운영체제

담당 교수님: 김태석 교수님

학번: 2016722074

이름: 김영태

실습번호: Assignment1

Assignment 1-1

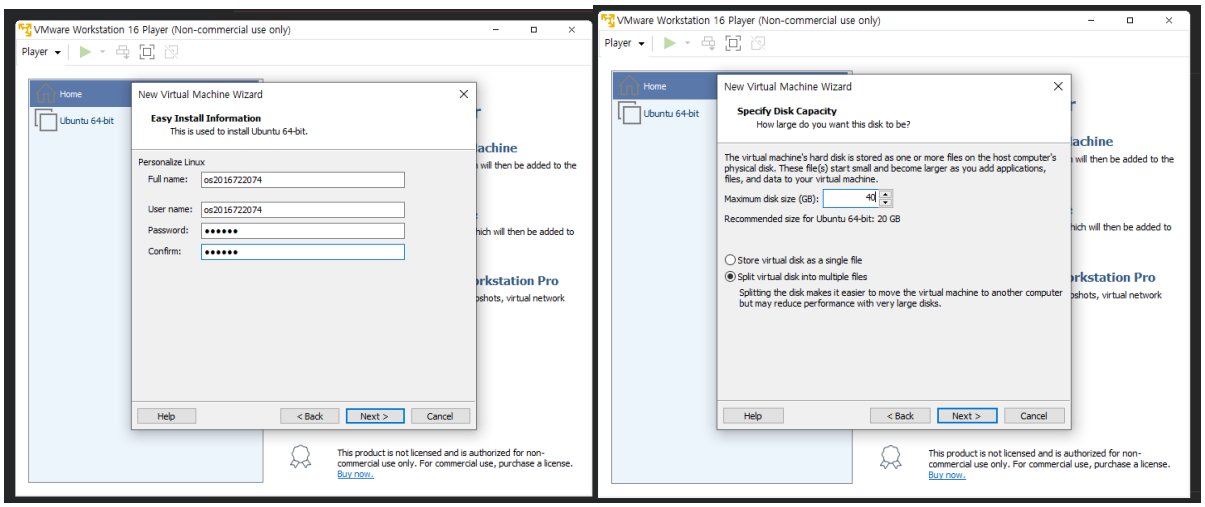
Introduction

가상 환경에서 OS를 설치할 수 있는 VMware를 설치하고 그 위에 Ubuntu를 설치한다.

