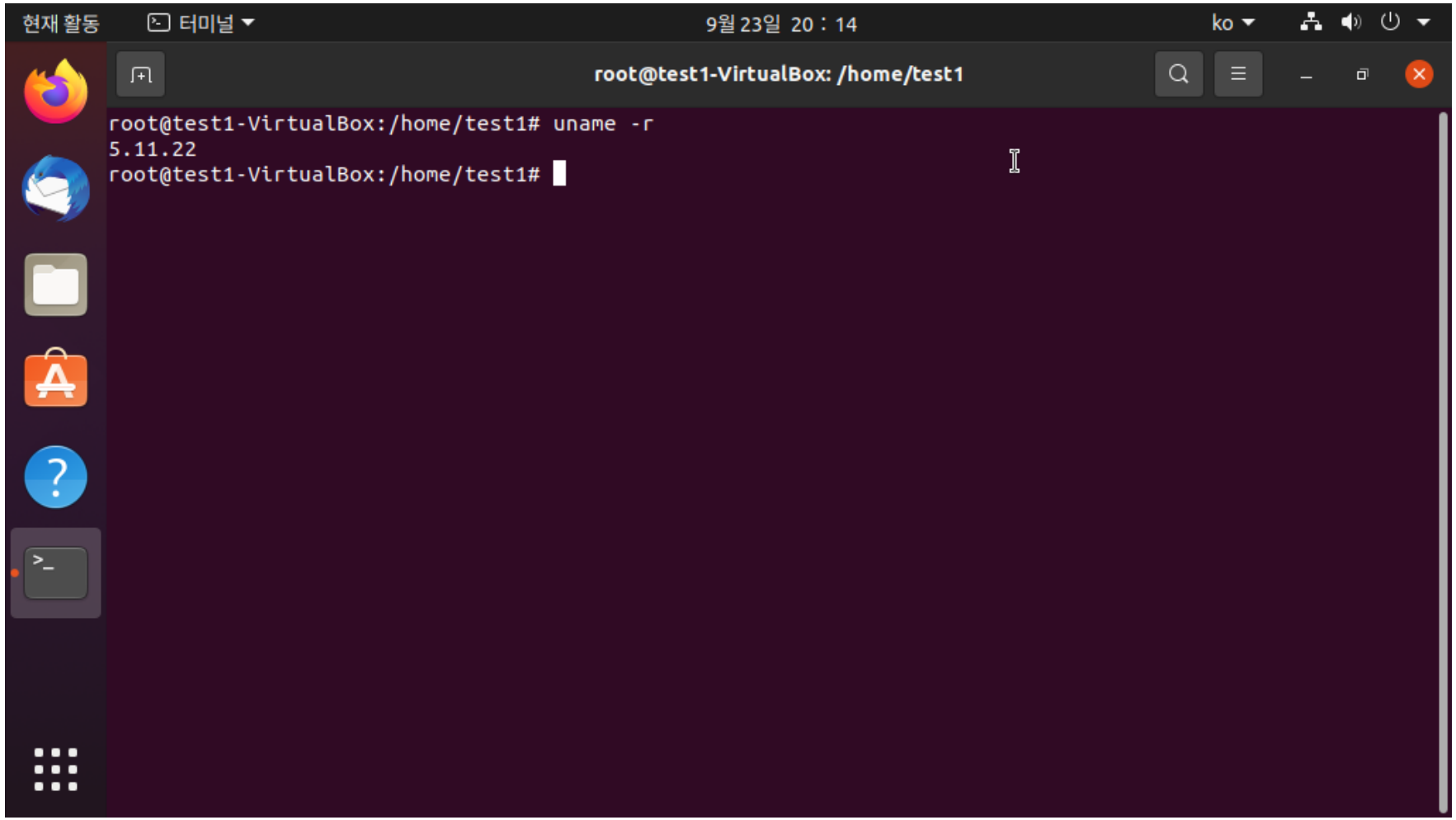
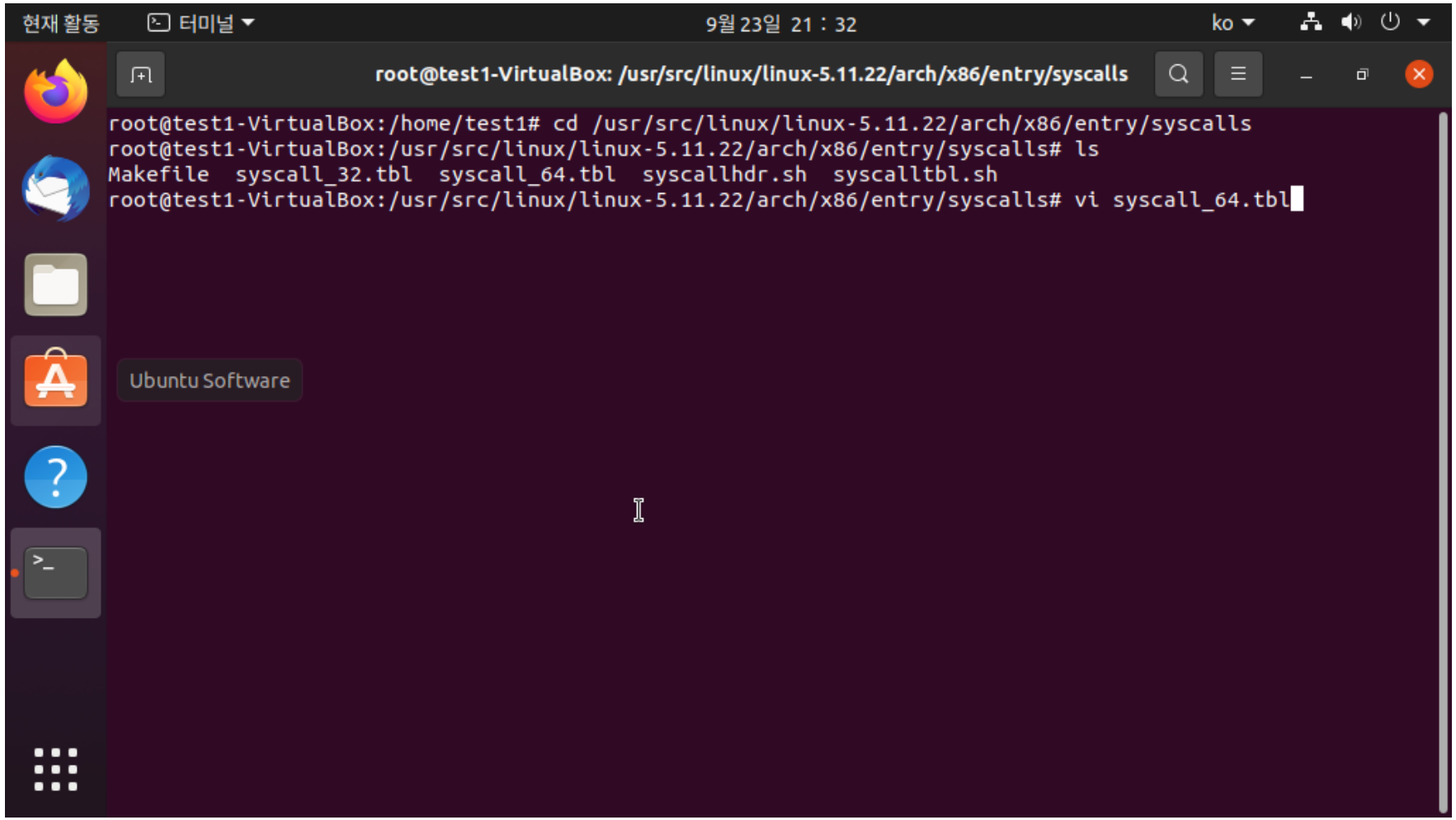
과제 #3 참고자료

