How to add system call your the linux OS Kernel

Turtles Team

Team Names:

- Omnia Hamada Husien.
- Shahd Emad Hamdy .

CPU Cores:

➔ A CPU core or (processor CPU) is an individual processor within a CPU. In the old days, every processor had just one core that could focus on one task at a time. Today, CPUs have been two and 18 cores, each of which can work on a different task. As you can see in our CPU Benchmarks Hierarchy, that can have a huge impact on performance.

RAM Capacity:

➔ Memory capacity is the amount of memory that can be used for an electronic device such as a computer, laptop, smartphone or other smart device. Every hardware device or computer has a minimum and maximum amount of memory. The performance of a device and the efficiency of its input/output operations is dependent on memory capacity.

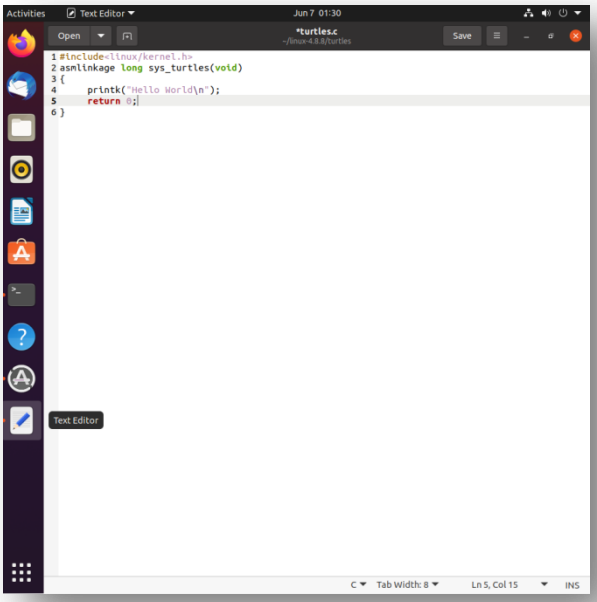
Kernel Version:

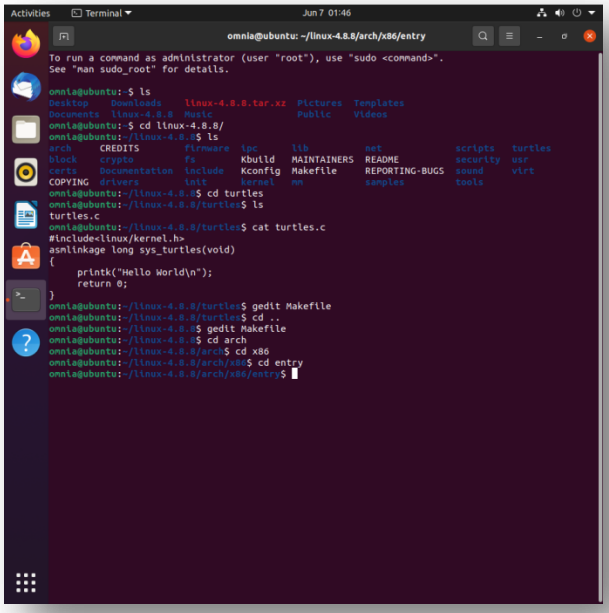
➔ The kernel is the core program of your operating system. It starts just right after the bootloader, and it manages all the available hardware resources, providing an abstraction layer for what is known as application programs. The kernel serves as the bridge between your computer hardware and the software you wish to run. It talks to the hardware via the drivers that are included in the kernel.

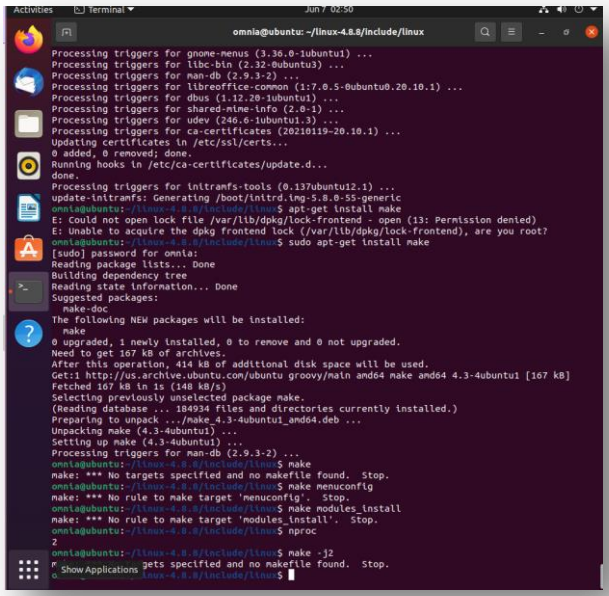
steps of how adding the system call in linux kernel:

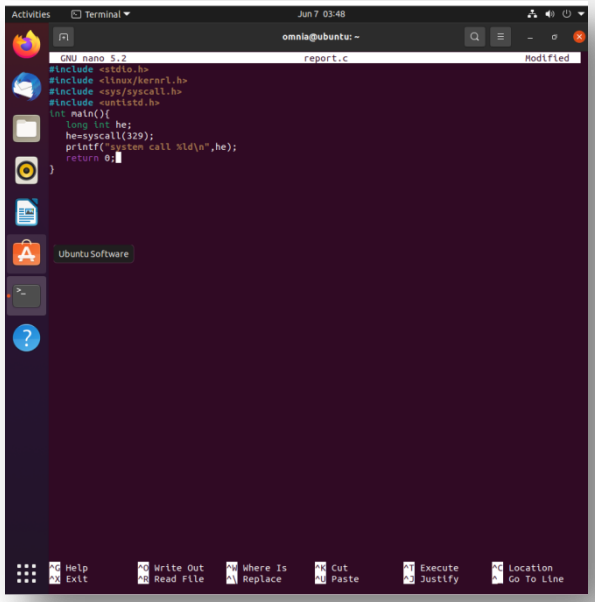
1. Download the source code of the latest stable version of the Linux kernel to your home folder.
2. Unpack the tarball you just downloaded to your home folder.
3. Change your working directory to the root directory of the recently unpacked source code.
4. Create the home directory of your system call.
5. Create a C file for your system call.
6. Create a Makefile for your system call.
7. Add the home directory of your system call to the main Makefile of the kernel.

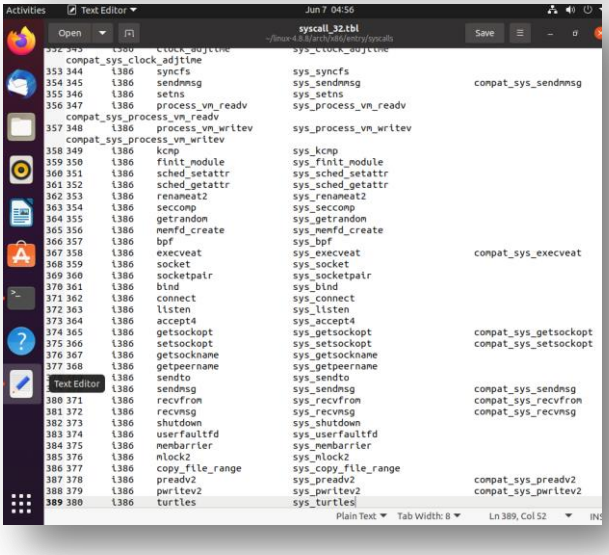
   - Search for core-y. In the second result, you will see a series of directories.kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/
8. Add a corresponding function prototype for your system call to the header file of system calls.
9. Add your system call to the kernel's system call table.
10. install the new kernel and prepare your operating system to boot into it.
11. Configure the kernel.
12. Find out how many logical cores you have.
13. Compile the kernel's source code.
14. Prepare the installer of the kernel.
15. Install the kernel.
16. Update the bootloader of the operating system with the new kernel.
17. Change your working directory to your home directory.
18. Create a C file to generate a report of the success or failure of your system call.
19. Compile the C file you just created.
20. Run the C file you just compiled.
21. Check the last line of the dmesg output.