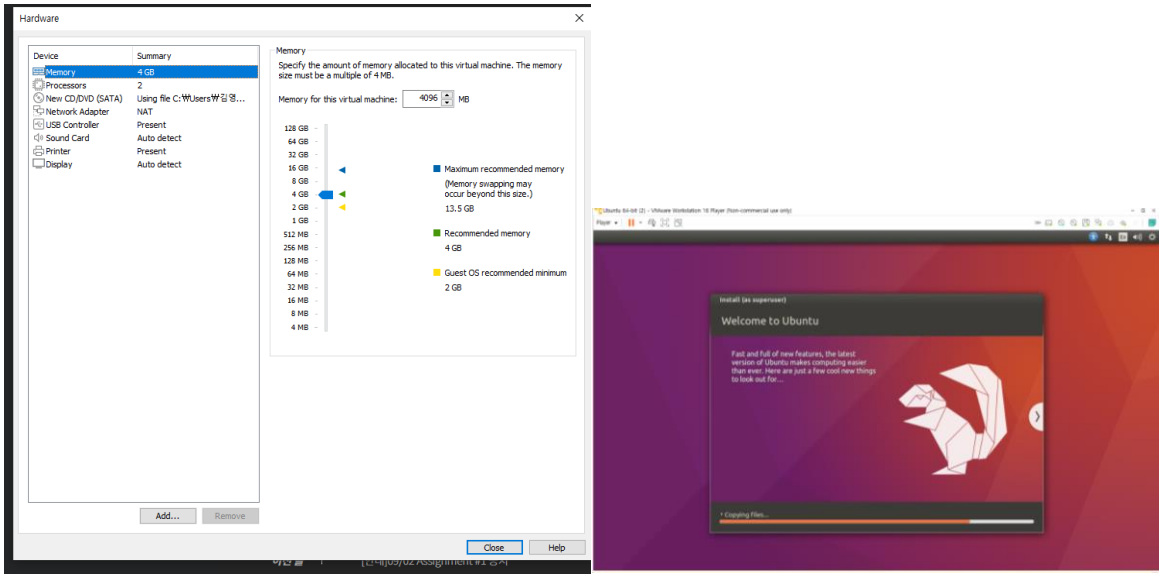
Reference

강의자료

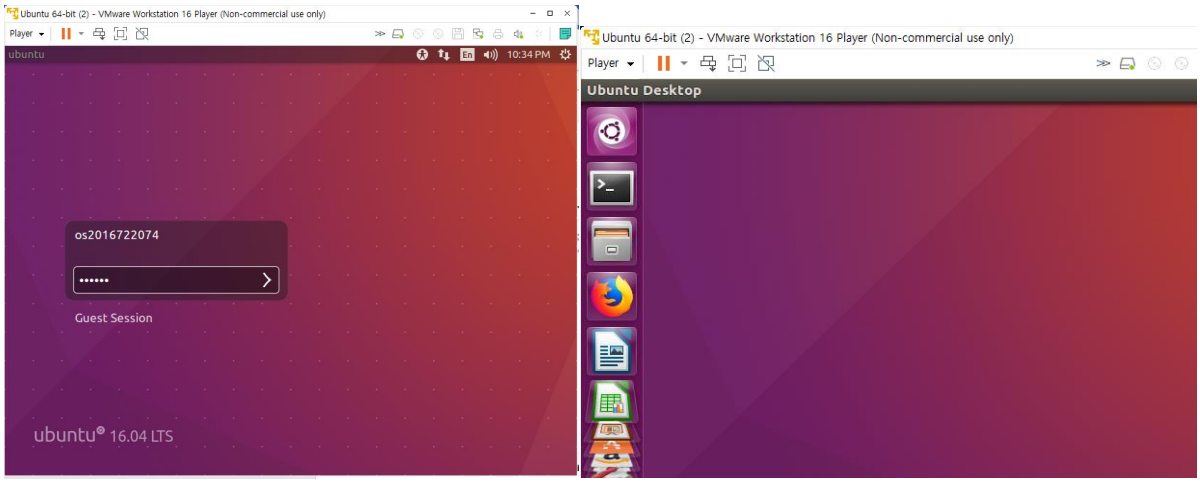
Conclusion



먼저 VMware를 실행시키고 강의자료를 통해 다운받은 iso파일을 이용해 우분투를 설치한다. 사용자 이름과 비밀번호를 설정한다. 그 후 가상머신에 할당할 저장공간 크기를 설정한다. Disk Size는 40Gbytes로 설정했다.



메모리 크기와 CPU개수를 설정한 후 Finish버튼을 누르면 Ubuntu 설치가 진행된다.



Ubuntu 설치 완료 후 방금 전 설정했던 사용자 비밀번호를 입력한다. 터미널을 dock에 고정한다.

Assignment 1-2

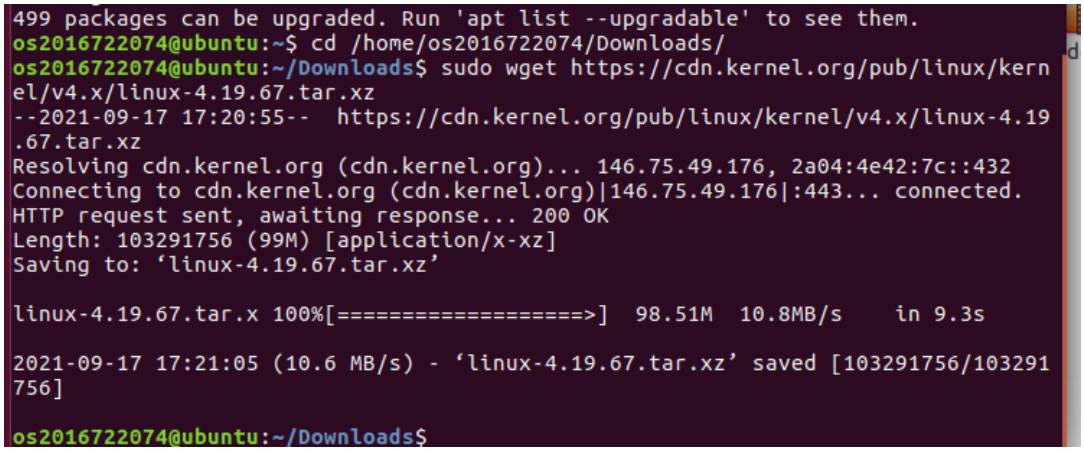
Introduction

가상머신에 설치된 우분투에 Kenrel을 다운로드 후 kernel을 컴파일 한다. 커널 컴파일 과정을 확인해 본다.