(Hello World 시스템 콜 추가)



*** 3,4번 과제는 커널 버전 5.11.22 설치된 환경에서 실시함 ***

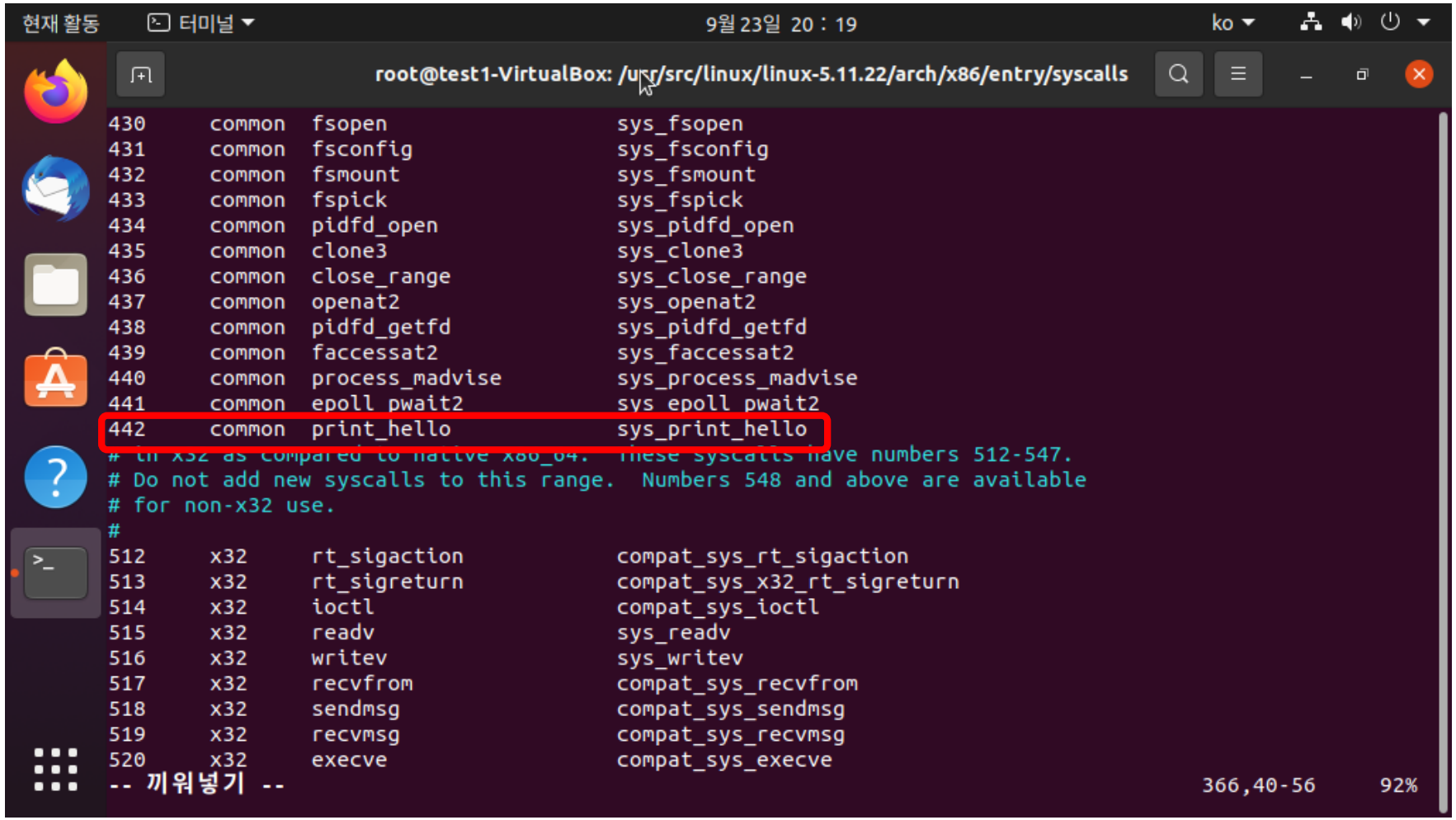
1. 시스템 콜 추가 – 시스템 콜 테이블 등록



/usr/src/linux/linux-5.11.22/arch/x86/entry/syscalls 디렉토리로 이동 후 운영체제 종류 (64bit 혹은 32bit)에 맞는 syscall_xx.tbl 파일 편집

