




IAMROOT.ORG

2012년 신규 멤버 OT

발표 : 백창우



반갑습니다.

- 반갑습니다.
 - 환영합니다.
 - 어쩌다가 이 험난한 길을 선택하셨나요?
 - 잘 오셨습니다.
 - 복 받으실 겁니다.
- 

저희 스터디를 소개 하겠습니다.

- 2003년 2월에 아무 대책 없는 사람들 아홉명이서 리눅스 커널 소스를 분석하려고 모였습니다.
- 정말 맨땅에 헤딩 많이 했습니다.
- 하도 많이 해서 다섯명이나 장렬히 전사하고 네명만 남았습니다.
- 2004년 5월에 네명이서 끝까지 커널 소스 분석을 완료했습니다.
- 좋다고 파티하고 각자의 길로 갔습니다.
- 2005년 1월에 제가 회사에 들어갔습니다.
- 이런~ x86은 안하고 ARM만 하네요.

저희 스터디를 소개 하겠습니다.

- 기존 멤버들과 새로운 사람들로 해서 2005년 7월에 다시 모였습니다.
- 이번에는 ARM 커널을 분석했습니다.
- 2006년 6월에 못나온다고 전화 돌리는게 귀찮아져서 게시판 하나 만들었습니다.
- 그것이 지금 iamroot.org 사이트가 되었습니다.
- 2006년 12월에 ARM 커널 분석을 완료했습니다.
- 그대로 끝낼까 했는데 너무 아쉬웠습니다.
- 너무나 좋은 모임이었고,
- 인생에 길잡이가 되어준 모임이었습니다.
- 다른 누군가도 길을 찾지 못해 헤메고 있을거라 **생각했습니다.**

저희 스터디를 소개 하겠습니다.

- 뜻을 모아 새로운 분들을 모집하고 저희가 스터디 했던 방식을 알려드리고 길잡이가 되어 드렸습니다.
- 모두 미쳐서 공부하기 시작했습니다.
- 다들 훌륭하신 분들었고, 훌륭하게 되셨습니다.
- 그래서 한해 더 했습니다.
- 그리고 또 한해 더 했습니다.
- 그렇게 계속하다 이제는 의무감 때문이라도 그만 둘 수 없게 되어 버렸습니다.
- 덕분에 컴퓨팅 분야 최고의 하드코어 스터디 그룹
- 이라는 낙인도 찍혔습니다.



저희 스터디를 소개 하겠습니다.

- 이것이 그간 스터디의 역사 입니다.
- 어떤 곳인지 감 잡히시나요?



나중에 이런 사람이 됩니다.

1. linux kernel에 대해서 아주 깊이 알게 됩니다.
(linux kernel의 이해 책의 틀린 부분을 찾을 정도, 머리로 부팅할 정도)
2. 단순히 kernel을 아는것이 아니라 원하는 대로 바꾸고 고칠수 있게 됩니다.
(device driver 정도는 손쉽게 만들 수 있을 정도, embedded는 취미삼아)
3. OS에 대한 이해가 깊어집니다.
(간단한 OS 정도는 심심풀이 삼아 만들어 볼수 있을정도)
4. 궁극의 code 최적화 기술을 배울수 있습니다.
(architecture 최적화, 메모리 최적화 , C의 어두운 면, 돈주고도 못배우는 각종 사파의 기술들 등등등)
5. 프로그래밍 스킬이 향상됩니다.
(설계, 알고리즘, compiler, linker, assembly는 덤으로)



나중에 이런 사람이 됩니다.

6. 대규모 프로젝트를 이해하고 분석하는 능력 및 실제로 개발하는 능력이 향상됩니다.
(당연하겠죠.)
7. Compiler에 대해서 깊이 이해하게 됩니다.
(C 소스를 머리로 컴파일해서 assembly로 보는 정도)
8. Architecture에 대해서 이해가 깊어집니다.
(토론중에 CISC, RISC, EISC, VLIW, Superscalar, RP 등등의 이야기가 절로 나옵니다.)
9. Architecture 및 여러 컴퓨팅 기술의 흐름을 읽을수 있게 됩니다.
(가만히 앉아서 최신 커널 소스를 보는것 만으로도 그냥 저절로 알수 있습니다.)
10. 마지막으로 각 분야에 전문가 분과 교류할 수 있습니다.
(주위에서 geek이라 불릴만한 분들이 수두룩합니다.)
11. 가끔 미친놈 소리도 듣지만, 먹고 살 걱정은 안해도 됩니다.

