

REPORT

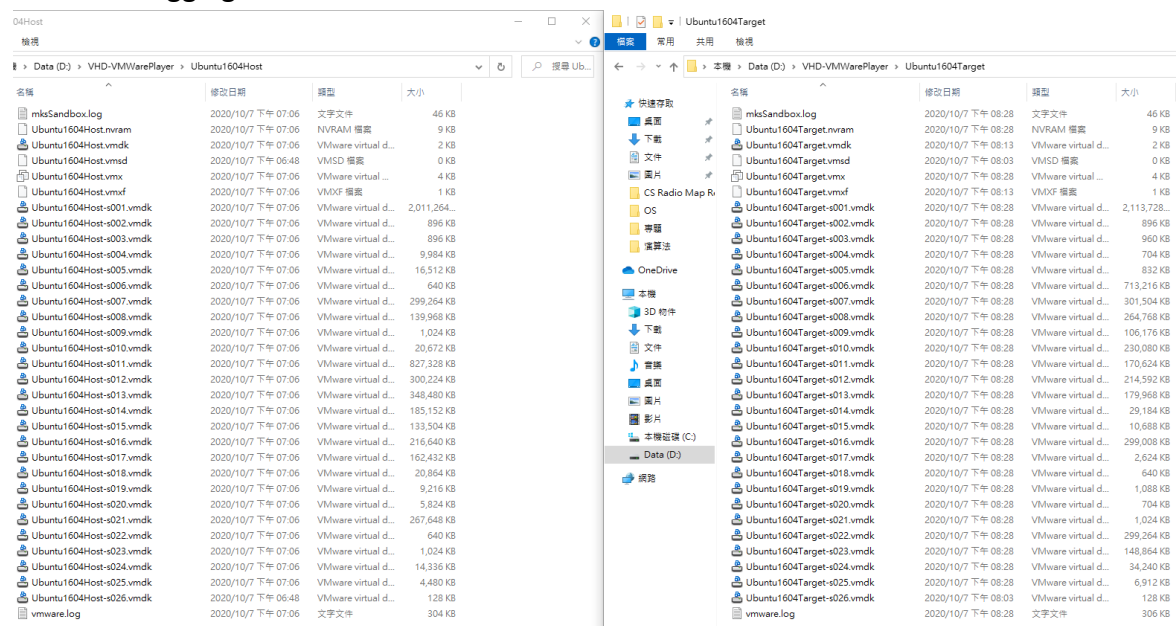
Video: <https://youtu.be/AbDNHbzGXl8>

The “gredit” file I say in 7:17 is actually grub.cfg.

I apologize for the length of my video, but I think it is necessary to explain my problem, sorry for the inconvenience.

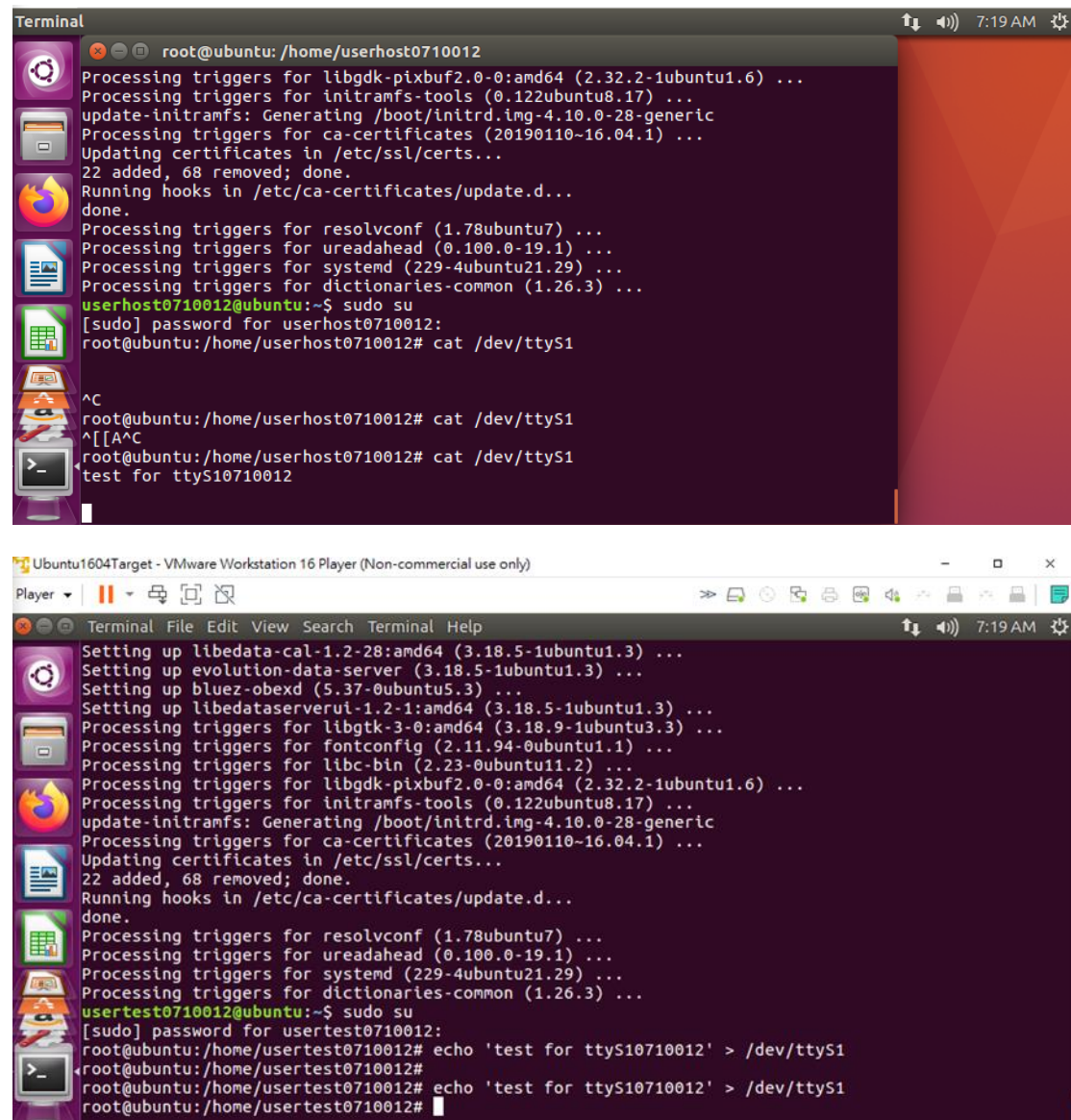
Screenshot 1:

I create two virtual machine named one to be Ubuntu1604Target and make the username as usertest0710012 for the later kernel install, and the other virtual machine as Ubuntu1604Host and make the name as userhost0710012 for later kernel debugging with GDB



Screenshot 2:

Use the host machine to cat a port(/dev/ttyS1), and print(echo) the sentence 'test for ttyS10710012' to check if the port is successfully connected.