* 본 참고자료의 경우 64bit로 진행함

1. 시스템 콜 추가 – 시스템 콜 테이블 등록



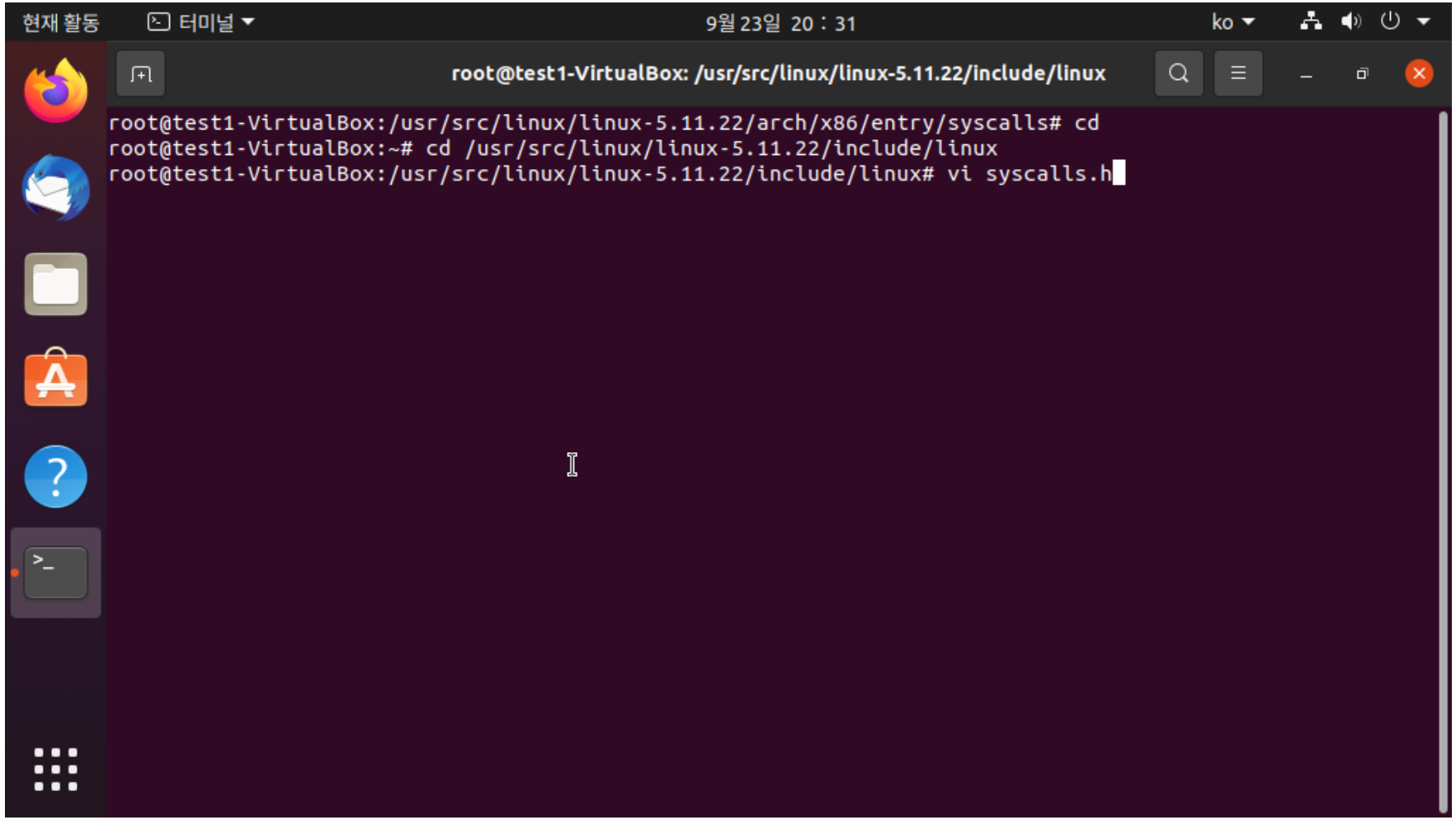
시스템 콜 테이블은 <number> <abi> <name> <entry point> 순으로 구성되어 있음