어떻게 진행할까요?

커널 소스 분석을 예를 들어 보겠습니다.

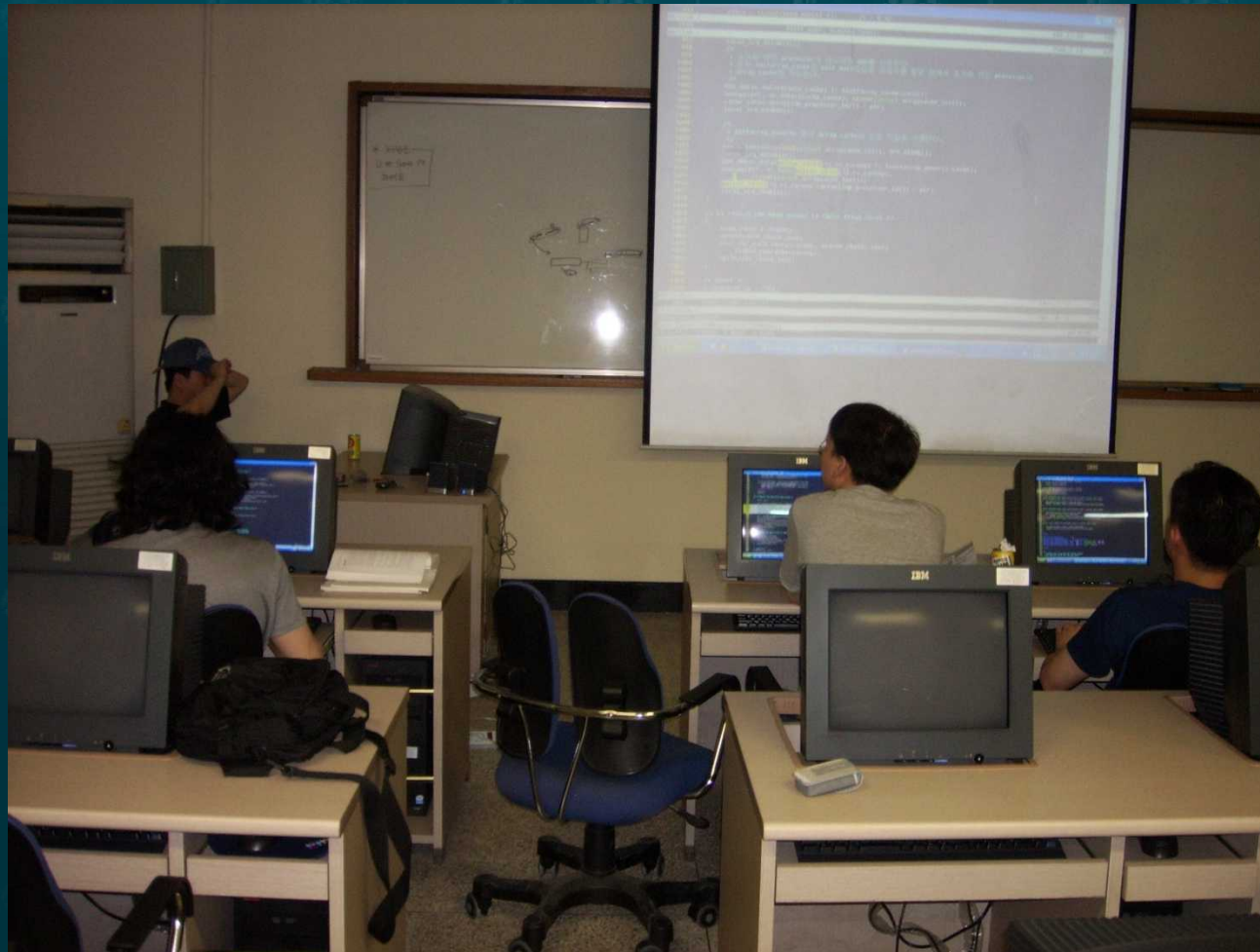
1. 스터디 그룹 결성 (IAMROOT.ORG) 및 장소 선정
2. 각 그룹별 스터디 게시판 생성 (홈페이지 관리자)
3. 분석 Processor 선정 (최신 Architecture)
4. 커널 이론 학습 (쉬운 책으로 선정할 것, OB들에게 문의하세요.)
5. Processor Architecture 학습 (책은 OB들에게 문의하세요.)
6. 해당 Processor Architecture Assembly 학습 (Boot loader 또는 Kernel에 있는 코드로 1일~2일간 단체 학습, 책 따로 보지 마세요.)
7. 최신 버전 커널 소스를 받아 SCM 서버에 넣으세요.
8. 전원 인가 시부터 shell이 뜰때까지 kernel 소스를 분석하시면 됩니다.
 - ◆ 빔 프로젝트를 켜놓고 한 명이 vim + ctags + cscope로 커널 소스를 드라이빙하면서 각 라인별로 토론하며 토론 결과를 주석 남김 (주석은 가능한 꼼꼼하고, 컴파일 가능하게 남길 것)
 - ◆ x86은 lilo 분석, ARM은 그냥 kernel부터 분석 (절대 grub나 u-boot 분석하지 말 것)
9. Shell이 실행되기 직전까지 분석이 끝났으면 끝낼질 다른 sub system을 분석할지 결정하시면 됩니다.