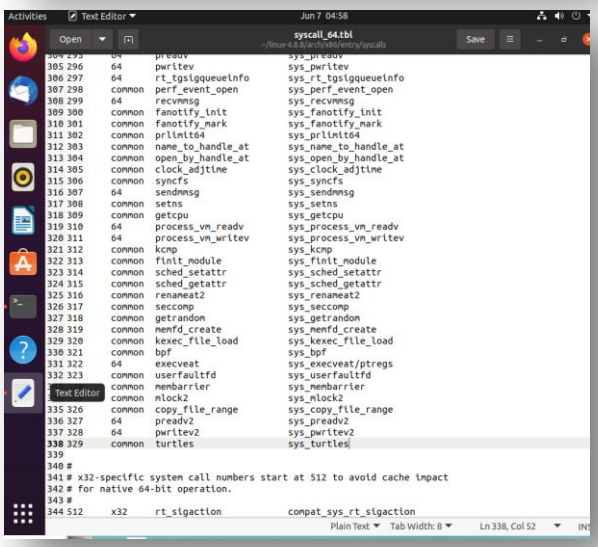

Activities
Text Editor
Jun 7 01:30
Open
*turtles.c
~/linux-4.8.8/turtles
Save
1 #include<linux/kernel.h>
2 asmlinkage long sys_turtles(void)
3 {
4 printk("Hello World\n");
5 return 0;
6 }
Text Editor
C
Tab Width: 8
Ln 5, Col 15
INS