64bit 시스템 콜 테이블의 마지막에 이어지도록 작성함

도중에 사용 불가능한 구간이 존재하므로 주석을 잘 읽어보고 작성

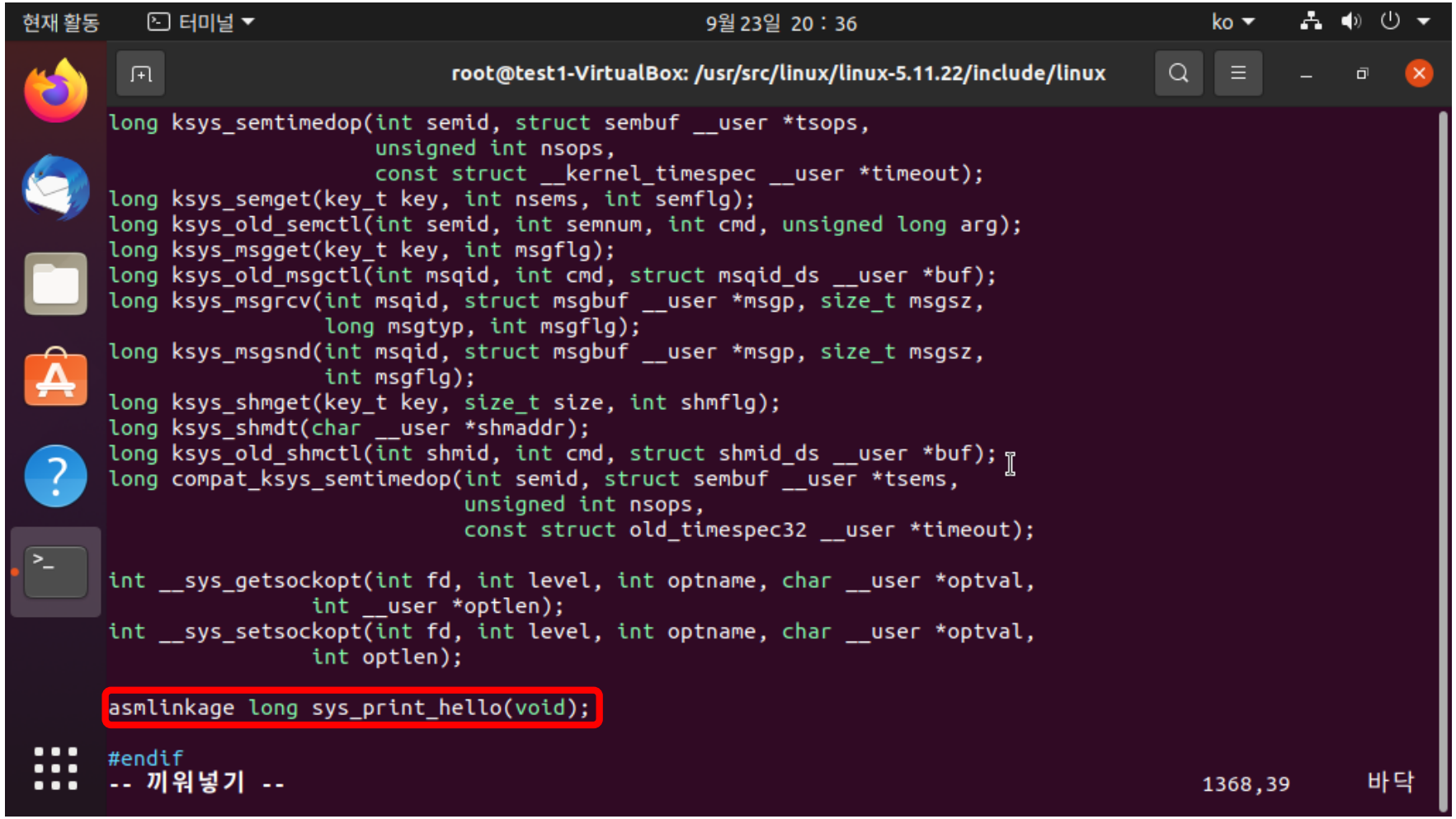
추가한 시스템 콜의 번호는 기억해야 함

1. 시스템 콜 추가 – 시스템 콜 헤더 파일에 등록



/usr/src/linux/linux-5.11.22/include/linux 