운영체제 : 시스템콜

2015-03-29

2013210111 남세현

**개발 환경**

VMWare 위에 Fedora 14를 설치하고, 커널을 linux-2.6.35.6 으로 설치했습니다.

**인터럽트와 시스템 콜**

운영체제는 하드웨어부터 소프트웨어까지 컴퓨터 시스템의 모든 것을 관리하는 역할을 가지고 있습니다. 일반 사용자들에게 서비스를 제공하는 것 외에도, 소프트웨어를 개발하는 개발자에게 운영체제 서비스를 제공해야 합니다. 모든 소프트웨어에 컴퓨터 시스템의 관리 권한을 주면 큰 문제가 생길 수 있습니다. 보안, 안정성, 모든 것을 고려했을 때, 컴퓨터 과학자들은 그 권한을 운영체제 스스로 가지는 것으로 하기로 했습니다.

운영체제가 모든 권한을 가지고 있지만, 분명 우리의 소프트웨어는 시스템을 사용해야 합니다. 그래서 운영체제는 ‘시스템 콜’이란 것을 제공합니다.

**시스템 콜이란, 운영체제가 관리하는 리소스 등을 일반 소프트웨어도 사용할 수 있도록 간접적으로 실행권을 주는 것입니다.** 운영체제는 함수 형태로 시스템을 건들일 수 있는 권한을 소프트웨어에 줍니다. 우리가 흔히 알고있는, 메모리를 동적으로 할당하는 **malloc** 등의 함수가 바로 그것입니다.

운영체제가 제공하는 함수를 실행하기 위해서는 CPU가 우리의 소프트웨어를 돌리는 것을 멈추고, 운영체제가 작업할 수 있도록 해줘야 합니다. 우리의 소프트웨어가 돌아갈 때를 ‘**유저모드**’라고 하고, 운영체제(커널)이 돌아갈 때를 ‘**커널모드**’라고 합니다. **이 사이를 전환하는 것이 바로 인터럽트입니다.**

이번 숙제에서는 시스템 콜을 직접 만들어봄으로서, 시스템 콜의 실제 구현을 어떻게 하는지 알아보았습니다.

**목적 : Enqueue와 Dequeue 시스템 콜 만들기**

**# 코드 내용은 첨부파일에서 확인 가능합니다. 본 보고서에는 복사해놓지 않겠습니다.**

1. syscall\_table\_32.S

위 파일에 모든 시스템 콜 함수들의 이름, 심볼 정보가 들어있습니다. Enqueue와 Dequeue 함수 2개를 추가하였습니다.

1. unistd\_32.h

syscall\_table과 마찬가지로 시스템 콜 함수들의 번호가 들어있습니다. Enqueue와 Dequeue를 역시 추가합니다.

1. syscall.h

함수의 Prototype을 정의하는 헤더파일입니다. Enqueue와 Dequeue를 추가하는데, 반환타입 앞에 asmlinkage를 추가합니다.

Asmlinkage는 어셈블리 프로그램에서 사용할 수 있는 함수라는 의미입니다.

1. my\_queue\_syscall.c

실제 시스템 콜의 구현이 들어있습니다.

1. 전역변수로 int형 100사이즈를 가지는 Queue를 선언합니다.
2. Queue에 들어온 원소의 갯수를 저장하는 g\_queue\_count
3. Enqueue : 배열의 맨 마지막에 원소를 넣고, count를 증가
4. Dequeue : 배열의 맨 앞을 빼고, 전체적으로 앞으로 한번 이동 후 count 감소
5. Makefile

컴파일 시 해야하는 행동을 적어놓은 파일입니다. 위 my\_queue\_syscall.c를 컴파일 하도록 설정했습니다.

1. call\_queue.cpp

사용자에게 입력(E or D)를 받아 Enqueue 혹은 Dequeue를 하는 프로그램입니다.

**C++ 이기 때문에, GCC가 아닌 G++로 컴파일해야합니다.**

위 1, 2 항목에서 시스템 콜 번호를 338, 339 로 했기 때문에

syscall(338, value)로 Enqueue를, syscall(339)로 Dequeue를 할 수 있습니다.

Dequeue의 경우 결과물이 반환됩니다.

**결과**

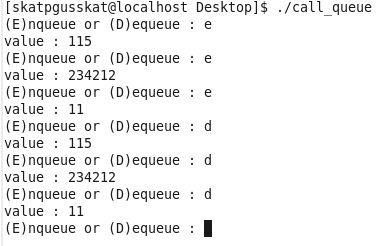


Figure - call\_queue 실행화면

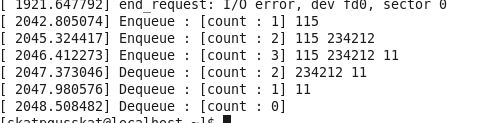


Figure - dmesg로 시스템 콜 확인

**숙제 수행 과정 중 발생한 문제점과 해결방법**

이미 여러 프로젝트를 많이 해보면서 프로그래밍과 컴퓨터에 익숙해져 있어서, 큰 어려움 없이 숙제를 했습니다.