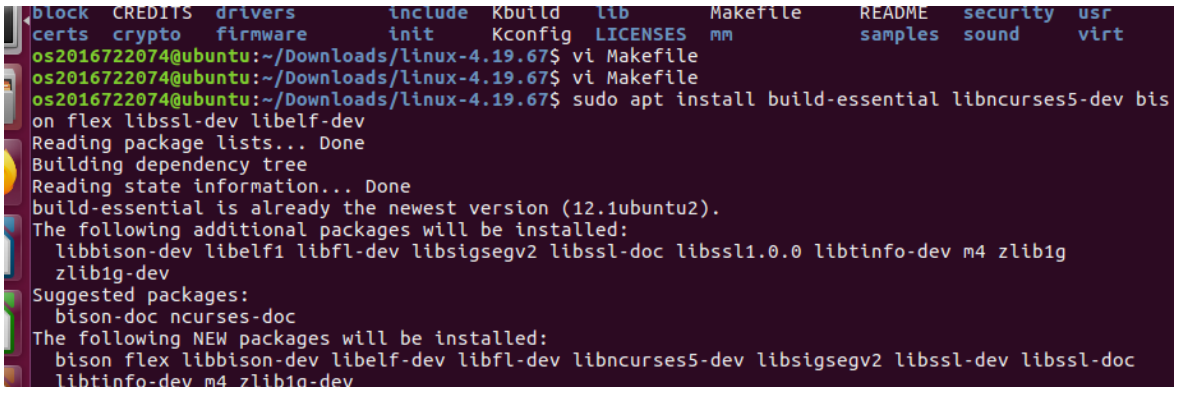
Reference

강의 자료

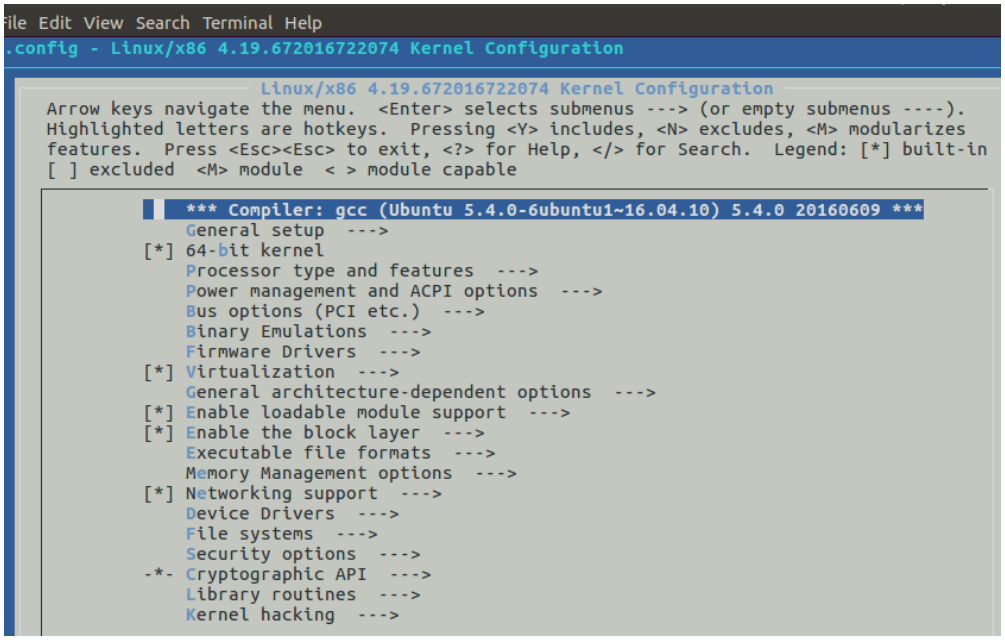
Conclusion



먼저 우분투의 Downloads에 커널 압축 파일을 다운받는다. 이때 sudo명령어를 사용해 root권한을 얻는다.