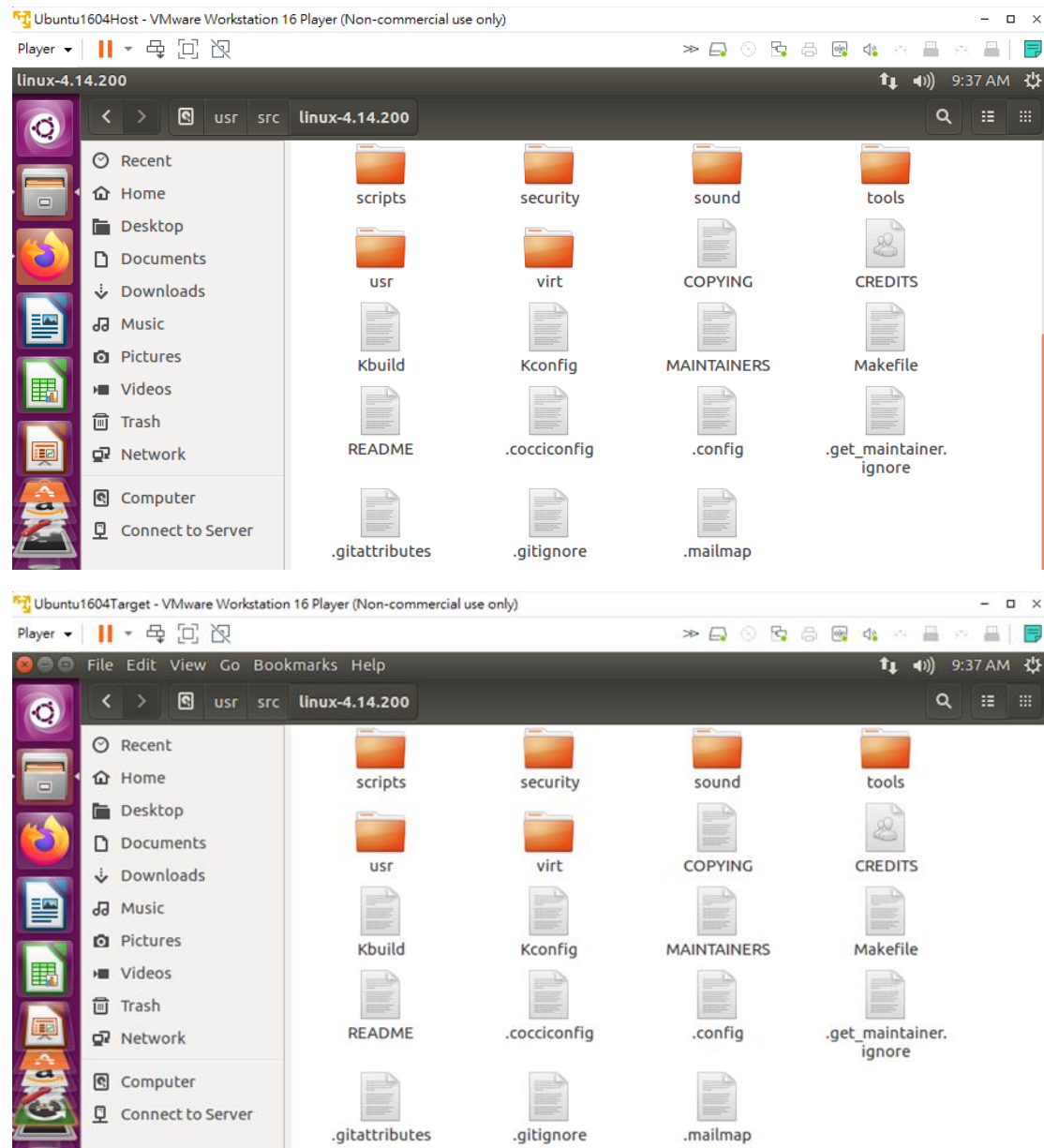
The image consists of two screenshots of a terminal window. The top screenshot shows a terminal session where the user is running system updates. The output includes messages about processing triggers for various packages like libgdk-pixbuf2.0-0, initramfs-tools, ca-certificates, resolvconf, ureadahead, systemd, and dictionaries-common. The user then runs 'sudo su' and 'cat /dev/ttyS1', which returns '^C' and '^[[A^C'. The bottom screenshot shows the same terminal session continuing with more system updates. The user then runs 'echo 'test for ttyS10710012' > /dev/ttyS1' and 'echo 'test for ttyS10710012' > /dev/ttyS1' again, which returns 'test for ttyS10710012'.

```
Terminal
root@ubuntu: /home/userhost0710012
Processing triggers for libgdk-pixbuf2.0-0:amd64 (2.32.2-1ubuntu1.6) ...
Processing triggers for initramfs-tools (0.122ubuntu8.17) ...
update-initramfs: Generating /boot/initrd.img-4.10.0-28-generic
Processing triggers for ca-certificates (20190110-16.04.1) ...
Updating certificates in /etc/ssl/certs...
22 added, 68 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
Processing triggers for resolvconf (1.78ubuntu7) ...
Processing triggers for ureadahead (0.100.0-19.1) ...
Processing triggers for systemd (229-4ubuntu21.29) ...
Processing triggers for dictionaries-common (1.26.3) ...
userhost0710012@ubuntu:~$ sudo su
[sudo] password for userhost0710012:
root@ubuntu: /home/userhost0710012# cat /dev/ttyS1
^C
root@ubuntu: /home/userhost0710012# cat /dev/ttyS1
^[[A^C
root@ubuntu: /home/userhost0710012# cat /dev/ttyS1
test for ttyS10710012

Ubuntu1604Target - VMware Workstation 16 Player (Non-commercial use only)
Player
Terminal File Edit View Search Terminal Help
Setting up libedata-cal-1.2-28:amd64 (3.18.5-1ubuntu1.3) ...
Setting up evolution-data-server (3.18.5-1ubuntu1.3) ...
Setting up bluez-obexd (5.37-0ubuntu5.3) ...
Setting up libedataserverui-1.2-1:amd64 (3.18.5-1ubuntu1.3) ...
Processing triggers for libgtk-3-0:amd64 (3.18.9-1ubuntu3.3) ...
Processing triggers for fontconfig (2.11.94-0ubuntu1.1) ...
Processing triggers for libc-bin (2.23-0ubuntu11.2) ...
Processing triggers for libgdk-pixbuf2.0-0:amd64 (2.32.2-1ubuntu1.6) ...
Processing triggers for initramfs-tools (0.122ubuntu8.17) ...
update-initramfs: Generating /boot/initrd.img-4.10.0-28-generic
Processing triggers for ca-certificates (20190110-16.04.1) ...
Updating certificates in /etc/ssl/certs...
22 added, 68 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
Processing triggers for resolvconf (1.78ubuntu7) ...
Processing triggers for ureadahead (0.100.0-19.1) ...
Processing triggers for systemd (229-4ubuntu21.29) ...
Processing triggers for dictionaries-common (1.26.3) ...
usertest0710012@ubuntu:~$ sudo su
[sudo] password for usertest0710012:
root@ubuntu: /home/usertest0710012# echo 'test for ttyS10710012' > /dev/ttyS1
root@ubuntu: /home/usertest0710012#
root@ubuntu: /home/usertest0710012# echo 'test for ttyS10710012' > /dev/ttyS1
root@ubuntu: /home/usertest0710012#
```

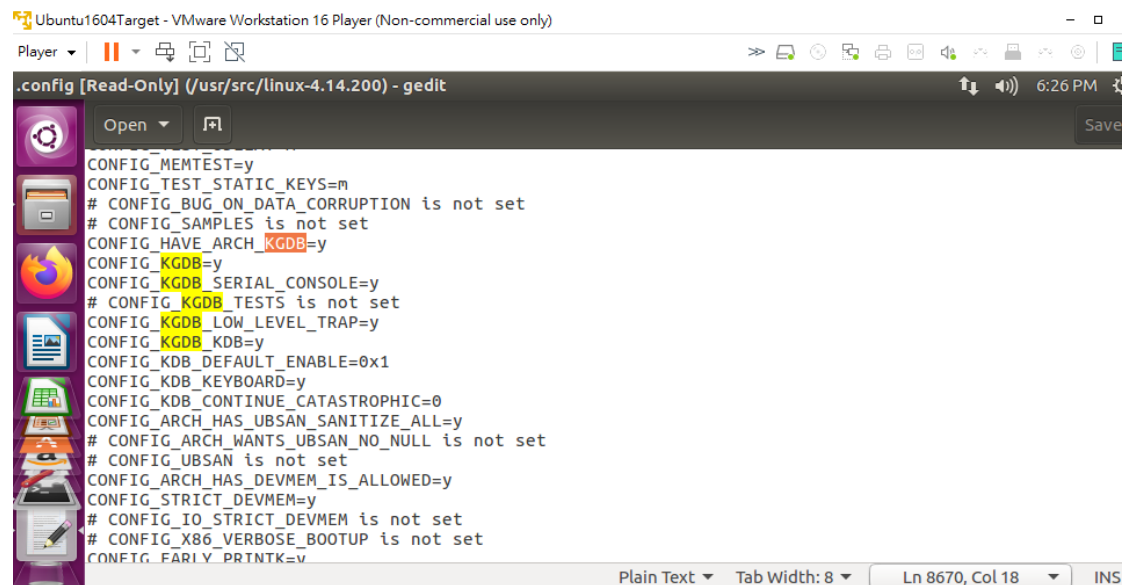
Screenshot 3:

Download the new version of linux kernel and decompress, untar the folder. After that, we pre-build the additional kernel configuration by copying the .config file from the /boot/ folder to the new kernel folder and install the needed software in both target and host machine.



