 5027 605 73 91 consoles  kpagemcount sys
1079 1257 220 3399 4276 5051 607 74 92 cpufreq  kpagemflags sysrq-trigger
1080 1260 23 35 4277 5055 608 77 93 crypto   latency_stats sysvipc
1081 13 24 362 43 5059 61 78 930 devices  loadavg  thread-self
1082 14 25 3777 4300 5060 62 79 934 diskstats locks   timer_list
1084 15 26 39 4344 5069 629 8 935 dma     mdstat  tty
1088 154 269 3933 4361 5072 634 858 943 driver  meminfo uptime
109 16 27 4 4362 51 636 861 944 dynamic_debug misc   version
1098 161 29 40 45 52 637 875 945 execdomains modules version_signature
1125 162 3 4046 4607 526 642 88 946 fb     mounts  vmallocinfo
1133 163 30 4068 4608 527 65 880 947 filesystems mtrr   vmstat
1139 1665 3007 4072 4651 53 651 884 956 fs     net    zoneinfo
1140 17 3014 4079 4660 54 66 886 967 interrupts pagetypeinfo

arshiyos@arshia-oslabbox:/proc$ cat cpuinfo
processor : 0
vendor_id : GenuineIntel
cpu family : 6
model : 142
model name : Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
stepping : 9
cpu MHz : 2903.988
cache size : 4096 KB
physical id : 0

```

شکل ۱۸: چاپ محتوای مسیر /proc و /proc/cpuinfo

```

arshiyos@arshia-oslabbox:~$ cd /proc
arshiyos@arshia-oslabbox:/proc$ cat cmdline
BOOT_IMAGE=/boot/vmlinuz-6.14.0-24-generic root=UUID=8cf9cd8c-4574-467f-a69e-19a0f51ec0ca ro quiet splash
arshiyos@arshia-oslabbox:/proc$ cat ioports
0000-0000 : PCI Bus 0000:00
    0000-0000 : dma1
    0000-0000 : pic1
    0000-0000 : timer0
    0000-0000 : timer1
    0000-0000 : keyboard
    0000-0000 : keyboard
    0000-0000 : rtc_cmos
    0000-0000 : rtc0
    0000-0000 : dma page reg
    0000-0000 : pic2
    0000-0000 : dma2
    0000-0000 : fpu
    0000-0000 : 0000:00:01.1
        0000-0000 : ata_piix
        0000-0000 : 0000:00:01.1
            0000-0000 : ata_piix
            0000-0000 : 0000:00:01.1
            0000-0000 : ata_piix
            0000-0000 : vga+
            0000-0000 : 0000:00:01.1
                0000-0000 : ata_piix
            0000-0000 : PCI conf1
        0000-0000 : PCI Bus 0000:00
        0000-0000 : 0000:00:07.0
            0000-0000 : ACPI PM1a_EVT_BLK
            0000-0000 : ACPI PM1a_CNT_BLK
            0000-0000 : ACPI PM_TMR
            0000-0000 : ACPI GPE0_BLK
        0000-0000 : 0000:00:07.0

arshiyos@arshia-oslabbox...

```

شکل ۱۹: چاپ محتوای /proc/cmdline و /proc/ioports

arshiyos@arshia-oslabbox: /proc