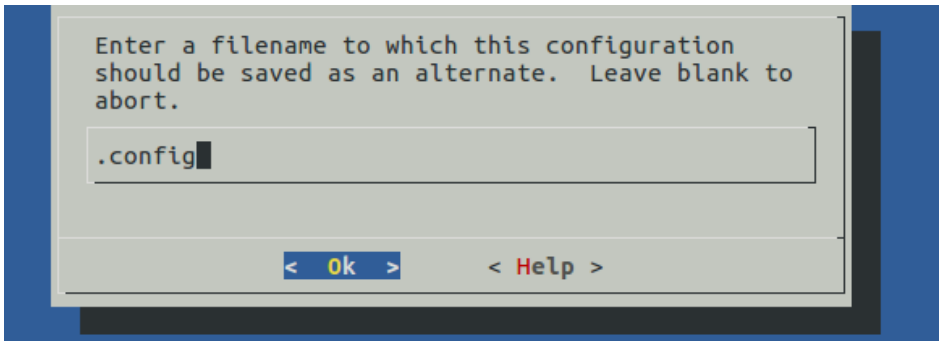


그 후 압축 파일을 압축 풀기 후 vi에디터인 vim을 사용해 Makefile의 코드를 수정한다.



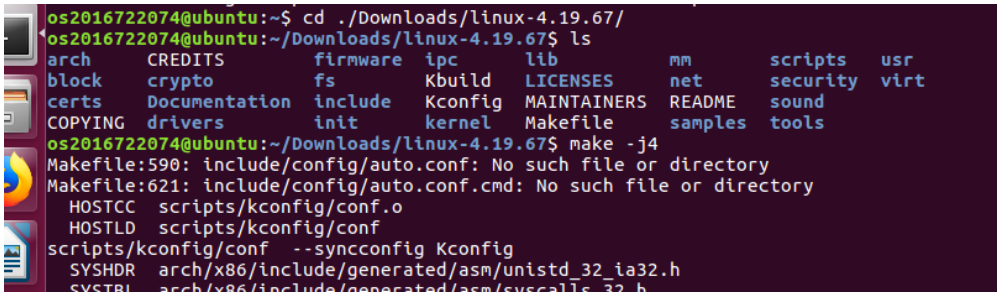
다음으로 커널의 환경설정을 위해 root권한으로 파일을 설치 후 커널환경을 설정한다.

커널 모듈 적재 시 발생할 수 있는 문제를 해결하기 위해 "Enable loadable module support" 의 "Forced module loading"을 체크하고, 컴파일 시 문제가 될 수 있는 모듈을 제거하기 위해 "Device Drivers"의 "Staging drivers"를 체크 해제한다.

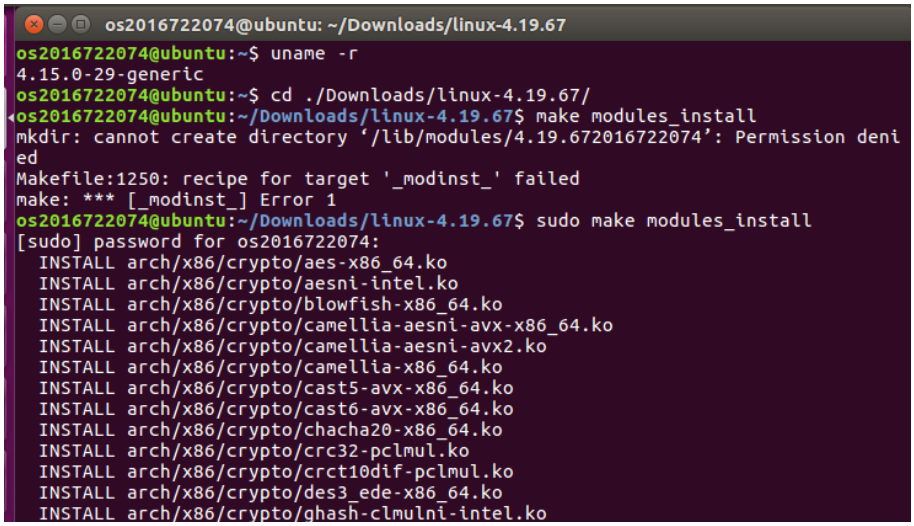


모든 설정을 마친 후 .config파일에 저장하고 종료한다.