Screenshot 4:

First we find the and check if the command exist and their current status



```
.config [Read-Only] (/usr/src/linux-4.14.200) - gedit
CONFIG_MENTEST=y
CONFIG_TEST_STATIC_KEYS=m
# CONFIG_BUG_ON_DATA_CORRUPTION is not set
# CONFIG_SAMPLES is not set
CONFIG_HAVE_ARCH_KGDB=y
CONFIG_KGDB=y
CONFIG_KGDB_SERIAL_CONSOLE=y
# CONFIG_KGDB_TESTS is not set
CONFIG_KGDB_LOW_LEVEL_TRAP=y
CONFIG_KGDB_KDB=y
CONFIG_KDB_DEFAULT_ENABLE=0x1
CONFIG_KDB_KEYBOARD=y
CONFIG_KDB_CONTINUE_CATASTROPHIC=0
CONFIG_ARCH_HAS_UBSAN_SANITIZE_ALL=y
# CONFIG_ARCH_WANTS_UBSAN_NO_NULL is not set
# CONFIG_UBSAN is not set
CONFIG_ARCH_HAS_DEVMEM_IS_ALLOWED=y
CONFIG_STRICT_DEVMEM=y
# CONFIG_IO_STRICT_DEVMEM is not set
# CONFIG_X86_VERBOSE_BOOTUP is not set
CONFIG_EARLY_PRINTK=y
```

Screenshot 5:

Use nano command to modify the mentioned commands to be in the status they should be to continue the process

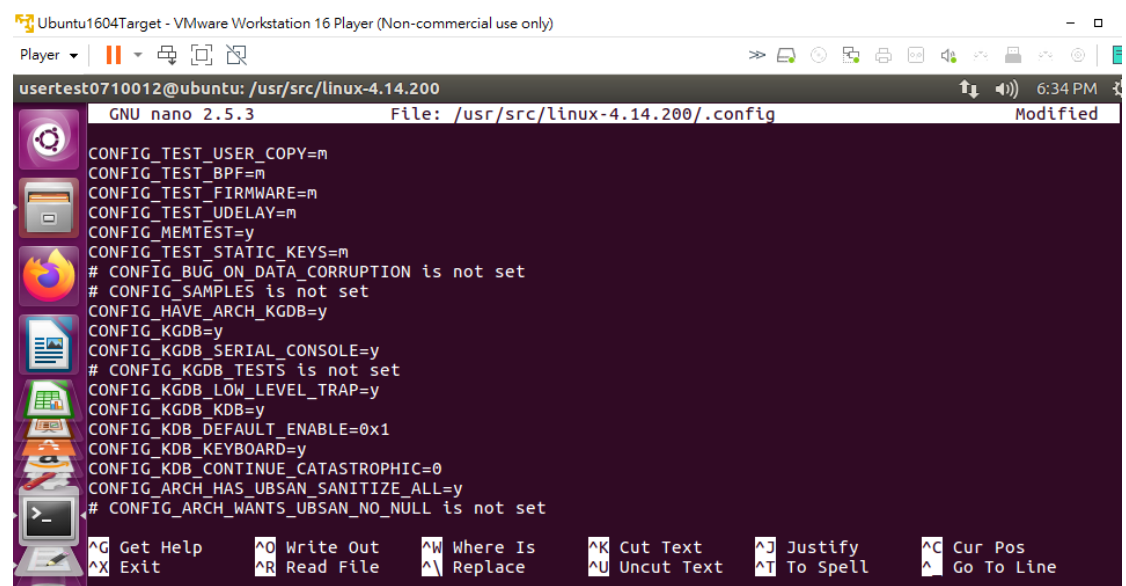
CONFIG_FRAME_POINTER→allow to compile the kernel through frame pointer

CONFIG_KGDB→allow the kernel GNU debugger to work

CONFIG_KGDB_SERIAL_CONSOLE→allow to use the kgdb over the serial console

CONFIG_KGDB_KDB→include the kdb frontend for kgdb

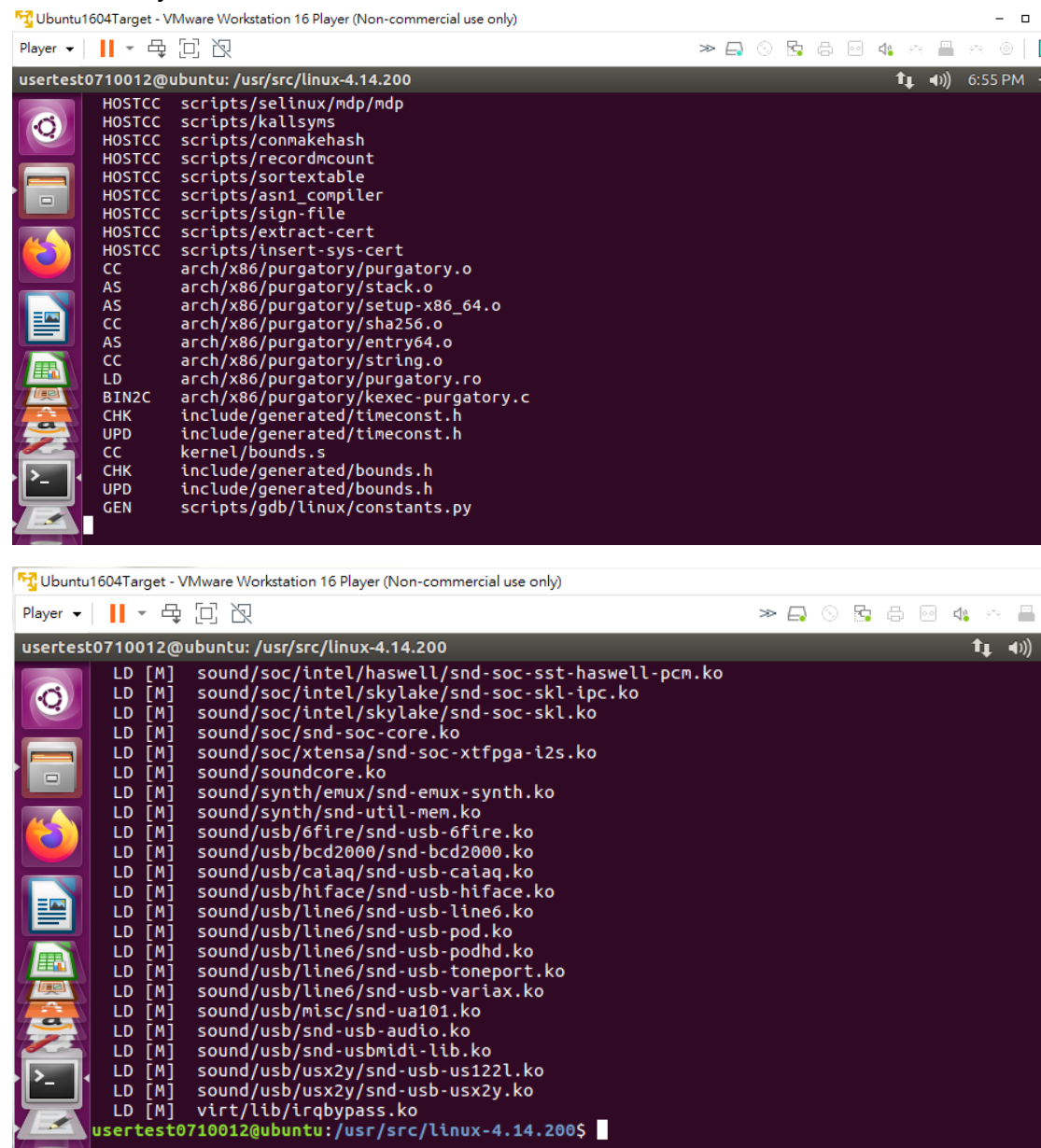
CONFIG_KDB_KEYBOARD→allow keyboard as input device



```
user@ubuntu: /usr/src/linux-4.14.200
GNU nano 2.5.3 File: /usr/src/linux-4.14.200/.config Modified
CONFIG_TEST_USER_COPY=m
CONFIG_TEST_BPF=m
CONFIG_TEST_FIRMWARE=m
CONFIG_TEST_UDELAY=m
CONFIG_MENTEST=y
CONFIG_TEST_STATIC_KEYS=m
# CONFIG_BUG_ON_DATA_CORRUPTION is not set
# CONFIG_SAMPLES is not set
CONFIG_HAVE_ARCH_KGDB=y
CONFIG_KGDB=y
CONFIG_KGDB_SERIAL_CONSOLE=y
# CONFIG_KGDB_TESTS is not set
CONFIG_KGDB_LOW_LEVEL_TRAP=y
CONFIG_KGDB_KDB=y
CONFIG_KDB_DEFAULT_ENABLE=0x1
CONFIG_KDB_KEYBOARD=y
CONFIG_KDB_CONTINUE_CATASTROPHIC=0
CONFIG_ARCH_HAS_UBSAN_SANITIZE_ALL=y
# CONFIG_ARCH_WANTS_UBSAN_NO_NULL is not set
```