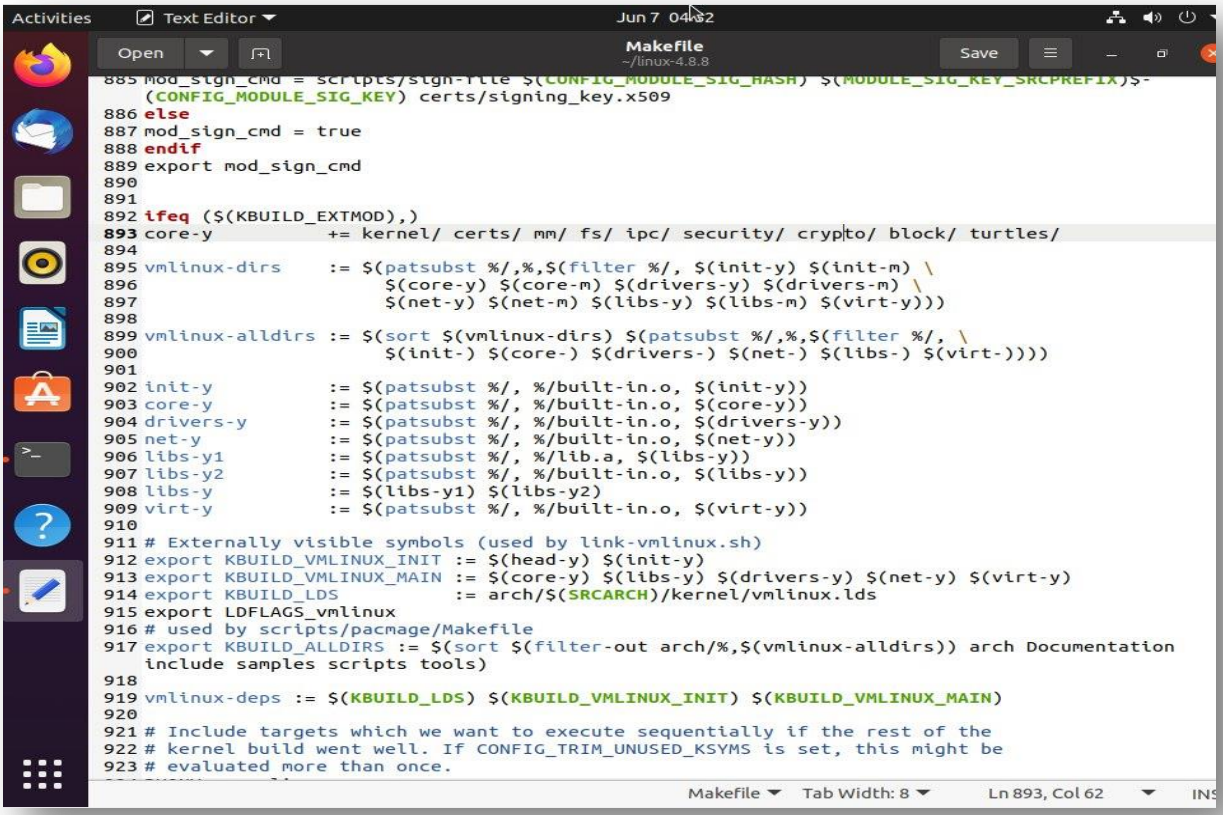

Activities
Terminal
Jun 7 01:46
omnia@ubuntu: ~/linux-4.8.8/arch/x86/entry
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
omnia@ubuntu:~$ ls
Desktop Downloads linux-4.8.8.tar.xz Pictures Templates
Documents linux-4.8.8 Music Public Videos
omnia@ubuntu:~$ cd linux-4.8.8/
omnia@ubuntu:~/linux-4.8.8$ ls
arch CREDITS firmware ipc lib net scripts turtles
block crypto fs Kbuild MAINTAINERS README security usr
certs Documentation include Kconfig Makefile REPORTING-BUGS sound virt
COPYING drivers init kernel mm samples tools
omnia@ubuntu:~/linux-4.8.8$ cd turtles
omnia@ubuntu:~/linux-4.8.8/turtles$ ls
turtles.c
omnia@ubuntu:~/linux-4.8.8/turtles$ cat turtles.c
#include<linux/kernel.h>
asmlinkage long sys_turtles(void)
{
printk("Hello World\n");
return 0;
}
omnia@ubuntu:~/linux-4.8.8/turtles$ gedit Makefile
omnia@ubuntu:~/linux-4.8.8/turtles$ cd ..
omnia@ubuntu:~/linux-4.8.8$ gedit Makefile
omnia@ubuntu:~/linux-4.8.8$ cd arch
omnia@ubuntu:~/linux-4.8.8/arch$ cd x86
omnia@ubuntu:~/linux-4.8.8/arch/x86$ cd entry
omnia@ubuntu:~/linux-4.8.8/arch/x86/entry$