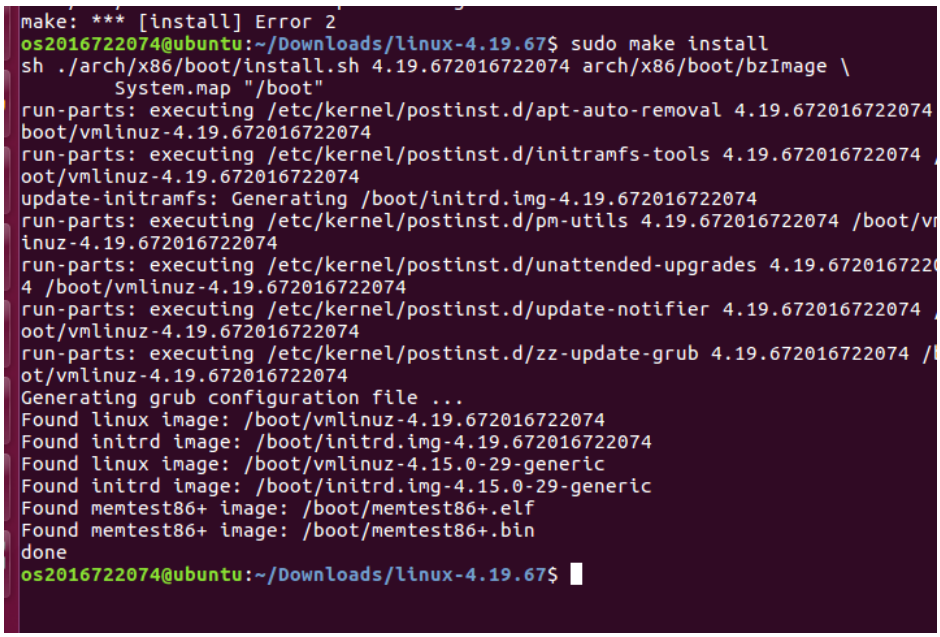
```
scripts/kconfig/mconf  Kconfig
#
# using defaults found in /boot/config-4.15.0-29-generic
#
/boot/config-4.15.0-29-generic:890:warning: symbol value 'm' invalid for HOTPLUG_PCI_SHPC
/boot/config-4.15.0-29-generic:1144:warning: symbol value 'm' invalid for NF_NAT_REDIRECT
/boot/config-4.15.0-29-generic:1147:warning: symbol value 'm' invalid for NF_TABLES_INET
/boot/config-4.15.0-29-generic:1148:warning: symbol value 'm' invalid for NF_TABLES_NETDEV
/boot/config-4.15.0-29-generic:1331:warning: symbol value 'm' invalid for NF_TABLES_IPV4
/boot/config-4.15.0-29-generic:1336:warning: symbol value 'm' invalid for NF_TABLES_ARP
/boot/config-4.15.0-29-generic:1343:warning: symbol value 'm' invalid for NF_NAT_MASQUERADE_IPV4
/boot/config-4.15.0-29-generic:1378:warning: symbol value 'm' invalid for NF_TABLES_IPV6
/boot/config-4.15.0-29-generic:1388:warning: symbol value 'm' invalid for NF_NAT_MASQUERADE_IPV6
/boot/config-4.15.0-29-generic:1416:warning: symbol value 'm' invalid for NF_TABLES_BRIDGE
/boot/config-4.15.0-29-generic:3992:warning: symbol value 'm' invalid for HW_RANDOM_TPM
/boot/config-4.15.0-29-generic:4941:warning: symbol value 'm' invalid for LIRC
/boot/config-4.15.0-29-generic:6167:warning: symbol value 'm' invalid for SND_SOC_INTEL_SST_TOPLEVE
L
/boot/config-4.15.0-29-generic:6172:warning: symbol value 'm' invalid for SND_SOC_INTEL_MACH
/boot/config-4.15.0-29-generic:7725:warning: symbol value 'm' invalid for DELL_SMBIOS_WMI
/boot/config-4.15.0-29-generic:7726:warning: symbol value 'm' invalid for DELL_SMBIOS_SMM


*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

os2016722074@ubuntu:~/Downloads/linux-4.19.67$
```