Screenshot 6:

Read the text in makefile and compile related files, and run two jobs simultaneously
`sudo make -j 2`



The first screenshot shows the output of the first two jobs of the make command. The second screenshot shows the output of the remaining jobs.

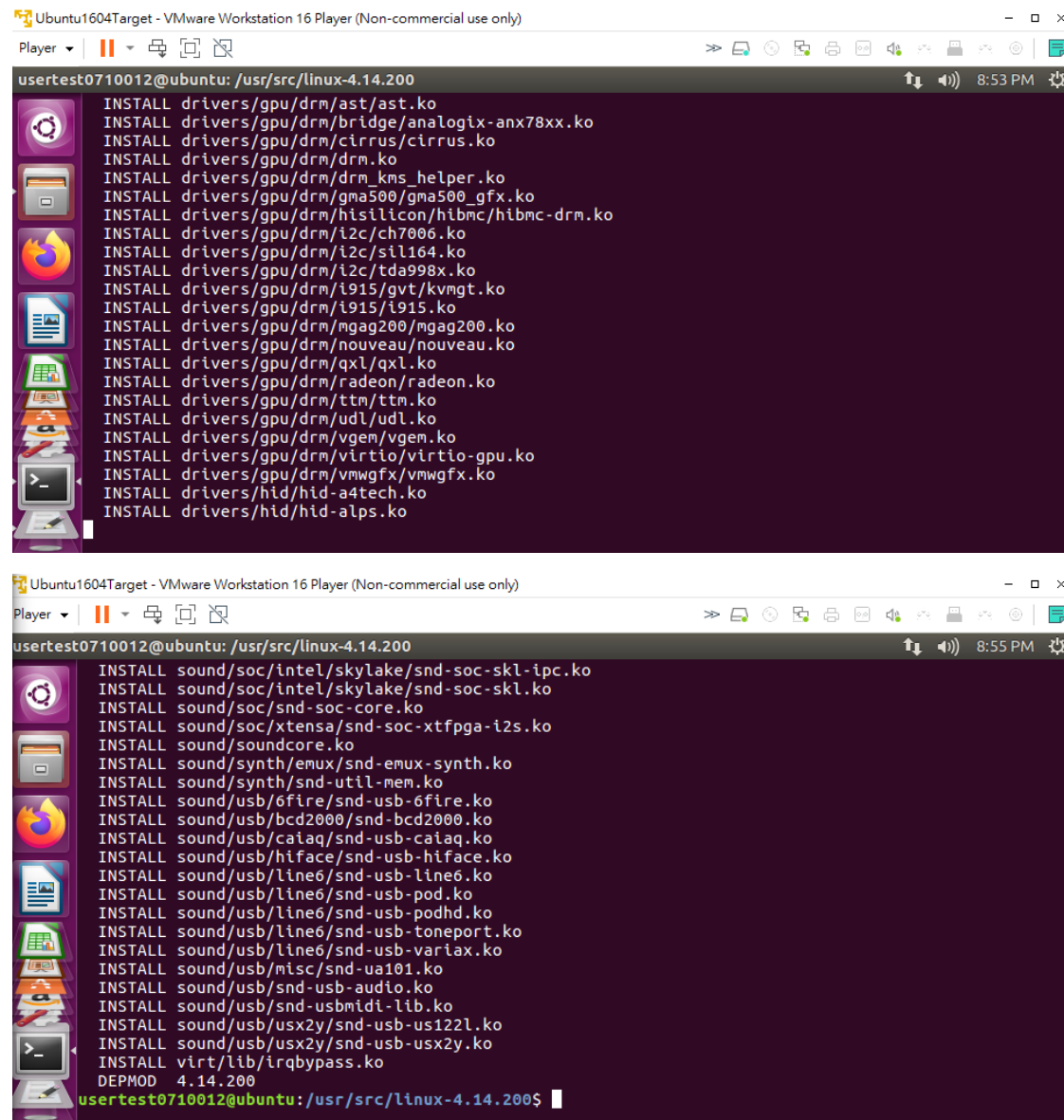
```
user@ubuntu: /usr/src/linux-4.14.200
HOSTCC scripts/selinux/mdp/mdp
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
HOSTCC scripts/recordmcount
HOSTCC scripts/sortextable
HOSTCC scripts/asn1_compiler
HOSTCC scripts/sign-file
HOSTCC scripts/extract-cert
HOSTCC scripts/insert-sys-cert
CC arch/x86/purgatory/purgatory.o
AS arch/x86/purgatory/stack.o
AS arch/x86/purgatory/setup-x86_64.o
CC arch/x86/purgatory/sha256.o
AS arch/x86/purgatory/entry64.o
CC arch/x86/purgatory/string.o
LD arch/x86/purgatory/purgatory.ro
BIN2C arch/x86/purgatory/kexec-purgatory.c
CHK include/generated/timeconst.h
UPD include/generated/timeconst.h
CC kernel/bounds.s
CHK include/generated/bounds.h
UPD include/generated/bounds.h
GEN scripts/gdb/linux/constants.py

LD [M] sound/soc/intel/haswell/snd-soc-sst-haswell-pcm.ko
LD [M] sound/soc/intel/skylake/snd-soc-skl-ipc.ko
LD [M] sound/soc/intel/skylake/snd-soc-skl.ko
LD [M] sound/soc/snd-soc-core.ko
LD [M] sound/soc/xtensa/snd-soc-xtfpga-i2s.ko
LD [M] sound/soundcore.ko
LD [M] sound/synth/emux/snd-emux-synth.ko
LD [M] sound/synth/snd-util-mem.ko
LD [M] sound/usb/6fire/snd-usb-6fire.ko
LD [M] sound/usb/bcd2000/snd-bcd2000.ko
LD [M] sound/usb/caiaq/snd-usb-caiaq.ko
LD [M] sound/usb/hiface/snd-usb-hiface.ko
LD [M] sound/usb/line6/snd-usb-line6.ko
LD [M] sound/usb/line6/snd-usb-pod.ko
LD [M] sound/usb/line6/snd-usb-podhd.ko
LD [M] sound/usb/line6/snd-usb-toneport.ko
LD [M] sound/usb/line6/snd-usb-variak.ko
LD [M] sound/usb/misc/snd-ua101.ko
LD [M] sound/usb/snd-usb-audio.ko
LD [M] sound/usb/snd-usbmidi-lib.ko
LD [M] sound/usb/usx2y/snd-usb-usx2y.ko
LD [M] sound/usb/usx2y/snd-usb-usx2y.ko
LD [M] virt/lib/irqbypass.ko
user@ubuntu: /usr/src/linux-4.14.200$
```


Screenshot7:

Install the module in the `/lib/modules/`uname -r``

`Sudo make modules_install`



The image shows two screenshots of a terminal window running inside a VMware Workstation 16 Player. The terminal is titled "Ubuntu1604Target - VMware Workstation 16 Player (Non-commercial use only)". The user is logged in as "usertest0710012@ubuntu" and is in the directory "/usr/src/linux-4.14.200".