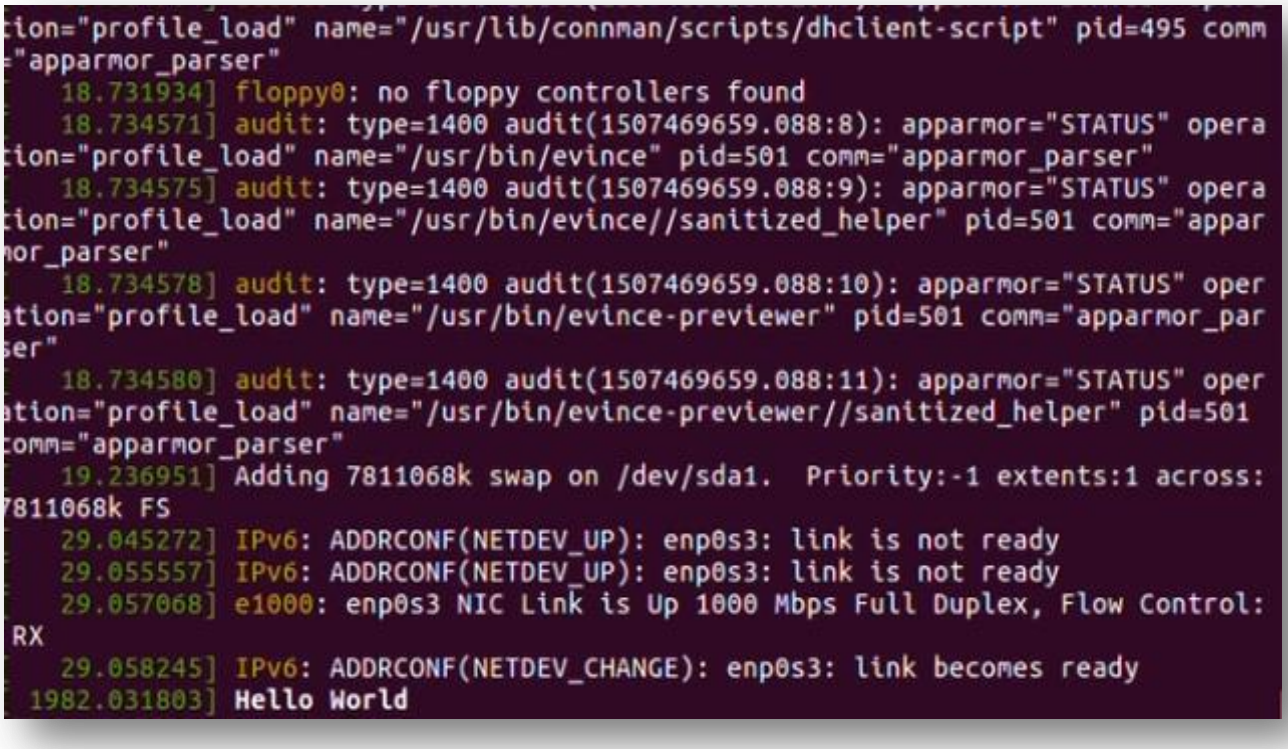

Activities
Terminal
Jun 7 02:50
omnia@ubuntu: ~/linux-4.8.8/include/linux
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for libc-bin (2.32-0ubuntu3) ...
Processing triggers for man-db (2.9.3-2) ...
Processing triggers for libreoffice-common (1:7.0.5-0ubuntu0.20.10.1) ...
Processing triggers for dbus (1.12.20-1ubuntu1) ...
Processing triggers for shared-mime-info (2.0-1) ...
Processing triggers for udev (246.6-1ubuntu1.3) ...
Processing triggers for ca-certificates (20210119~20.10.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
Processing triggers for initramfs-tools (0.137ubuntu12.1) ...
update-initramfs: Generating /boot/initrd.img-5.8.0-55-generic
omnia@ubuntu:~/linux-4.8.8/include/linux$ apt-get install make
E: Could not open lock file /var/lib/dpkg/lock-frontend - open (13: Permission denied)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontend), are you root?
omnia@ubuntu:~/linux-4.8.8/include/linux$ sudo apt-get install make
[sudo] password for omnia:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
make-doc
The following NEW packages will be installed:
make
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 167 kB of archives.
After this operation, 414 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu groovy/main amd64 make amd64 4.3-4ubuntu1 [167 kB]
Fetched 167 kB in 1s (148 kB/s)
Selecting previously unselected package make.
(Reading database ... 184934 files and directories currently installed.)
Preparing to unpack .../make_4.3-4ubuntu1_amd64.deb ...
Unpacking make (4.3-4ubuntu1) ...
Setting up make (4.3-4ubuntu1) ...
Processing triggers for man-db (2.9.3-2) ...
omnia@ubuntu:~/linux-4.8.8/include/linux$ make
make: *** No targets specified and no makefile found. Stop.
omnia@ubuntu:~/linux-4.8.8/include/linux$ make menuconfig
make: *** No rule to make target 'menuconfig'. Stop.
omnia@ubuntu:~/linux-4.8.8/include/linux$ make modules_install
make: *** No rule to make target 'modules_install'. Stop.
omnia@ubuntu:~/linux-4.8.8/include/linux$ nproc
2
omnia@ubuntu:~/linux-4.8.8/include/linux$ make -j2
make: *** No targets specified and no makefile found. Stop.
Show Applications


Activities
Terminal
Jun 7 03:48
omnia@ubuntu: ~
GNU nano 5.2
report.c
Modified
#include <stdio.h>
#include <linux/kernrl.h>
#include <sys/syscall.h>
#include <untistd.h>
int main(){
long int he;
he=syscall(329);
printf("system call %ld\n",he);
return 0;
}
Ubuntu Software
^G Help
^X Exit
^O Write Out
^R Read File
^W Where Is
^\ Replace
^K Cut
^U Paste
^T Execute
^J Justify
^C Location
^_ Go To Line