```
arshiyos@arshia-oslabbox:/proc$ cat meminfo
MemTotal:           8132848 kB
MemFree:            4787148 kB
MemAvailable:       7301584 kB
Buffers:             82156 kB
Cached:              2620412 kB
SwapCached:          0 kB
Active:              1368992 kB
Inactive:            1691420 kB
Active(anon):        367376 kB
Inactive(anon):      0 kB
Active(file):        1001616 kB
Inactive(file):      1691420 kB
Unevictable:          0 kB
Mlocked:              0 kB
SwapTotal:            524284 kB
SwapFree:             524284 kB
Zswap:                  0 kB
Zswapped:              0 kB
Dirty:                  136 kB
Writeback:                 0 kB
AnonPages:            357944 kB
Mapped:               269028 kB
Shmem:                  9532 kB
KReclaimable:         80444 kB
Slab:                  198696 kB
SReclaimable:         80444 kB
SUnreclaim:            118252 kB
KernelStack:            5728 kB
PageTables:            14248 kB
SecPageTables:          0 kB
NFS_Unstable:           0 kB
Bounce:                  0 kB
```

شکل ۲۱: چاپ محتوای /proc/uptime و /proc/loadavg و /proc/version و /proc/stat

File Machine View Input Devices Help

arshiyos@arshia-oslabbox: /proc \$ cat interrupts

	CPU0	CPU1	CPU2	
0:	225	0	0	IO-APIC 2-edge timer
1:	0	10512	0	IO-APIC 1-edge i8042
8:	0	0	0	IO-APIC 8-edge rtc0
9:	0	0	0	IO-APIC 9-fasteoi acpi
12:	3346	0	0	IO-APIC 12-edge i8042
14:	0	0	0	IO-APIC 14-edge ata_piix
15:	3715	0	0	IO-APIC 15-edge ata_piix
18:	0	0	1	IO-APIC 18-fasteoi vmmgfx
19:	0	0	235327	IO-APIC 19-fasteoi ehci_hcd:usb2, enp0s3
20:	0	11040	0	IO-APIC 20-fasteoi vboxguest
21:	34664	0	0	IO-APIC 21-fasteoi ahci[0000:00:0d.0], snd_intel8x0
22:	0	28	0	IO-APIC 22-fasteoi ohci_hcd:usb1
NMI:	0	0	0	Non-maskable interrupts
LOC:	1336838	790457	706342	Local timer interrupts
SPU:	0	0	0	Spurious interrupts
PMI:	0	0	0	Performance monitoring interrupts
IWI:	0	0	0	IRQ work interrupts
RTR:	0	0	0	APIC ICR read retries
RES:	10459	13697	13743	Rescheduling interrupts
CAL:	175991	277861	218793	Function call interrupts
TLB:	14267	17668	16783	TLB shootdowns
TRM:	0	0	0	Thermal event interrupts
THR:	0	0	0	Threshold APIC interrupts
DFR:	0	0	0	Deferred Error APIC interrupts
MCE:	0	0	0	Machine check exceptions
MCP:	11	11	11	Machine check polls
ERR:	0			
MIS:	0			
PIN:	0	0	0	Posted-interrupt notification event
NPI:	0	0	0	Nested posted-interrupt event
PIW:	0	0	0	Posted-interrupt wakeup event

شکل ۲۲: چاپ محتوای /proc/interrupts

The screenshot shows a terminal window titled "arshia.yousefnia.oslab.vm [Running] - Oracle VM VirtualBox". The terminal is running on a Lubuntu desktop environment. The command "ls" is run in the directory "/proc/net\$". The output lists various network-related files and directories such as fib_triestat, ip6_mr_cache, ipv6_route, protocols, rt_acct, stat, unix, wireless, netfilter, raw, snmp, tcp6, xfrm_stat, netlink, sockstat, udp6, ip_tables_matches, ip_tables_names, netsstat, route, sockstat6, udplite, ip6_flowlabel, ip_tables_targets, packet, rt6_stats, softnet_stat, udplite6, and a list of TCP connections. The command "cat tcp" is run, showing detailed information for each connection. The output for the first few connections is as follows:

```

arshiyos@arshia-oslabbox:/proc/net$ ls
anycast6  fib_triestat  ip6_mr_cache  ipv6_route  protocols  rt_acct      stat      unix
arp       icmp        ip6_mr_vif    mcfILTER   psched     rt_cache    tcp       wireless
connector  icmp6      ip_mr_cache   mcfILTER6  ptype      snmp      tcp6      xfrm_stat
dev       if_inet6    ip_mr_vif    netfilter   raw       sockstat   udp6
dev_mcast  igmp      ip_tables_matches  netlink   raw6      sockstat6  udplite
dev_smp6   igmp6     ip_tables_names   netsstat  route     sockstat6  udplite6
fib_trie  ip6_flowlabel  ip_tables_targets  packet   rt6_stats softnet_stat udplite6
arshiyos@arshia-oslabbox:/proc/net$ cat tcp
sl local_address rem_address      st tx_queue rx_queue tr tm->when retrnsmt  uid  timeout inode
0: 0100007F:0277 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 10473 1 000000000000
00000000 100 0 0 10 0
1: 3500007F:0035 00000000:0000 0A 00000000:00000000 00:00000000 00000000 991 0 7654 1 000000000000
00000000 100 0 0 10 5
2: 3600007F:0035 00000000:0000 0A 00000000:00000000 00:00000000 00000000 991 0 7656 1 000000000000
00000000 100 0 0 10 5
arshiyos@arshia-oslabbox:/proc/net$ cd ..
arshiyos@arshia-oslabbox:/proc$ cat filesystems
nodev sysfs
nodev tmpfs
nodev bdev
nodev proc
nodev cgroup
nodev cgroup2
nodev devtmpfs
nodev configfs
nodev debugfs
nodev tracefs
nodev securityfs
nodev sockfs
nodev bpf
nodev pipes

```

شکل ۲۳: چاپ محتوای /proc/filesystems و /proc/net/

در ادامه یک برنامه می‌نویسیم که نام مدل پردازنده، فرکانس آن و مقدار حافظه نهان آن را چاپ کنی. شکل ۲۴ این کد را نشان می‌دهد و شکل ۲۵ هم کامپایل و اجرای آن را نشان می‌دهد.

```

GNU nano 7.2
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define MAX_LINE 256

void print_cpu_info() {
    FILE *cpuinfo = fopen("/proc/cpuinfo", "r");
    if (!cpuinfo) {
        perror("Error opening /proc/cpuinfo");
        exit(1);
    }

    char line[MAX_LINE];
    char model_name[MAX_LINE] = "Unknown";
    char cpu_mhz[MAX_LINE] = "Unknown";
    char cache_size[MAX_LINE] = "Unknown";

    while (fgets(line, sizeof(line), cpuinfo)) {
        if (strstr(line, "model name")) {
            sscanf(line, "%*[^\n]: %[^\\n]", model_name);
        } else if (strstr(line, "cpu MHz")) {
            sscanf(line, "%*[^\n]: %s", cpu_mhz);
        } else if (strstr(line, "cache size")) {
            sscanf(line, "%*[^\n]: %[^\\n]", cache_size);
        }
    }

    printf("processor model: %s\n", model_name);
    printf("frequency: %s MHz\n", cpu_mhz);
    printf("cache mem: %s\n", cache_size);

    fclose(cpuinfo);
}

int main() {
    print_cpu_info();
    return 0;
}

```

شکل ۲۴: برنامه چاپ مدل و مشخصات پردازنده

```
File Actions Edit View Help
arshiyos@arshia-oslabbox:~/syscalls/proc_codes ×
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ ls
cpu_info.c listprocesses.sh process_info process_info.c
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ gcc cpu_info.c -o cpu_info
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ ./cpu_info
processor model: Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
frequency: 2903.988 MHz
cache mem: 4096 KB
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$
```

شکل ۲۵: کامپایل و اجرای برنامه شکل ۲۴

در قسمت بعد یک برنامه دیگر می‌نویسیم که مقدار حافظه کل، حافظه استفاده شده و حافظه آزاد را در خروجی چاپ کند. شکل ۲۶ این کد را نشان می‌دهد و شکل ۲۷ هم کامپایل و اجرای آن را نشان می‌دهد.