디렉토리로 이동 후 syscalls.h를 vi 편집기로 열기

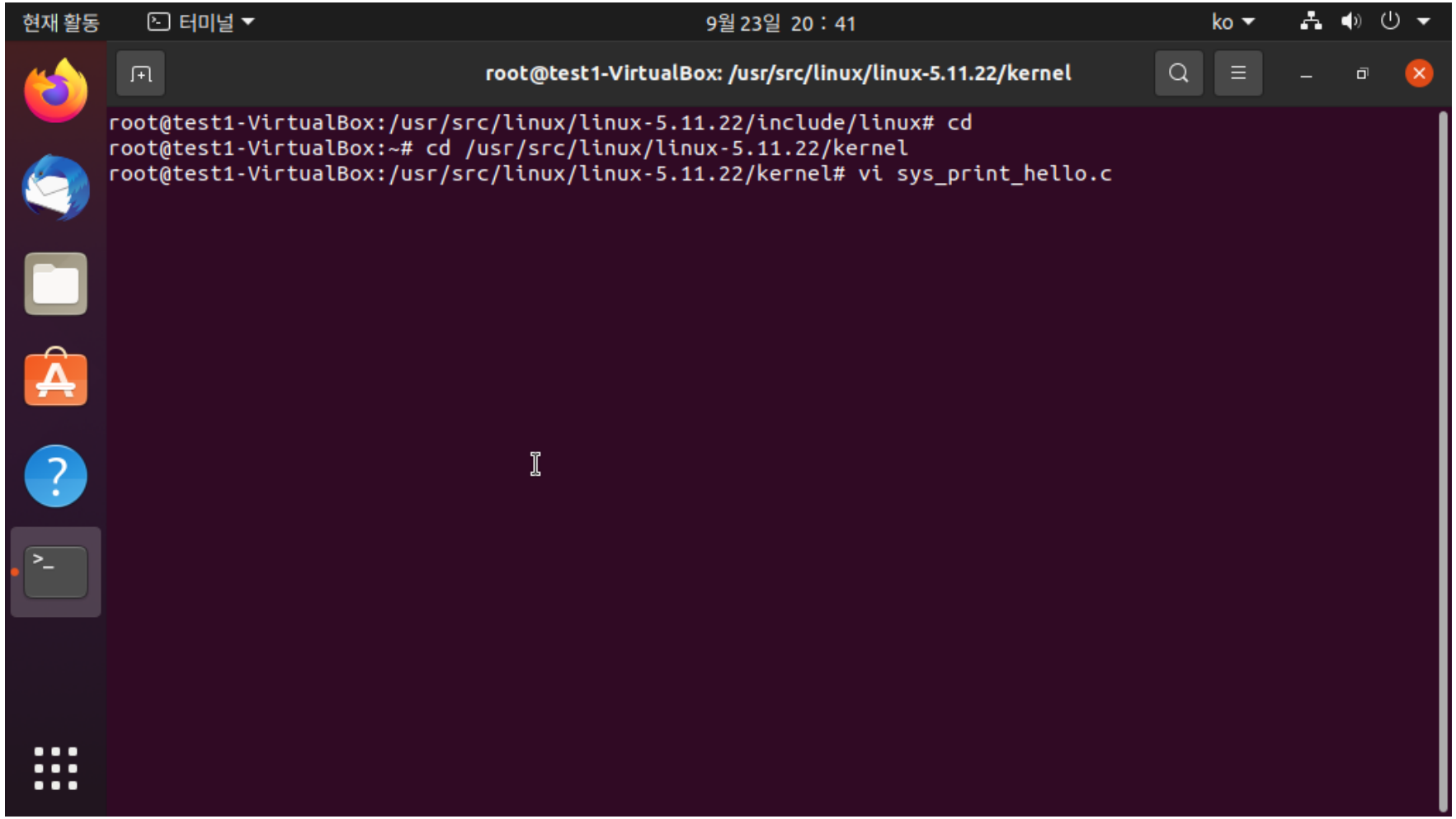
1. 시스템 콜 추가 – 시스템 콜 헤더 파일에 등록



함수의 프로토타입을 정의함

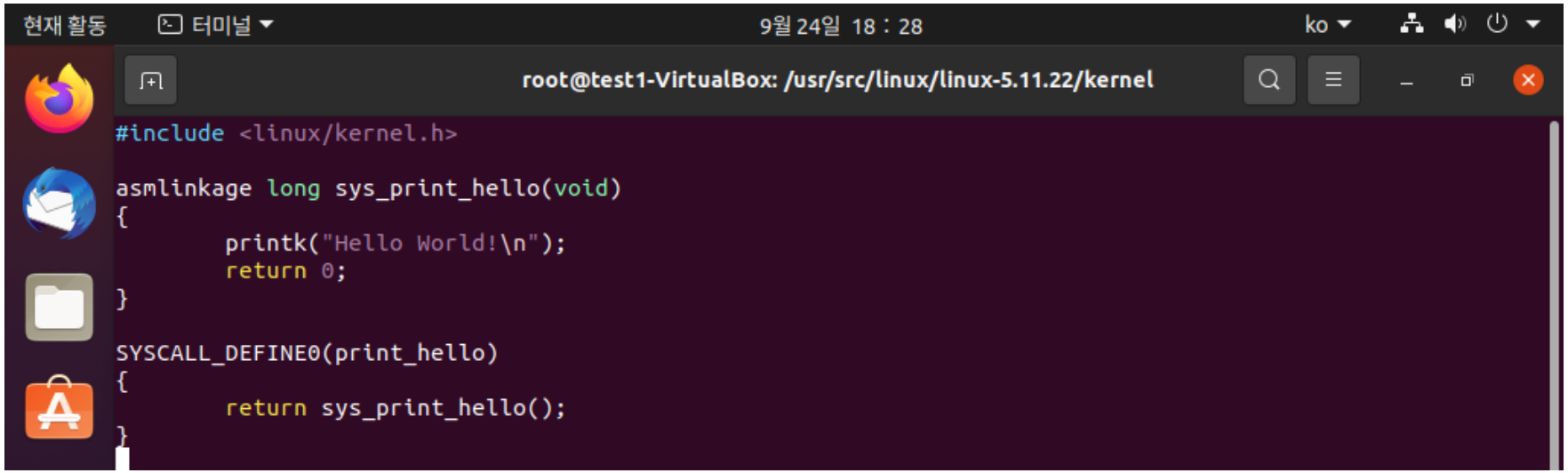
asmlinkage를 앞에 붙힘으로서 어셈블리 코드에서도 C 함수 호출이 가능해짐