Activities
Text Editor
Jun 7 04:56
Open
syscall_32.tbl
~/linux-4.8.8/arch/x86/entry/syscalls
Save
compat_sys_clock_adjtime
353 344 i386 syncfs sys_syncfs
354 345 i386 sendmmsg sys_sendmmsg compat_sys_sendmmsg
355 346 i386 setns sys_setns
356 347 i386 process_vm_readv sys_process_vm_readv
compat_sys_process_vm_readv
357 348 i386 process_vm_writev sys_process_vm_writev
compat_sys_process_vm_writev
358 349 i386 kcmp sys_kcmp
359 350 i386 finit_module sys_finit_module
360 351 i386 sched_setattr sys_sched_setattr
361 352 i386 sched_getattr sys_sched_getattr
362 353 i386 renameat2 sys_renameat2
363 354 i386 seccomp sys_seccomp
364 355 i386 getrandom sys_getrandom
365 356 i386 memfd_create sys_memfd_create
366 357 i386 bpf sys_bpf
367 358 i386 execveat sys_execveat compat_sys_execveat
368 359 i386 socket sys_socket
369 360 i386 socketpair sys_socketpair
370 361 i386 bind sys_bind
371 362 i386 connect sys_connect
372 363 i386 listen sys_listen
373 364 i386 accept4 sys_accept4
374 365 i386 getsockopt sys_getsockopt compat_sys_getsockopt
375 366 i386 setsockopt sys_setsockopt compat_sys_setsockopt
376 367 i386 getsockname sys_getsockname
377 368 i386 getpeername sys_getpeername
378 369 i386 sendto sys_sendto
379 370 i386 sendmsg sys_sendmsg compat_sys_sendmsg
380 371 i386 recvfrom sys_recvfrom compat_sys_recvfrom
381 372 i386 recvmsg sys_recvmsg compat_sys_recvmsg
382 373 i386 shutdown sys_shutdown
383 374 i386 userfaultfd sys_userfaultfd
384 375 i386 membarrier sys_membarrier
385 376 i386 mlock2 sys_mlock2
386 377 i386 copy_file_range sys_copy_file_range
387 378 i386 preadv2 sys_preadv2 compat_sys_preadv2
388 379 i386 pwritev2 sys_pwritev2 compat_sys_pwritev2
389 380 i386 turtles sys_turtles
Text Editor
Plain Text
Tab Width: 8
Ln 389, Col 52
INS


Activities
Text Editor
Jun 7 04:58
Open
syscall_64.tbl
~/linux-4.8.8/arch/x86/entry/syscalls
Save
305 296 64 pwritev sys_pwritev
306 297 64 rt_tgsigqueueinfo sys_rt_tgsigqueueinfo
307 298 common perf_event_open sys_perf_event_open
308 299 64 recvmmsg sys_recvmmsg
309 300 common fanotify_init sys_fanotify_init
310 301 common fanotify_mark sys_fanotify_mark
311 302 common prlimit64 sys_prlimit64
312 303 common name_to_handle_at sys_name_to_handle_at
313 304 common open_by_handle_at sys_open_by_handle_at
314 305 common clock_adjtime sys_clock_adjtime
315 306 common syncfs sys_syncfs
316 307 64 sendmmsg sys_sendmmsg
317 308 common setns sys_setns
318 309 common getcpu sys_getcpu
319 310 64 process_vm_readv sys_process_vm_readv
320 311 64 process_vm_writev sys_process_vm_writev
321 312 common kcmp sys_kcmp
322 313 common finit_module sys_finit_module
323 314 common sched_setattr sys_sched_setattr
324 315 common sched_getattr sys_sched_getattr
325 316 common renameat2 sys_renameat2
326 317 common seccomp sys_seccomp
327 318 common getrandom sys_getrandom
328 319 common memfd_create sys_memfd_create
329 320 common kexec_file_load sys_kexec_file_load
330 321 common bpf sys_bpf
331 322 64 execveat sys_execveat/ptregs
332 323 common userfaultfd sys_userfaultfd
333 324 common membarrier sys_membarrier
334 325 common mlock2 sys_mlock2
335 326 common copy_file_range sys_copy_file_range
336 327 64 preadv2 sys_preadv2
337 328 64 pwritev2 sys_pwritev2
338 329 common turtles sys_turtles
339
340 #
341 # x32-specific system call numbers start at 512 to avoid cache impact
342 # for native 64-bit operation.
343 #
344 512 x32 rt_sigaction compat_sys_rt_sigaction
Text Editor
Plain Text
Tab Width: 8
Ln 338, Col 52
INS