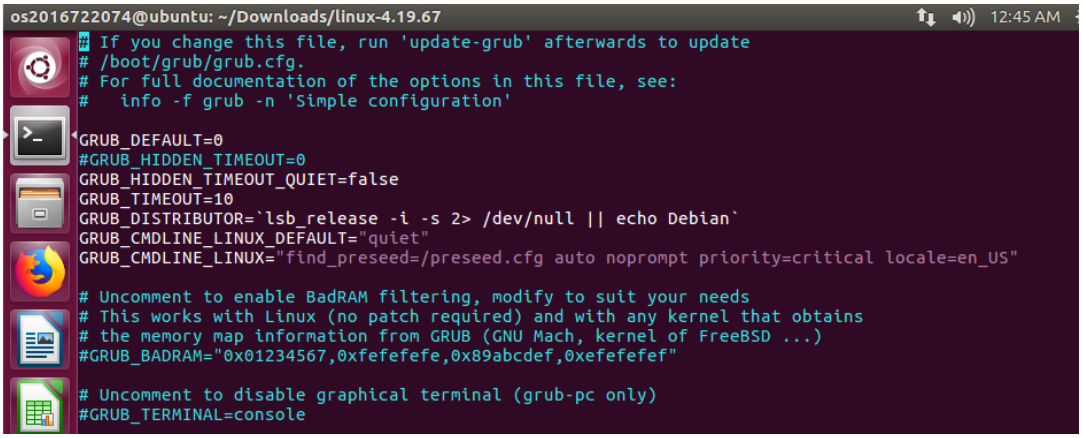
Make 명령어를 사용해서 Kernel을 컴파일 한다. 이때 인자로 스레드의 개수를 받는데, 스레드의 개수를 4개로 설정해서 컴파일을 진행했다.



커널 컴파일이 끝나면 컴파일된 모듈을 /lib/modules로 이동한다.

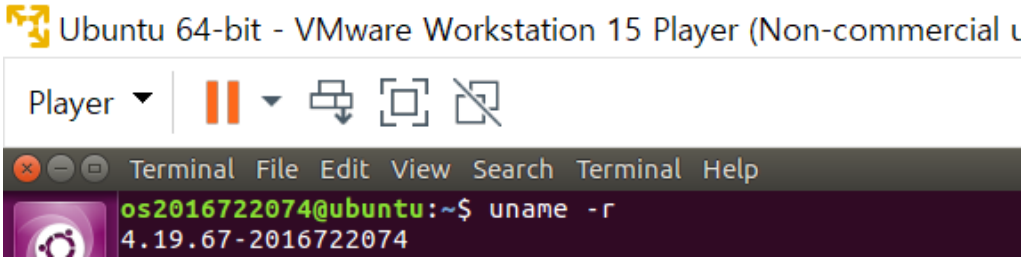


다시 root권한으로 명령어를 입력해 컴파일된 커널을 boot loader에 등록한다.



다음으로 Grub 파일을 수정했다. 이때 root권한이 없다면 파일이 readonly로 열리기 때문에 sudo 명령어를 사용해서 파일을 수정할 수 있게 한다.

위의 모든 과정을 마치고 터미널에서 reboot명령어를 입력해서 시스템을 재부팅 했다. 부팅 후 터미널에서 uname -r명령어를 입력해 현재 커널의 버전을 확인했다.