The first screenshot shows the output of the `make modules_install` command, listing the installation of various kernel modules:

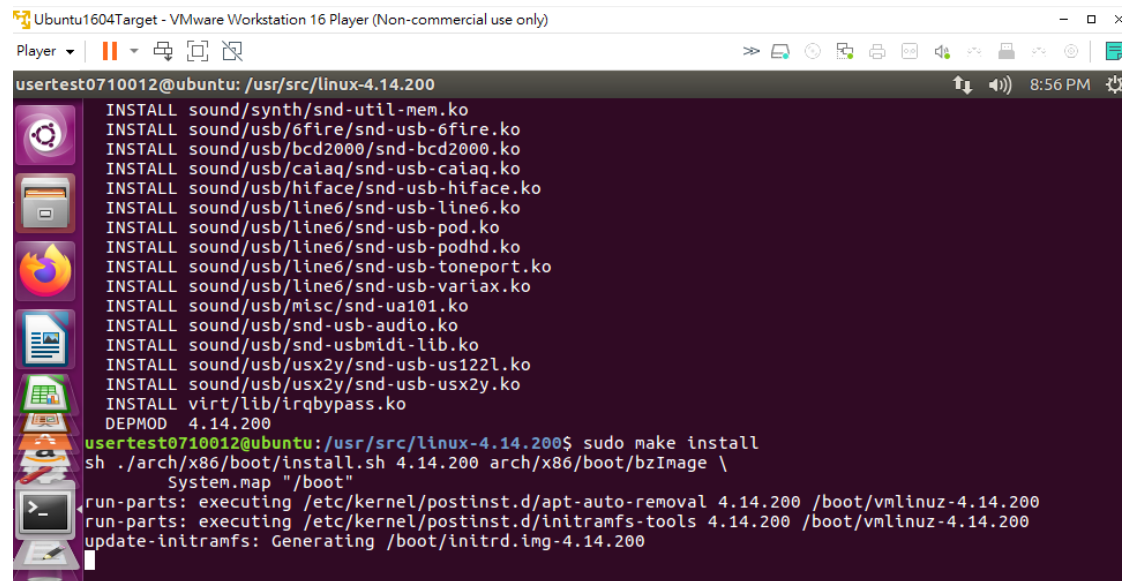
```
INSTALL drivers/gpu/drm/ast/ast.ko
INSTALL drivers/gpu/drm/bridge/analogix-anx78xx.ko
INSTALL drivers/gpu/drm/cirrus/cirrus.ko
INSTALL drivers/gpu/drm/drm.ko
INSTALL drivers/gpu/drm/drm_kms_helper.ko
INSTALL drivers/gpu/drm/gma500/gma500_gfx.ko
INSTALL drivers/gpu/drm/hisilicon/hibmc/hibmc-drm.ko
INSTALL drivers/gpu/drm/i2c/ch7006.ko
INSTALL drivers/gpu/drm/i2c/sil164.ko
INSTALL drivers/gpu/drm/i2c/tda998x.ko
INSTALL drivers/gpu/drm/i915/gvt/kvmgt.ko
INSTALL drivers/gpu/drm/i915/i915.ko
INSTALL drivers/gpu/drm/mgag200/mgag200.ko
INSTALL drivers/gpu/drm/nouveau/nouveau.ko
INSTALL drivers/gpu/drm/qxl/qxl.ko
INSTALL drivers/gpu/drm/radeon/radeon.ko
INSTALL drivers/gpu/drm/ttm/ttm.ko
INSTALL drivers/gpu/drm/udl/udl.ko
INSTALL drivers/gpu/drm/vgem/vgem.ko
INSTALL drivers/gpu/drm/virtio/virtio-gpu.ko
INSTALL drivers/gpu/drm/vmwgfx/vmwgfx.ko
INSTALL drivers/hid/hid-a4tech.ko
INSTALL drivers/hid/hid-alps.ko
```

The second screenshot shows the output of the `make modules_install` command, listing the installation of various kernel modules:

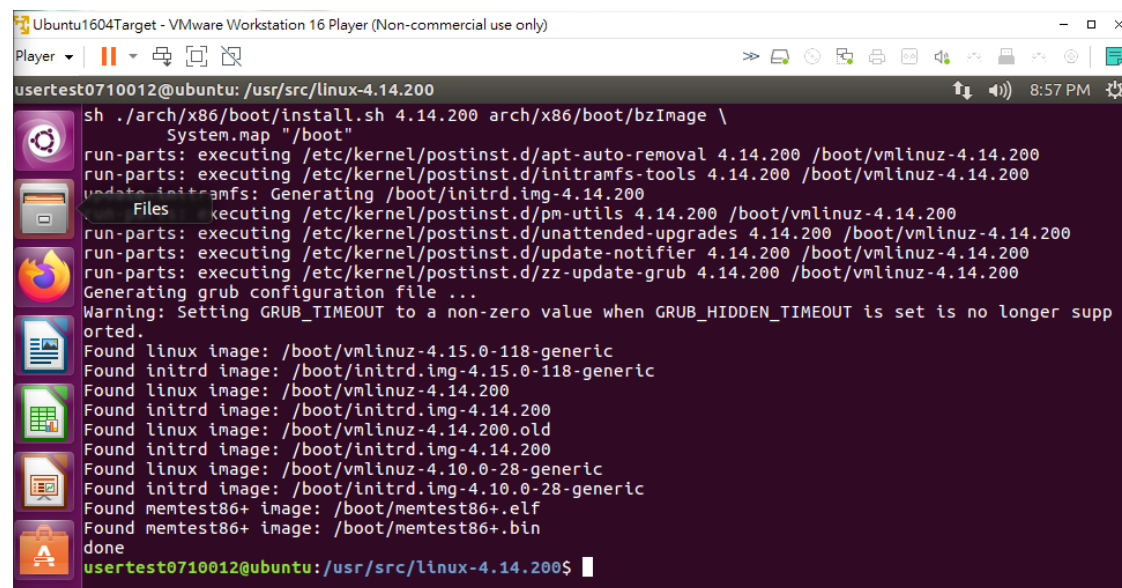
```
INSTALL sound/soc/intel/skylake/snd-soc-skl-ipc.ko
INSTALL sound/soc/intel/skylake/snd-soc-skl.ko
INSTALL sound/soc/snd-soc-core.ko
INSTALL sound/soc/xtensa/snd-soc-xtfpga-i2s.ko
INSTALL sound/soundcore.ko
INSTALL sound/synth/emux/snd-emux-synth.ko
INSTALL sound/synth/snd-util-mem.ko
INSTALL sound/usb/6fire/snd-usb-6fire.ko
INSTALL sound/usb/bcd2000/snd-bcd2000.ko
INSTALL sound/usb/caiaq/snd-usb-caiaq.ko
INSTALL sound/usb/hiface/snd-usb-hiface.ko
INSTALL sound/usb/line6/snd-usb-line6.ko
INSTALL sound/usb/line6/snd-usb-pod.ko
INSTALL sound/usb/line6/snd-usb-podhd.ko
INSTALL sound/usb/line6/snd-usb-toneport.ko
INSTALL sound/usb/line6/snd-usb-variix.ko
INSTALL sound/usb/misc/snd-ua101.ko
INSTALL sound/usb/snd-usb-audio.ko
INSTALL sound/usb/snd-usbmidi-lib.ko
INSTALL sound/usb/usx2y/snd-usb-us122l.ko
INSTALL sound/usb/usx2y/snd-usb-usx2y.ko
INSTALL virt/lib/irqbypass.ko
DEPMOD 4.14.200
usertest0710012@ubuntu: /usr/src/linux-4.14.200$
```

Screenshot 8:

Install the kernel to the machine



```
user@test0710012@ubuntu: /usr/src/linux-4.14.200
INSTALL sound/synth/snd-util-mem.ko
INSTALL sound/usb/6fire/snd-usb-6fire.ko
INSTALL sound/usb/bcd2000/snd-bcd2000.ko
INSTALL sound/usb/caiaq/snd-usb-caiaq.ko
INSTALL sound/usb/hiface/snd-usb-hiface.ko
INSTALL sound/usb/line6/snd-usb-line6.ko
INSTALL sound/usb/line6/snd-usb-pod.ko
INSTALL sound/usb/line6/snd-usb-podhd.ko
INSTALL sound/usb/line6/snd-usb-toneport.ko
INSTALL sound/usb/line6/snd-usb-variax.ko
INSTALL sound/usb/misc/snd-ua101.ko
INSTALL sound/usb/snd-usb-audio.ko
INSTALL sound/usb/snd-usbmidi-lib.ko
INSTALL sound/usb/usx2y/snd-usb-us122l.ko
INSTALL sound/usb/usx2y/snd-usb-usx2y.ko
INSTALL virt/lib/irqbypass.ko
DEPMOD 4.14.200
user@test0710012@ubuntu: /usr/src/linux-4.14.200$ sudo make install
sh ./arch/x86/boot/install.sh 4.14.200 arch/x86/boot/bzImage \
    System.map "/boot"
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 4.14.200 /boot/vmlinuz-4.14.200
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 4.14.200 /boot/vmlinuz-4.14.200
update-initramfs: Generating /boot/initrd.img-4.14.200
```



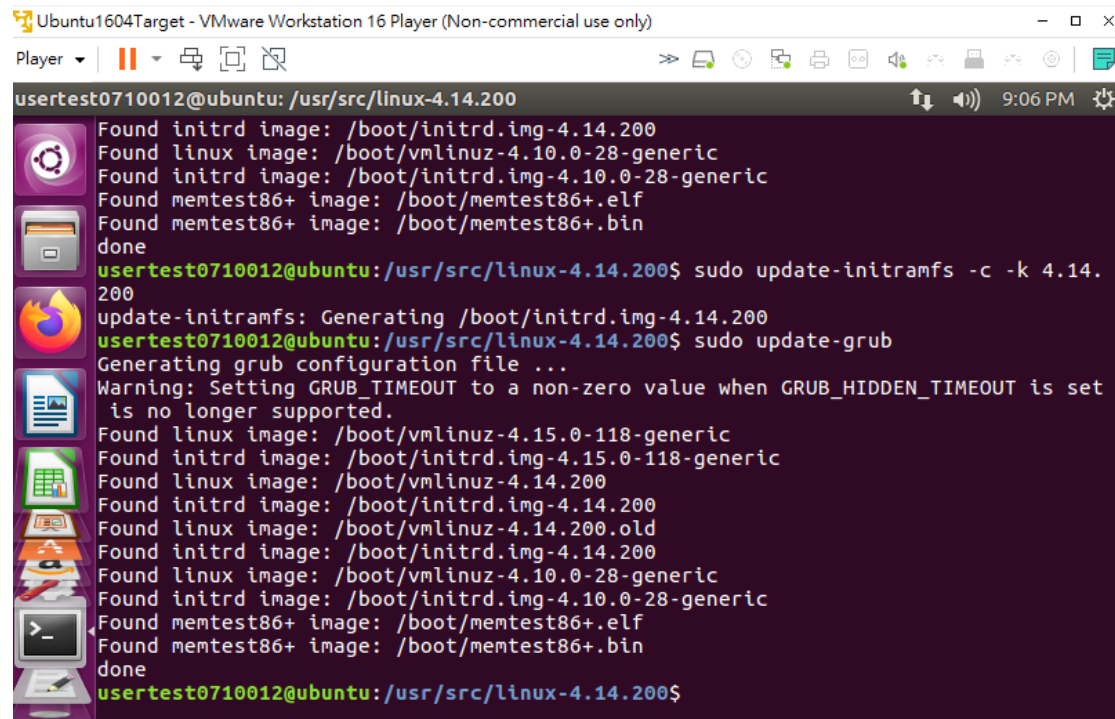
```
user@test0710012@ubuntu: /usr/src/linux-4.14.200
sh ./arch/x86/boot/install.sh 4.14.200 arch/x86/boot/bzImage \
    System.map "/boot"
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 4.14.200 /boot/vmlinuz-4.14.200
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 4.14.200 /boot/vmlinuz-4.14.200
update-initramfs: Generating /boot/initrd.img-4.14.200
Files: executing /etc/kernel/postinst.d/pm-utils 4.14.200 /boot/vmlinuz-4.14.200
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 4.14.200 /boot/vmlinuz-4.14.200
run-parts: executing /etc/kernel/postinst.d/update-notifier 4.14.200 /boot/vmlinuz-4.14.200
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 4.14.200 /boot/vmlinuz-4.14.200
Generating grub configuration file ...
Warning: Setting GRUB_TIMEOUT to a non-zero value when GRUB_HIDDEN_TIMEOUT is set is no longer supported.
Found linux image: /boot/vmlinuz-4.15.0-118-generic
Found initrd image: /boot/initrd.img-4.15.0-118-generic
Found linux image: /boot/vmlinuz-4.14.200
Found initrd image: /boot/initrd.img-4.14.200
Found linux image: /boot/vmlinuz-4.14.200.old
Found initrd image: /boot/initrd.img-4.14.200
Found linux image: /boot/vmlinuz-4.10.0-28-generic
Found initrd image: /boot/initrd.img-4.10.0-28-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
done
user@test0710012@ubuntu: /usr/src/linux-4.14.200$
```

Screenshot 9:

Refresh the information in the grub after commenting out the two command

#GRUB_HIDDEN_TIMEOUT=0→can jump to the menu when jumping up the machine

#GRUB_HIDDEN_TIMEOUT_QUIET=true→won't show count down at the menu

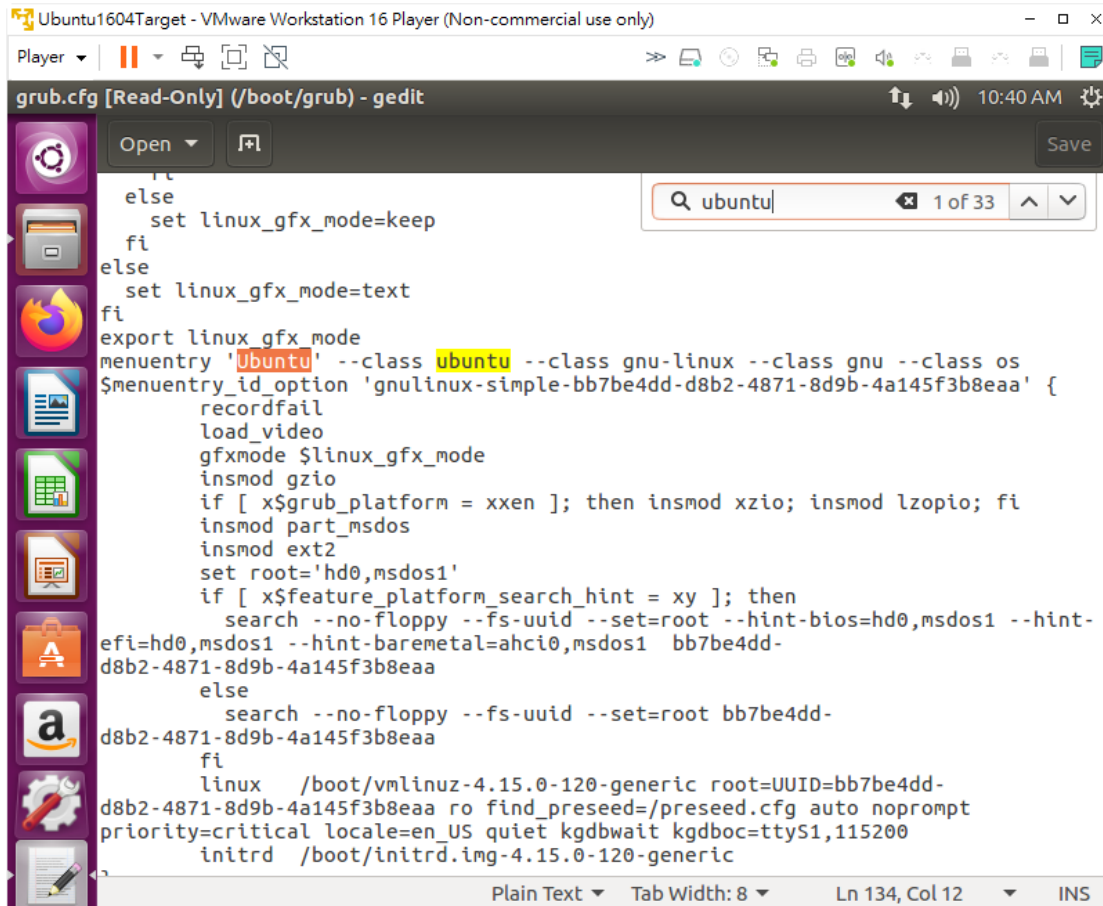


The screenshot shows a terminal window titled "Ubuntu1604Target - VMware Workstation 16 Player (Non-commercial use only)". The terminal is running as "usertest0710012@ubuntu: /usr/src/linux-4.14.200". The output of the commands is as follows:

```
Found initrd image: /boot/initrd.img-4.14.200
Found linux image: /boot/vmlinuz-4.10.0-28-generic
Found initrd image: /boot/initrd.img-4.10.0-28-generic
Found mentest86+ image: /boot/mentest86+.elf
Found mentest86+ image: /boot/mentest86+.bin
done
usertest0710012@ubuntu: /usr/src/linux-4.14.200$ sudo update-initramfs -c -k 4.14.200
update-initramfs: Generating /boot/initrd.img-4.14.200
usertest0710012@ubuntu: /usr/src/linux-4.14.200$ sudo update-grub
Generating grub configuration file ...
Warning: Setting GRUB_TIMEOUT to a non-zero value when GRUB_HIDDEN_TIMEOUT is set
is no longer supported.
Found linux image: /boot/vmlinuz-4.15.0-118-generic
Found initrd image: /boot/initrd.img-4.15.0-118-generic
Found linux image: /boot/vmlinuz-4.14.200
Found initrd image: /boot/initrd.img-4.14.200
Found linux image: /boot/vmlinuz-4.14.200.old
Found initrd image: /boot/initrd.img-4.14.200
Found linux image: /boot/vmlinuz-4.10.0-28-generic
Found initrd image: /boot/initrd.img-4.10.0-28-generic
Found mentest86+ image: /boot/mentest86+.elf
Found mentest86+ image: /boot/mentest86+.bin
done
usertest0710012@ubuntu: /usr/src/linux-4.14.200$
```


Screenshot 10:

Use nano command to insert the command to the desired place, making the target machine able to wait for the remote gdb control



Ubuntu1604Target - VMware Workstation 16 Player (Non-commercial use only)

grub.cfg [Read-Only] (/boot/grub) - gedit

10:40 AM

Open Save

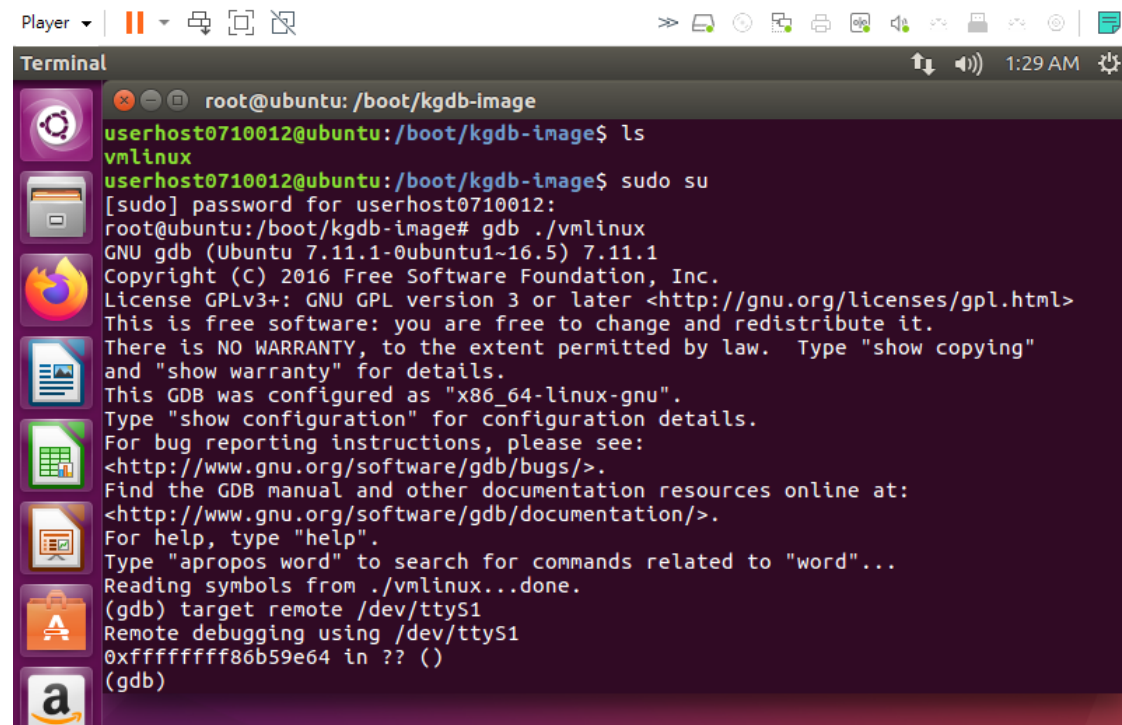
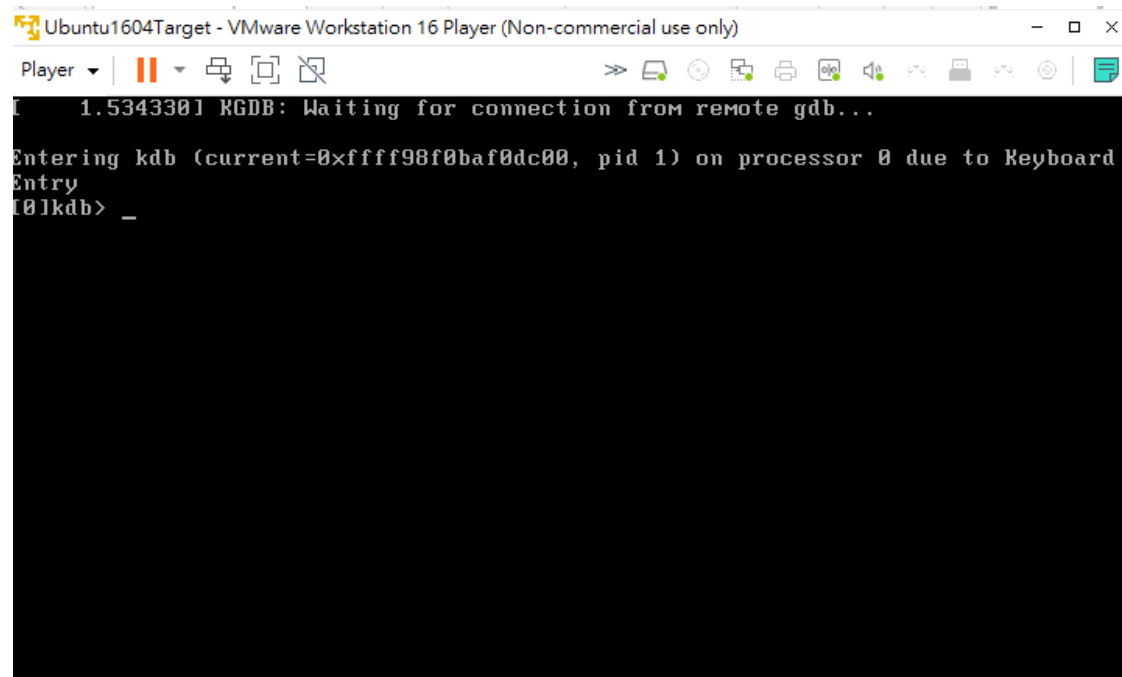
Search: ubuntu 1 of 33

```
else
    set linux_gfx_mode=keep
fi
else
    set linux_gfx_mode=text
fi
export linux_gfx_mode
menuentry 'Ubuntu' --class ubuntu --class gnu-linux --class gnu --class os
$menuentry_id_option 'gnulinux-simple-bb7be4dd-d8b2-4871-8d9b-4a145f3b8eaa' {
    recordfail
    load_video
    gfxmode $linux_gfx_mode
    insmod gzio
    if [ x$grub_platform = xxen ]; then insmod xzio; insmod lzopio; fi
    insmod part_msdos
    insmod ext2
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-
efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 bb7be4dd-
d8b2-4871-8d9b-4a145f3b8eaa
    else
        search --no-floppy --fs-uuid --set=root bb7be4dd-
d8b2-4871-8d9b-4a145f3b8eaa
    fi
    linux /boot/vmlinuz-4.15.0-120-generic root=UUID=bb7be4dd-
d8b2-4871-8d9b-4a145f3b8eaa ro find_preseed=/preseed.cfg auto noprompt
priority=critical locale=en_US quiet kgdbwait kgdboc=ttyS1,115200
    initrd /boot/initrd.img-4.15.0-120-generic
}
```