Activities
Text Editor
Jun 7 04:52
Open
Makefile
~/linux-4.8.8
Save
885 mod_sign_cmd = scripts/sign-file $(CONFIG_MODULE_SIG_HASH) $(MODULE_SIG_KEY_SRCPREFIX)$(CONFIG_MODULE_SIG_KEY) certs/signing_key.x509
886 else
887 mod_sign_cmd = true
888 endif
889 export mod_sign_cmd
890
891
892 ifeq ($(KBUILD_EXTMOD),)
893 core-y += kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/ turtles/
894
895 vmlinux-dirs := $(patsubst %/,%,$(filter %/, $(init-y) $(init-m) \
896 $(core-y) $(core-m) $(drivers-y) $(drivers-m) \
897 $(net-y) $(net-m) $(libs-y) $(libs-m) $(virt-y)))
898
899 vmlinux-alldirs := $(sort $(vmlinux-dirs) $(patsubst %/,%,$(filter %/, \
900 $(init-) $(core-) $(drivers-) $(net-) $(libs-) $(virt-))))
901
902 init-y := $(patsubst %/, %/built-in.o, $(init-y))
903 core-y := $(patsubst %/, %/built-in.o, $(core-y))
904 drivers-y := $(patsubst %/, %/built-in.o, $(drivers-y))
905 net-y := $(patsubst %/, %/built-in.o, $(net-y))
906 libs-y1 := $(patsubst %/, %/lib.a, $(libs-y))
907 libs-y2 := $(patsubst %/, %/built-in.o, $(libs-y))
908 libs-y := $(libs-y1) $(libs-y2)
909 virt-y := $(patsubst %/, %/built-in.o, $(virt-y))
910
911 # Externally visible symbols (used by link-vmlinux.sh)
912 export KBUILD_VMLINUX_INIT := $(head-y) $(init-y)
913 export KBUILD_VMLINUX_MAIN := $(core-y) $(libs-y) $(drivers-y) $(net-y) $(virt-y)
914 export KBUILD_LDS := arch/$(SRCARCH)/kernel/vmlinux.lds
915 export LDFLAGS_vmlinux
916 # used by scripts/pacmage/Makefile
917 export KBUILD_ALLDIRS := $(sort $(filter-out arch/%,$(vmlinux-alldirs)) arch Documentation include samples scripts tools)
918
919 vmlinux-deps := $(KBUILD_LDS) $(KBUILD_VMLINUX_INIT) $(KBUILD_VMLINUX_MAIN)
920
921 # Include targets which we want to execute sequentially if the rest of the
922 # kernel build went well. If CONFIG_TRIM_UNUSED_KSYMS is set, this might be
923 # evaluated more than once.
Makefile
Tab Width: 8
Ln 893, Col 62
INS


tion="profile_load" name="/usr/lib/connman/scripts/dhclient-script" pid=495 comm
="apparmor_parser"
18.731934] floppy0: no floppy controllers found
18.734571] audit: type=1400 audit(1507469659.088:8): apparmor="STATUS" opera
tion="profile_load" name="/usr/bin/evince" pid=501 comm="apparmor_parser"
18.734575] audit: type=1400 audit(1507469659.088:9): apparmor="STATUS" opera
tion="profile_load" name="/usr/bin/evince//sanitized_helper" pid=501 comm="appar
mor_parser"
18.734578] audit: type=1400 audit(1507469659.088:10): apparmor="STATUS" oper
ation="profile_load" name="/usr/bin/evince-previewer" pid=501 comm="apparmor_par
ser"
18.734580] audit: type=1400 audit(1507469659.088:11): apparmor="STATUS" oper
ation="profile_load" name="/usr/bin/evince-previewer//sanitized_helper" pid=501
comm="apparmor_parser"
19.236951] Adding 7811068k swap on /dev/sda1. Priority:-1 extents:1 across:
7811068k FS
29.045272] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
29.055557] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
29.057068] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control:
RX
29.058245] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
1982.031803] Hello World

References:

- How to add system call (syscall) to the kernel, compile and test it?
- https://www.androidcentral.com/android-z-what-kernel
- https://www.rose-hulman.edu/Class/ee/yoder/ece332/Papers/RAM%20Technologies.pdf
- https://arxiv.org/pdf/1110.3535