이론 학습은 이렇게 합니다.

(커널, 프로세서 아키텍처 분석, 책 스터디)



실제 소스 분석은 이렇게 해요.



반드시 지켜야 할 스터디 규칙이 있습니다.

- 저희 스터디 그룹은 하루 아침에 이루어지지 않았습니다.
- 10년이란 세월 동안 스터디에 대한 많은 경험과 노하우를 가지고 있습니다.
- 그간의 경험을 토대로 반드시 지켜야 할 스터디 규칙을 만들었습니다.
- 이대로 하지 않으면 망합니다. 장담합니다.

반드시 지켜야 할 규칙 1

절대 **상호 존칭** 사용해주세요.

형님, 동생 하는 순간 스타디는 끝납니다.



반드시 지켜야 할 규칙 2

절대 개인에게 **부담주지** 마세요.

- 개인에게 부담을 주는 순간 재미 없는 스터디가 됩니다.
- 개인에게 부담을 주는 예
 - 이론 스터디 할 때 책 분량 각자 나누어서 나중에 발표하게 하는 행위
 - 다른 대안이 있는 상황에서 굳이 어려운 책(예:원서)을 고집하는 행위
 - 각자 무언가를 조사해서 발표 시키는 행위
 - 한 사람만 계속 소스 드라이빙 시키는 행위
 - 기타 부담 될 수 있는 모든 행위들



반드시 지켜야 할 규칙 2

가장 좋은 스터디는 아무런 준비 없이 와도 부담 없이 재미있고, 무언가를 얻어갈 수 있는 곳입니다.

그래서 매주 스터디 시간이 기다려 지게끔 되어야 합니다.

절대 부담을 만들지 마세요.



반드시 지켜야 할 규칙 3

스터디 시간 임의로 변경 마세요.

- 현재 오후 3시 ~ 밤 10시 30분 이상 하는 스터디 시간은 그간의 스터디를 진행해오면서 가장 부담 없는 시간으로 저절로 최적화된 시간입니다.
- 스터디 시간을 바꾸는 순간 대부분에게 시간적 부담을 줄 수 있는 스터디가 되게 됩니다.



반드시 지켜야 할 규칙 4

스터디 초반에는 절대로 빠지지 마세요.