1. 시스템 콜 추가 – 시스템 콜 함수 구현



`/usr/src/linux/linux-5.11.22/kernel` 디렉토리로 이동 후 추가할 시스템 콜의 구현 파일을 편집

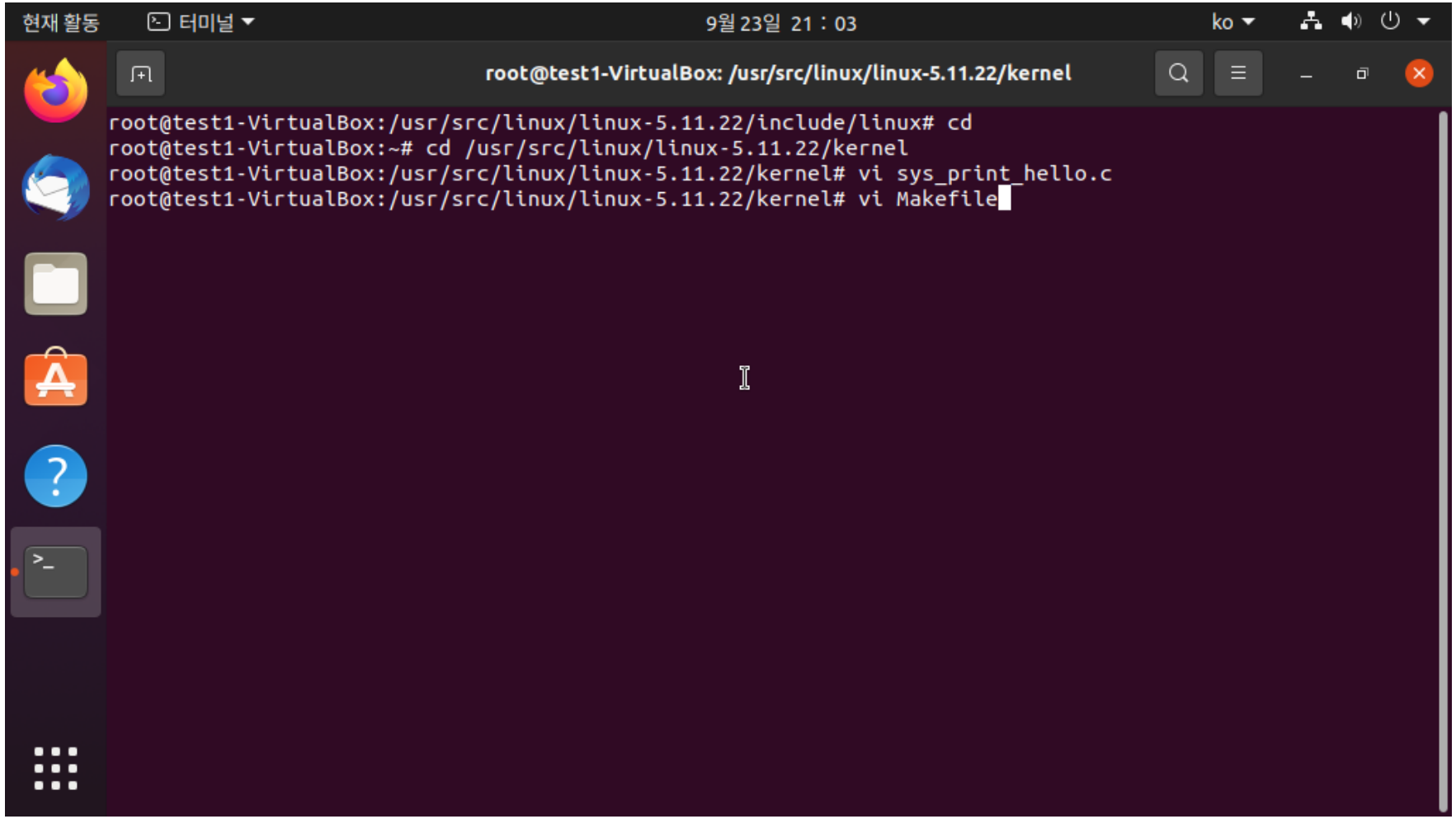
1. 시스템 콜 추가 – 시스템 콜 함수 구현



이때 파일명은 시스템 콜 이름과 달라도 되지만 파일 내부의 함수는 `sys_`시스템 콜 이름으로 작성해야 함

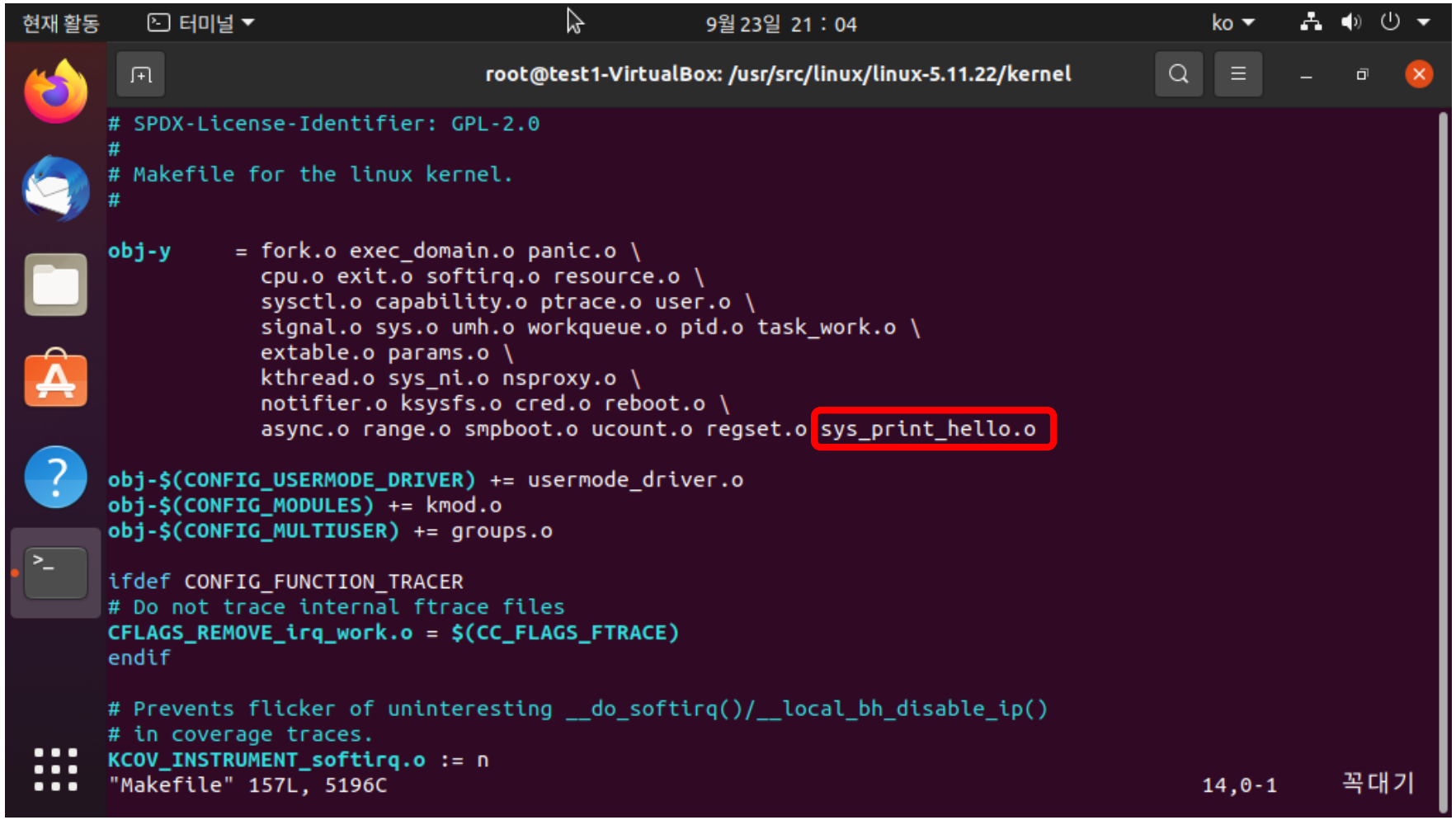
`printk` 명령어를 통해 커널 공간에서 메시지 출력이 되도록 작성함