```
File Actions Edit View Help
arshilayos@arshia-oslabox: ~/syscalls/proc_codes ×

GNU nano 7.2
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define MAX_LINE 256

void print_memory_info() {
    FILE *meminfo = fopen("/proc/meminfo", "r");
    if (!meminfo) {
        perror("Error opening /proc/meminfo");
        exit(1);
    }

    char line[MAX_LINE];
    unsigned long mem_total = 0, mem_free = 0, mem_available = 0;

    while (fgets(line, sizeof(line), meminfo)) {
        if (strstr(line, "MemTotal")) {
            sscanf(line, "%*s %lu", &mem_total);
        } else if (strstr(line, "MemFree")) {
            sscanf(line, "%*s %lu", &mem_free);
        } else if (strstr(line, "MemAvailable")) {
            sscanf(line, "%*s %lu", &mem_available);
        }
    }

    printf("total mem: %lu KB\n", mem_total);
    printf("free mem: %lu KB\n", mem_free);
    printf("available mem: %lu KB\n", mem_available);
    printf("used mem: %lu KB\n", mem_total - mem_available);

    fclose(meminfo);
}

int main() {
    print_memory_info();
    return 0;
}
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T
^X Exit ^R Read File ^\ Replace ^U Paste ^D

شكل ٢٦ : برنامه سلیپ وضعیت حافظه

```

File Actions Edit View Help
arshiyos@arshia-oslabbox: ~/syscalls/proc_codes ×
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ ls
cpu_info cpu_info.c listprocesses.sh memory_info.c process_info process_info.c
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ gcc memory_info.c -o memory_info
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ ./memory_info
total mem: 8132848 KB
free mem: 4552780 KB
available mem: 7069196 KB
used mem: 1063652 KB
arshiyos@arshia-oslabbox:~/syscalls/proc_codes$ █

```

شکل ۲۷: کامپایل و اجرای برنامه شکل ۲۶

۱.۴ تمرین

پنج فایل مهم در /proc/sys/kernel/ و کاربرد آن‌ها:

hostname

کاربرد: نام میزبان (Hostname) سیستم را نگه می‌دارد یا نمایش می‌دهد. این نام شناسه‌ی سیستم در شبکه است.
مثال: تغییر محتوی این فایل، نام میزبان را تغییر می‌دهد.

osrelease

کاربرد: نسخه کرنل لینوکس که در حال اجراست را نشان می‌دهد.
مثال: بررسی ورژن کرنل برای عیب‌یابی یا مدیریت سیستم.

panic

کاربرد: مشخص می‌کند سیستم پس از مواجهه با خطا بحرانی (kernel panic) چه مدت صبر کند تا ریبوت شود. مقدار عددی این فایل نشان‌دهنده تعداد ثانیه‌ها است.
مثال: تنظیم این مقدار برای خودکار ریبوت کردن سیستم پس از کرش.

printk

کاربرد: سطح اولویت پیام‌های کرنل را تعیین می‌کند که چه پیام‌هایی باید در لایگ نمایش داده شوند.
مثال: برای کنترل میزان اطلاعات گزارش شده توسط کرنل، مثلًا برای دیباگ.

version

کاربرد: اطلاعات کامل‌تری درباره نسخه کرنل، شامل شماره نسخه، تاریخ کامپایل و اطلاعات بیشتر.
مثال: استفاده برای تشخیص دقیق نسخه کرنل و اطلاعات مربوط به کامپایل.
منع: [۱]

/proc/self

در مورد /proc/self و کاربرد آن: /proc/self یک لینک نمادین (symbolic link) است که به شاخه‌ی مربوط به فرآیند (process) ای که در حال دسترسی به این مسیر است اشاره می‌کند.

یعنی اگر فرآیندی /proc/self را باز کند، در واقع به [proc/[PID]] خودش اشاره دارد، یعنی شناسه فرآیند خود را نشان می‌دهد.

کاربرد: این مسیر به برنامه‌ها و اسکریپت‌ها اجازه می‌دهد بدون نیاز به دانستن PID خود، اطلاعات مرتبط با فرآیند خود را بخوانند یا دستکاری کنند.

مثال:

/proc/self/status اطلاعات وضعیت فرآیند جاری را نشان می‌دهد.
/proc/self/fd لیست توصیفگرهای فایل (file descriptors) باز فرآیند جاری است.

منبع: [۲]

مراجع

- [١] URL: <https://www.kernel.org/doc/html/latest/admin-guide/sysctl/kernel.html#kernel-parameters>.
- [٢] URL: <https://unix.stackexchange.com/questions/333225/which-process-is-proc-self-for>.