Assignment 1-3

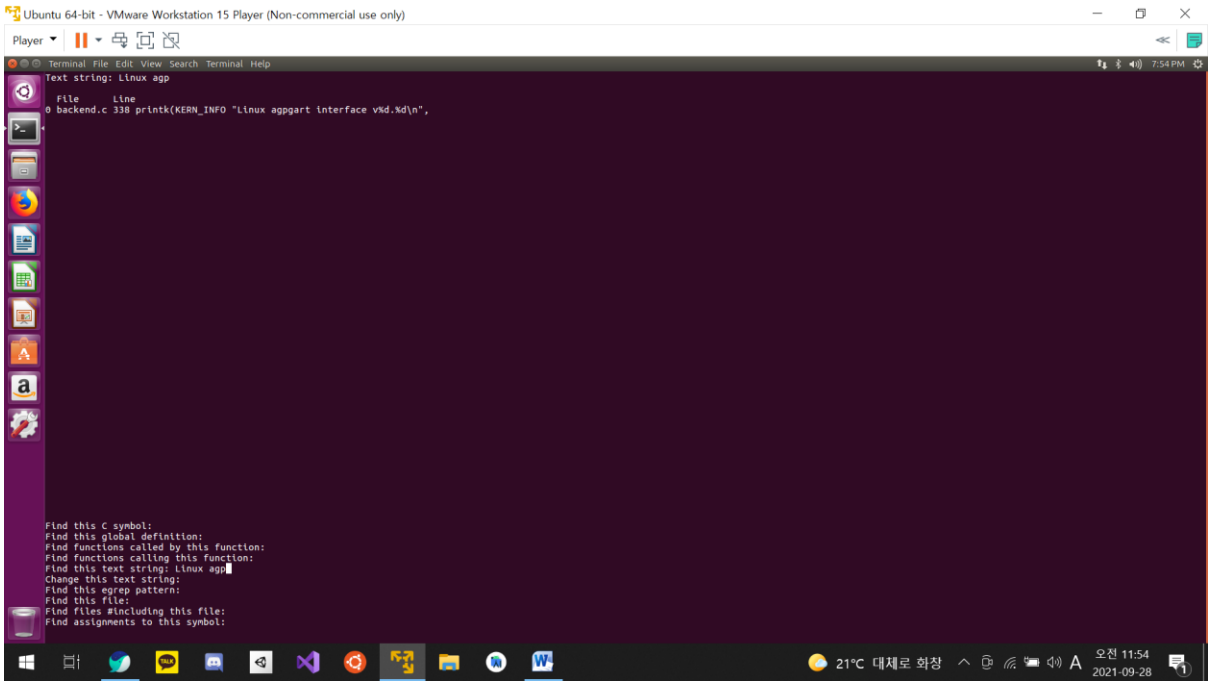
Introduction

cscope를 사용해커널 코드 내의 코드를 수정하여 Linux agp...이 실행되는 지점에 커널 메시지가 출력되도록 커널 코드를 수정한다. 이때 Linux agp...을 실행 시키는 함수의 함수명과 argument의 값을 출력하도록 커널코드를 수정한다.