1. 시스템 콜 추가 – Makefile에 등록



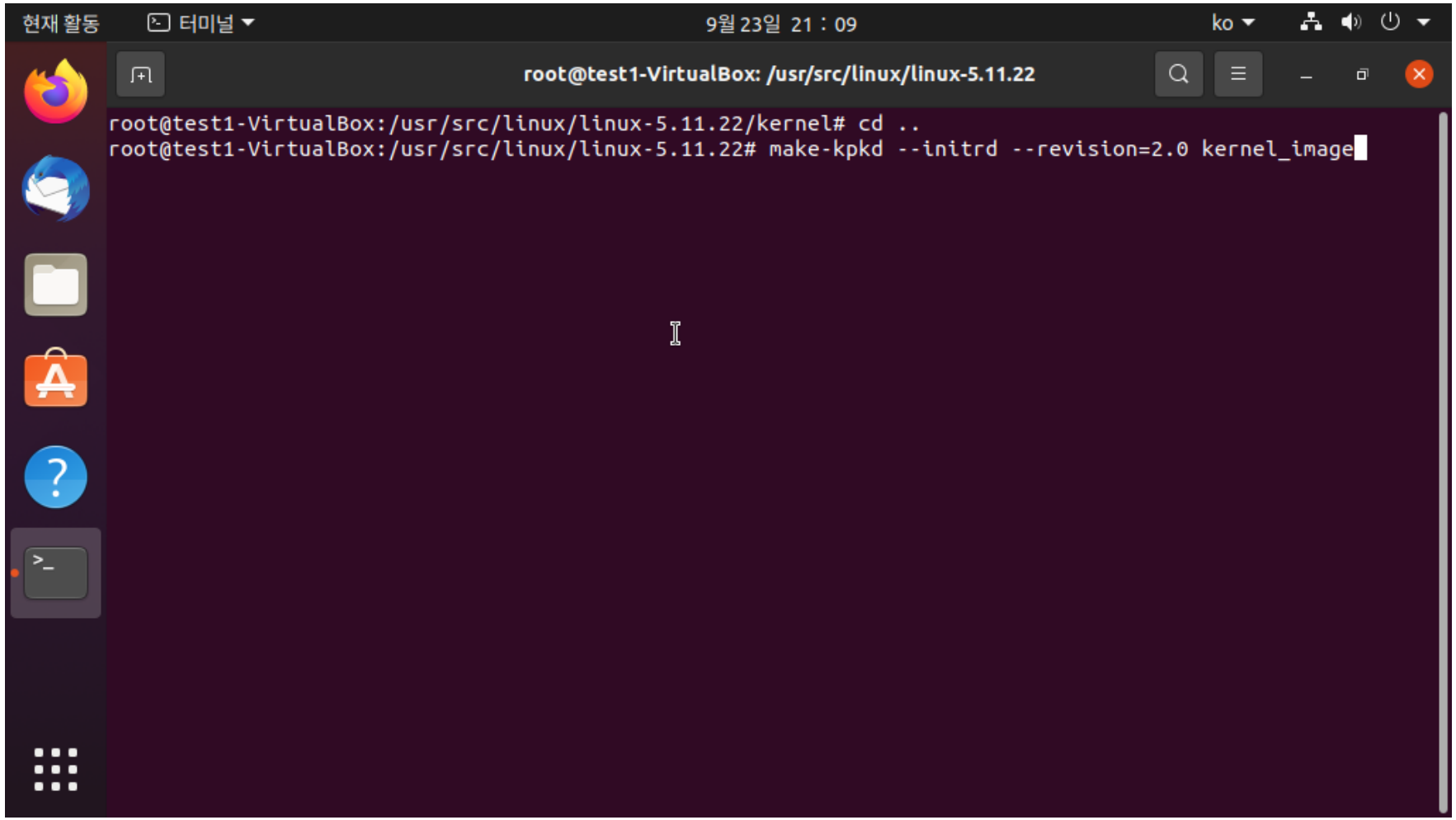
이전 단계와 동일한 디렉토리 내에서 `Makefile` 편집

1. 시스템 콜 추가 – Makefile에 등록



추가한 시스템 콜이 다른 시스템 콜과 함께 컴파일될 수 있도록 편집

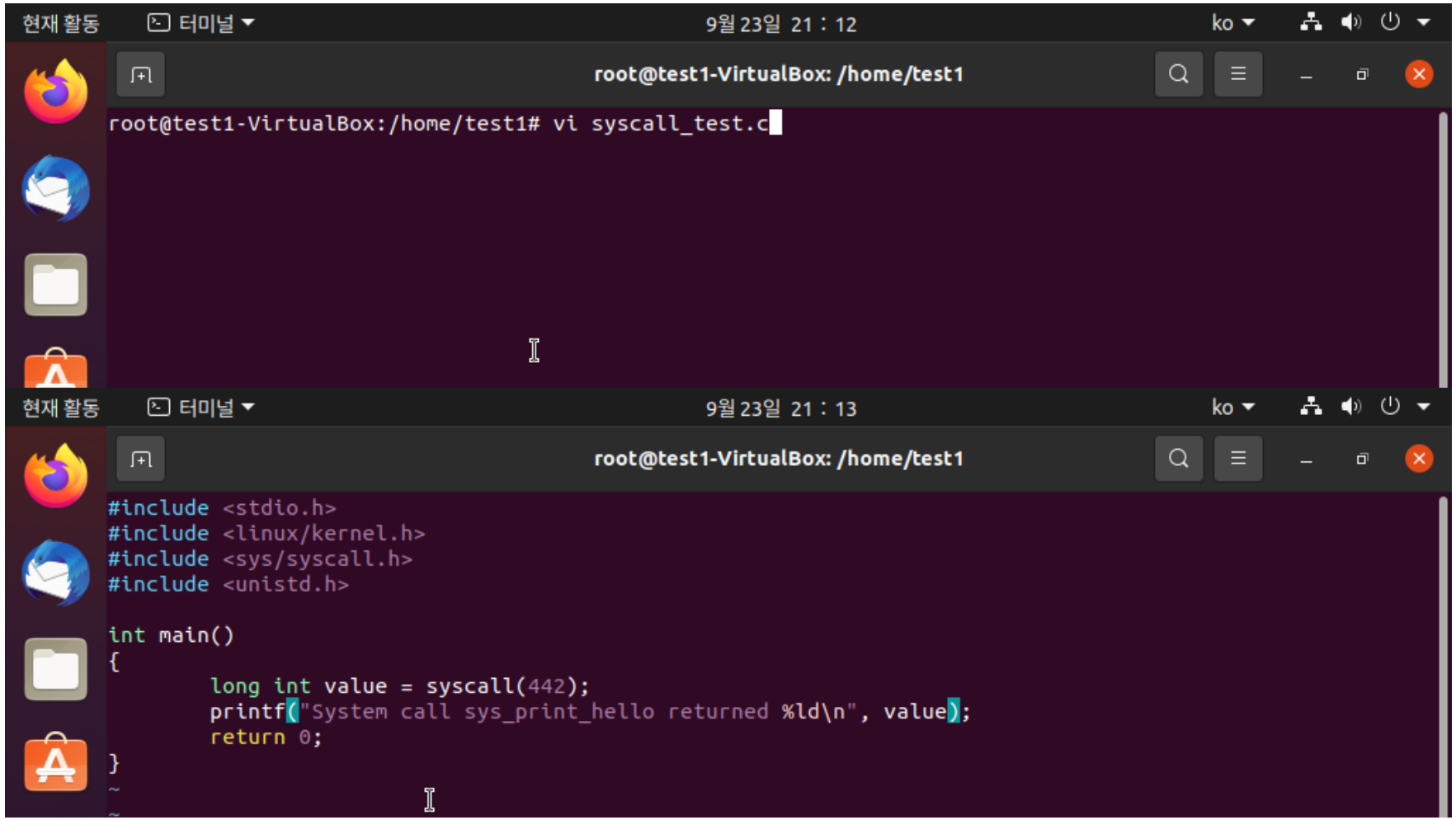
1. 시스템 콜 추가 – Makefile에 등록



커널 소스 디렉토리로 이동하여 새로 컴파일 후 재부팅 실시함

이때 `revision`의 값은 기존의 `1.0`과 구분을 할 수 있도록 다른 값을 입력하여 컴파일 실시(정수 값으로 설정)