Plain Text Tab Width: 8 Ln 134, Col 12 INS

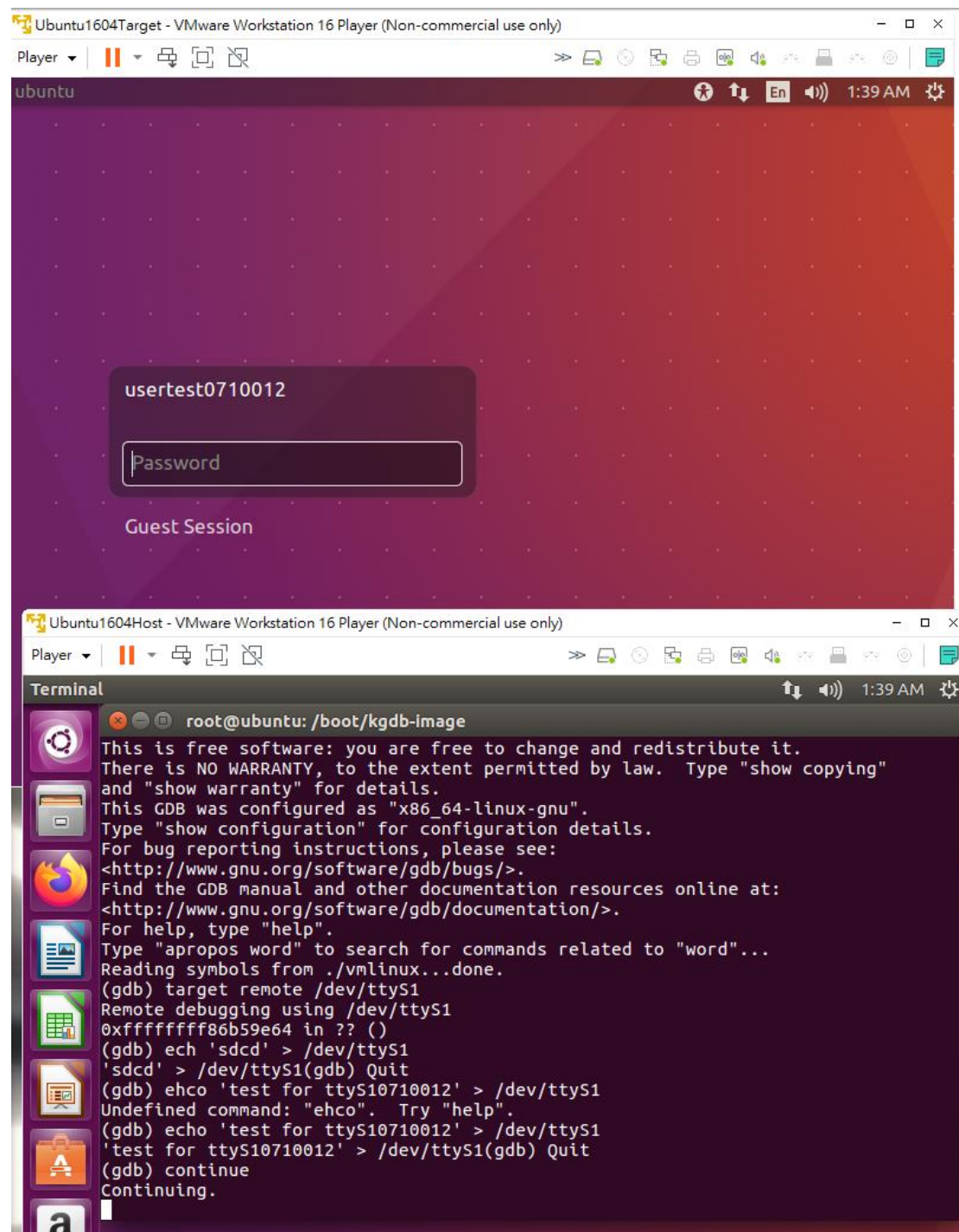
Screenshot 11:

Activate gdb from the ./vmlinux file in kgdb-image and connect to the target machine through the /dev/ttyS1 prot



Screenshot 12:

The target machine finish booting up process after the host command it to continue



Questions:

1. Kernel: the one program running at all time on the computer
Mainline kernel: where new development and new features introduced
Stable kernel: when mainline kernel is released, it is consider "stable". Bugfix version of mainline kernel
Longterm kernel: for backporting bugfixes for older kernel tree
Kernel panic: the action the operating system will take when it find a fatal error it can't solve safely
2. building: to make a program executable
debugging: modify the program to make it perform as the programmer wishes
profiling: find out which part of the program takes most of the time, and optimize it
3. GCC: the GNU compiler collection, standard compiler for most projects related to GNU and Linux
GDB: the GNU debugger, a free debugging program under Linux
KGDB: debugger for the Linux kernel
4. /usr/: user system resources, has similar catalog as root, store user's files
/boot/: store the file when activate when the Linux machine turn on
/home/: store normal user's personal file
/boot/grub: store bootloader
5. Have two machine, and install the kernel in one of them(target), after that, we use the other machine to run GDB and debug the kernel in the target machine through serial port.
6. Because we need one of the machine(target) to run the patched kernel and the other(host) to run GDB to debug the target.
7. Make: read the text in makefile and compile related files
Make modules_install: install the module in the /lib/modules/`uname -r`
Make install: install the kernel to the machine
8. Kgdbwait: command that makes kgdb to wait for a debugger connection during booting of a kernel
Kgdboc=ttyS1,115200: kgbd over console, through port ttyS1 with 115200 Bdn data transfer speed
9. Grub: Grand unified bootloader, let user choose to boot one of the multiple OS, or select a specific kernel.
Grub.cfg: a file containing information of grub, refresh when "update-grub" is run
10. Commands: ex, help, show, list, file, clear, info, delete, enable, disable, condition

Player ▾ | [Icons] | 11:35 PM

Terminal

root@ubuntu: /home/userhost0710012

```
running -- Running the program
stack -- Examining the stack
status -- Status inquiries
support -- Support facilities
tracepoints -- Tracing of program execution without stopping the program
user-defined -- User-defined commands

Type "help" followed by a class name for a list of commands in that class.
Type "help all" for the list of all commands.
Type "help" followed by command name for full documentation.
Type "apropos word" to search for commands related to "word".
Command name abbreviations are allowed if unambiguous.
(gdb) break
No default breakpoint address now.
(gdb) clear
No source file specified.
(gdb) deldeete
Undefined command: "deldeete". Try "help".
(gdb) delete
(gdb) file
No executable file now.
No symbol file now.
(gdb) enable
(gdb) |
```

Ubuntu1604Host - VMware Workstation 16 Player (Non-commercial use only)

Player ▾ | [Icons] | 11:35 PM

Terminal

root@ubuntu: /home/userhost0710012

```
(gdb) enable
(gdb) info
"info" must be followed by the name of an info command.
List of info subcommands:

info address -- Describe where symbol SYM is stored
info all-registers -- List of all registers and their contents
info args -- Argument variables of current stack frame
info auto-load -- Print current status of auto-loaded files
info auto-load-scripts -- Print the list of automatically loaded Python scripts
info auxv -- Display the inferior's auxiliary vector
info bookmarks -- Status of user-settable bookmarks
info breakpoints -- Status of specified breakpoints (all user-settable breakpoi
ts if no argument)
info checkpoints -- IDs of currently known checkpoints
info classes -- All Objective-C classes
info common -- Print out the values contained in a Fortran COMMON block
info copying -- Conditions for redistributing copies of GDB
info dcache -- Print information on the dcache performance
info display -- Expressions to display when program stops
info exceptions -- List all Ada exception names
info extensions -- All filename extensions associated with a source language
info files -- Names of targets and files being debugged
info float -- Print the status of the floating point unit
```