Reference

강의 자료

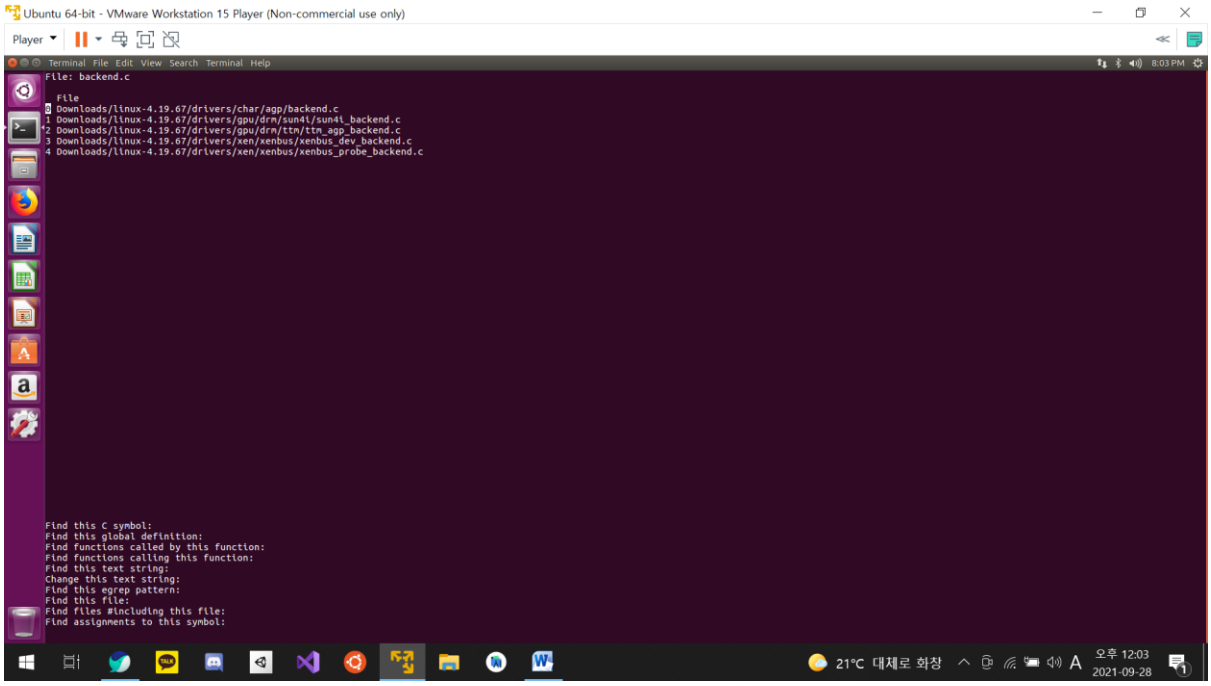
Conclusion



scope에서 Linux agp 라는 문자열을 검색했다. Backend.c 라는 파일의 338번째 줄에 Linux agp가 적혀있는 것을 확인 했다. tab키를 눌러 검색된 파일 목록의 선택 화면으로 바꾸고 엔터 키를 눌러 파일을 확인한다.

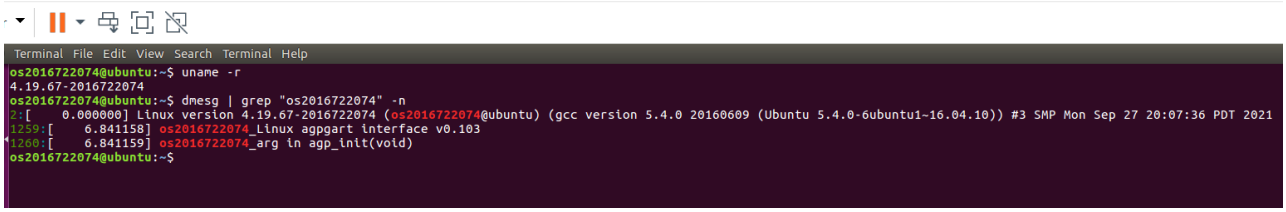
```
static int __init agp_init(void)
{
        if (!agp_off)
                printk(KERN_INFO "os2016722074_Linux agpgart interface v%d.%d\n",
                        AGPGART_VERSION_MAJOR, AGPGART_VERSION_MINOR);
        printk(KERN_INFO "os2016722074_arg in agp_init(void)\n");

        return 0;
}
```

agp_init 함수 내용을 수정한다. 파일의 내부를 확인해 보니 agp_init함수의 내부에서 Linux agp라는 문자열이 적혀있는 것을 확인할 수 있었다. printk함수는 커널메시지를 출력하는 함수 이므로 해당 함수 내에 적힌 문자열을 수정하여 과제의 요구사항을 만족시켰다. 그 후 다시 printk함수를 호출하여 Linux agp문자열이 적혀있는 함수의 이름과 argument인 void를 적어 커널메시지로 출력하게 했다.



조금전 위에서 확인했다시피 Linux agp문자열이 적혀있던 파일은 backend.c 파일 단 하나뿐이었다. Vi 에디터에서 :wq를 입력해 수정한 코드를 저장하고 다시 cscope화면으로 돌아왔다. Find this file: 에서 backend.c를 입력해서 이 파일이 어느 경로에 위치해 있는지 검색했다.

검색한 소스코드의 path는 Downloads/linux-4.10.67/drivers/char/agp/backend.c 인 것을 확인했다.



다시 커널컴파일의 모든과정을 진행하고 터미널에 reboot명령어를 입력해 시스템을 재부팅 시켰다. 그 후 dmesg | grep 명령어를 사용해 printk()로 출력한 내용 중 특정 문자열이 포함된 메시지를 확인했다. 위에서 backend.c 파일을 수정해서 적었던 내용이 출력된 것을 확인했다.