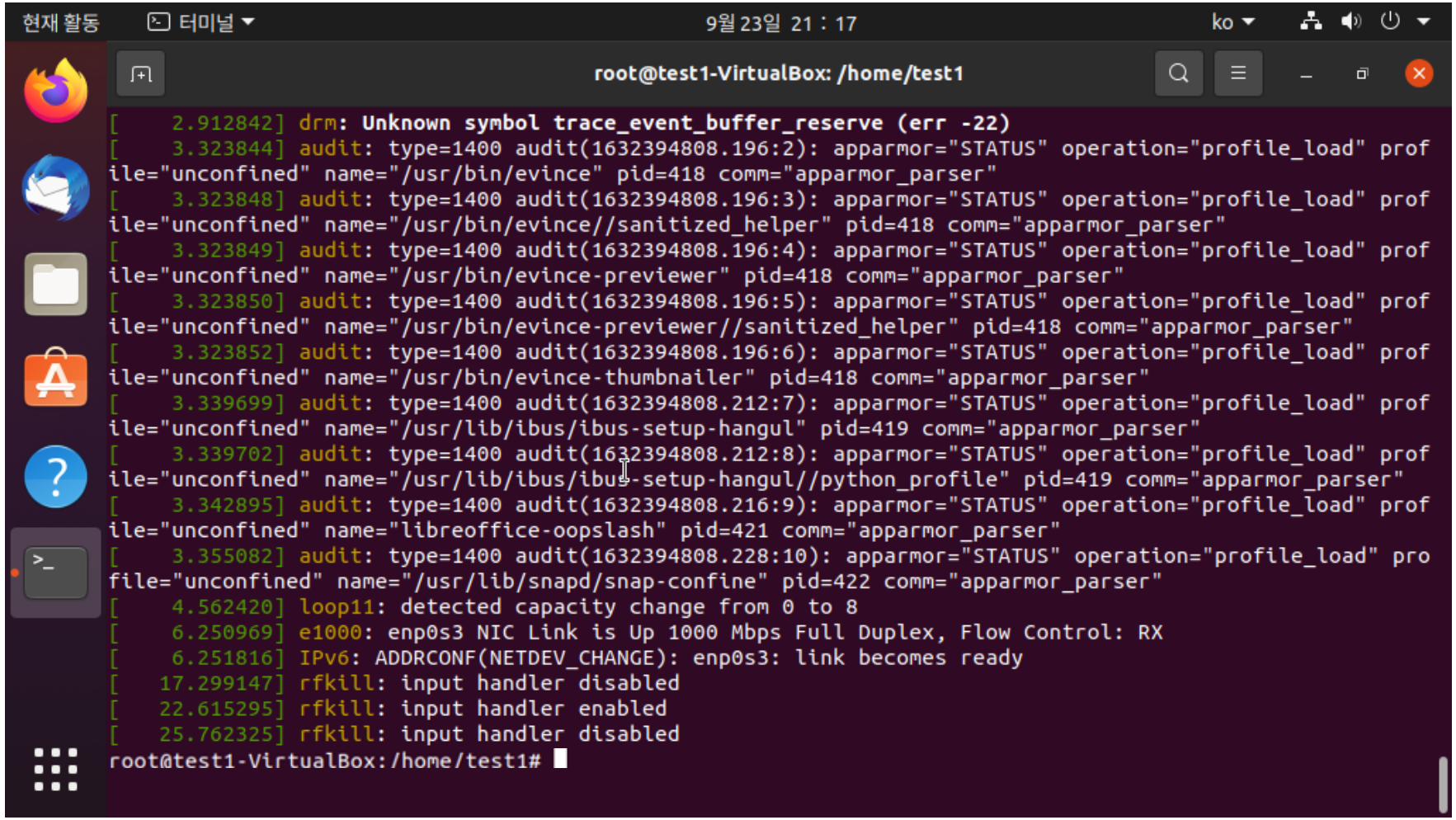
2. 추가된 시스템 콜 호출



재부팅 후 추가한 시스템 콜을 호출하는 함수를 생성함

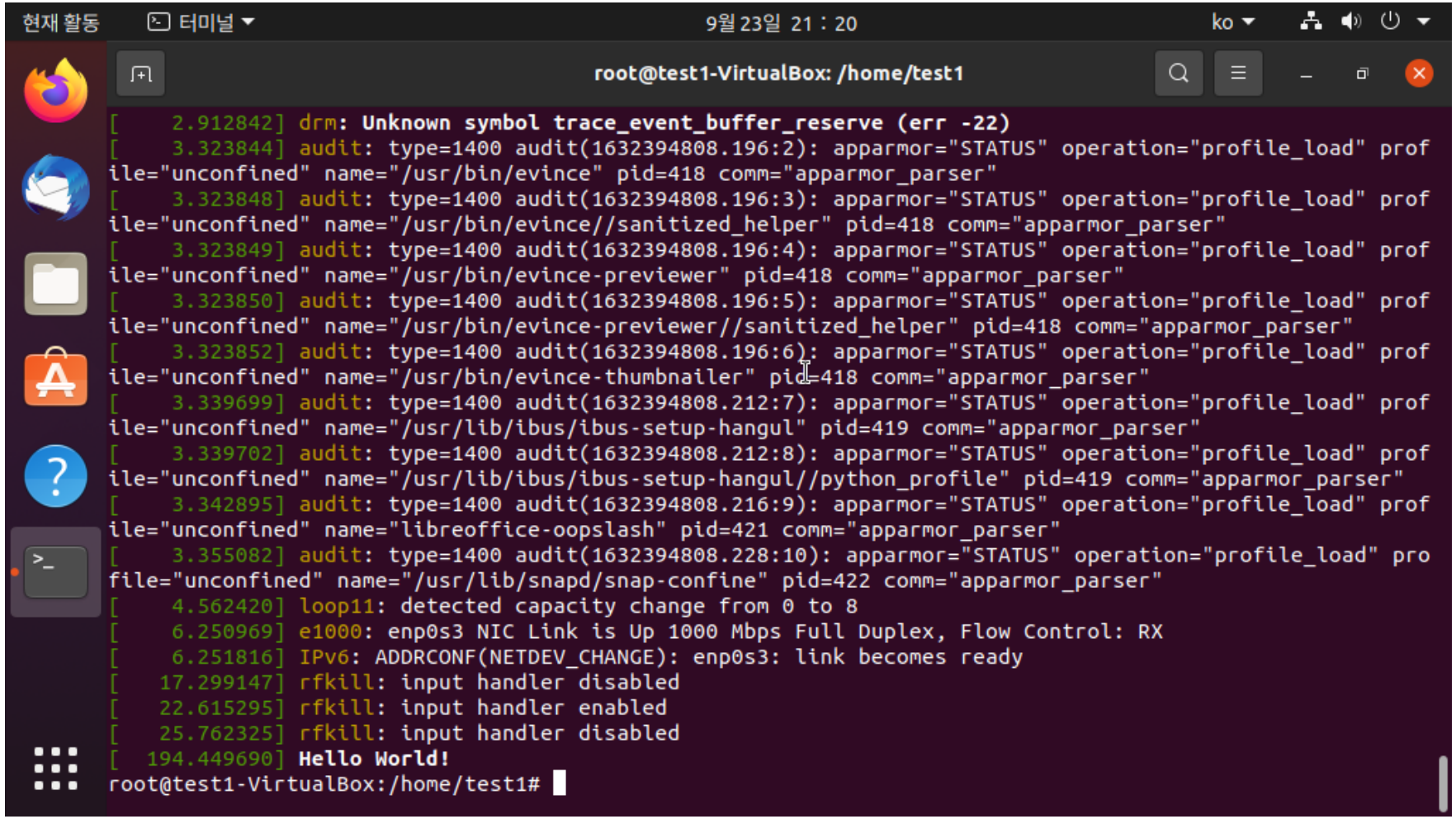
앞서 시스템 콜을 테이블에 등록할 때 생성한 번호를 사용하여 호출

2. 추가된 시스템 콜 호출



테스트 프로그램 실행 전 dmesg 명령어를 통해 커널의 로그를 출력함

2. 추가된 시스템 콜 호출



테스트 프로그램 실행 후 dmesg 명령어를 통해 커널 로그를 출력하면 Hello World! 가 출력된 것을 확인할 수 있음