- 초반 3개월간은 배움에 있어 무척 중요한 시간이고, 진도가 매우 빠르게 나가는 시간입니다.
- 초반 3개월에 불참하는 순간 따라가지 못하는 사태가 발생합니다.
- 그러니 되도록 초반 3개월은 절대 빠지지 마세요.

기타 알아두어야 할 것들

스터디 정신을 해치는 행위를 하지 마세요.

- 아마추어리즘을 해치는 모든 행위가 이에 해당 됩니다.
- 독재식 운영 (안됩니다.)
- 이기적인 스터디 운영 (안됩니다.)
- 목적이 순수하지 못한 사람 (절대 안됩니다.)
- 기타 스터디의 순수성을 해치는 모든 행위들 (막으셔야 합니다.)

기타 알아두어야 할 것들

- 모르겠으면 OB들에게 물어보세요.
(진행방향, 기술, 스터디 문제)
- 상호 존중하는 마음에서 말을 많이 하세요. 그게 도움이 많이 됩니다.
- 선불리 친해지려 하지 마세요. 1년이라는 시간 동안 저절로 친해집니다.
- 기록을 꼼꼼하게 하세요. 남는 것은 기록과 경험과, 사람입니다.

기타 알아두어야 할 것들

- 지각하지 마세요.
- 지각하는 사람 기다려 주지 마세요.
- 사정이 있어 한 주 빠진 사람 너무 배려하지 마세요.
- 학생이라면 예/복습 철저하게 하세요. 그대들은 이러한 경험을 가지게 된 것에 대해서 행운아들입니다. 그대들에게는 이 모임이 인생에 있어 길잡이가 되고 전환점이 될 것 입니다. 미친듯이 공부하세요.

강의실 구하는 법

- 공개 소프트웨어 역량 프라자
 - <http://www.nipa.kr/nuri/conferenceRoom.it>
 - 박준석 책임 : jspark 골뱅 nipa.kr
- 소프트웨어 마애스트로
 - 20인실, 5인실
 - 백창우 : bckddn 골뱅 gmail.com



OB님들의 말씀을 들어볼까요?

1. 자기소개 (이름 / 소속 / 하는일,)
2. 어떤 스터디들을 언제 했는가?
3. 스터디때 경험담
4. 스터디가 자신에게 어떤 영향을 미쳤나?
5. YB들에게 조언
6. 하고 싶은 말

내부 프로젝트 및 좋은 프로젝트 소개 좀 할께요.

1. 지속 발전 가능한 커널 책 개발
2. 고성능 가상머신 규격 및 기술 개발
 1. suhyun.kim 골뱅 imrc.kist.re.kr
3. 진료정보 공동활용을 위한 의료 정보 표준 프레임워크 개발
 1. bckddn 골뱅 gmail.com



그룹별 분할

1. 그룹별로 자기 소개 및 상호 환영의 시간을 가지세요.
2. 다음주 스터디할 장소를 선정하세요.
3. 그룹별로 회식하러 가세요.
 - MPSoC, Xen, Linux x86, Android, OS
 - Linux ARM A, Linux ARM B, Linux ARM C

박창원 박창현
최원재 정찬홍
오경범 이호섭
박세영 이용덕
최정기 이형준
조정흙 유정환
최문성 김태균
정영훈 윤석재
윤성웅 임재욱
권혁주 김익환
이종인 이한울
김성진 이종보
김용현 김건용
여태일 최원제
김효중 문용석
이종혁
임성근

A팀

안종현 박민식
김이슬 유훈찬
이동형 지승화
성한순 류주영
전원찬 박호열
김홍균 서치홍
안태진 유희재
한원진 김우철
정재훈 문계련
주재숙 김현철
김건우 임동함
한두경 임준혁
이승희 송주선
전형규 정지훈
정현아 정수연
전유진
최지영

B팀

마지호 박인이
이우성 이상정
강희봉 이선훈
박찬호 박상민
송은두 이영진
박희민 정지훈
김준수 정인호
김홍제 구본규
홍기성 유시광
어선택 지순영
문주성 박재상
노시현 황순찬
장인태 장태영
한기홍 이홍우
윤연식 신재진
곽근호

C팀