

Open Source: How I actually did my personal history

김남형

LG Electronics

KOSSCON 2016

소개

- Namhyung Kim <namhyung@kernel.org>
- LG전자 선임연구원
 - SW Platform Lab, Open source contribution project
- 리눅스 커널 개발자
 - since 2010
 - more than 1000 patches merged
- 오픈소스 개발자
 - ufttrace, trace-cmd, git, ...

오픈소스?

- 그저 컴퓨터가 동작하는 원리가 궁금했을 뿐..
 - 상대적으로 늦게 시작했기에 궁금한 것이 많았음
- 현업에서 사용되는 OS 커널 소스
 - 궁금한 내용을 최대한 찾아볼 수 있음
- 공유정신

컴퓨터?

- 초등학교 특별활동

컴퓨터?

- 초등학교 특별활동
- 게임

games



FINAL FANTASY V

대학생활

- 전자전기 공학부

대학생활

- 전자전기 공학부
- 동아리 활동

대학생활

- 전자전기 공학부
- 동아리 활동
- 군대

대학생활

- 전자전기 공학부
- 동아리 활동
- 군대
- 공부?

Linux

- 시스템 프로그래밍 수업
- 점차 흥미를 느끼게 됨
- 틈틈이 책을보며 공부

회사생활

- 작은 start-up에서 시작
 - RTOS/Linux 커널 포팅
 - filesystem, device driver

회사생활

- 작은 start-up에서 시작
 - RTOS/Linux 커널 포팅
 - filesystem, device driver
- 몇차례 이직
 - 고용불안
 - Linux 관련 작업 힘들

번역

- Ubuntu translation
- GNOME translation

전환

- 허태준님의 KLDP talk
- 회사를 그만두고 리눅스에 전념해보자
 - 2년간 집에서 은둔
 - 온오프라인 스터디
 - 블로그 운영: <http://studyfoss.egloos.com>

Study

- Computer Architecture, Compiler, C-Language

Study

- Computer Architecture, Compiler, C-Language
- git, shell, system programming

Study

- Computer Architecture, Compiler, C-Language
- git, shell, system programming
- Linux kernel

First patch

- kprobes code cleanup
 - 블로그에 정리하다가 발견한 사소한 수정 사항
 - 한번 보내볼까?
- quality vs. quantity

English?

- I have no English..
- 메일링리스트 구독
- 무식하게(?) 용감해지자!

전업 오픈소스 개발자로..

- LG전자 SW Platform 연구소
 - open source contribution project
 - 매우 유능한 팀원들
 - 사내 개발자 지원 및 문화 전파

Linux kernel

- 다양한 (100+) subsystem 존재
 - scheduler, memory, filesystem, network, drivers, ...
 - 각각의 maintainer들이 관리
- 주요 subsystem들에 대해 분석/기여
 - sched, mm, ext3, block, raid(md)
 - 현재는 주로 perf/tracing

Community



Linux perf tools

- Performance monitoring and analysis
 - H/W와 커널이 제공하는 기능 활용
 - 시스템 전체의 성능 정보 제공
 - 커널과 사용자 공간 모두 분석
 - 매우 활발한 개발 활동

Ftrace

- Function level tracing
 - 함수 단위의 성능 분석 가능
 - tracepoint/kprobes 이벤트 분석
 - toolchain의 도움이 필요

Uftrace

- Ftrace for userspace
 - function graph tracer의 이식판
 - C/C++ 프로그램의 실행 성능 분석
 - x86_64, ARM (v7) 지원 (Linux only)

Uftrace example

```

namhyung@danjae:uftrace$ sudo uftrace -k -T '.*@kernel,color=red' tests/t-signal
[sudo] namhyung의 암호:
# DURATION      TID      FUNCTION
    1.598 us [18718] | main() {
    0.639 us [18718] |     foo();
    4.385 us [18718] |     signal() {
    1.029 us [18718] |         sys_rt_sigaction();
    0.350 us [18718] |     } /* signal */
    0.339 us [18718] |     raise() {
    4.643 us [18718] |         sys_rt_sigprocmask();
    0.570 us [18718] |         sys_getpid();
    5.016 us [18718] |         sys_gettid();
    0.178 us [18718] |         sys_tgkill();
    0.958 us [18718] |         sys_rt_sigprocmask();
    2.542 us [18718] |         exit_to_usermode_loop();
    24.709 us [18718] |         sighandler() {
    0.161 us [18718] |             bar();
    47.653 us [18718] |         } /* sighandler */
    0.161 us [18718] |         do_syscall_64();
    0.161 us [18718] |     } /* raise */
    0.161 us [18718] |     foo();
    47.653 us [18718] | } /* main */

```

What's next?

- I don't know..
 - 아직도 궁금한 것은 너무도 많고
 - 흥미로운 기술들은 계속 생겨납니다. :)

Q & A

Thanks!