

과제 1 안내

컴퓨터공학과 장대희



경희대학교
KYUNG HEE UNIVERSITY



PWNJAB
@KYUNGHEE UNIVERSITY

Linux 크래시 실습

■ 개요

- 시스템콜 등 커널에서 실행될 코드를 분석하고 크래시를 유발하는 코드 삽입
- 수정된 커널소스를 새로 컴파일하여 QEMU 에뮬레이터로 부팅
- 부팅중 또는 부팅후 특정 명령 수행시 삽입된 크래시 유발코드가 실행

■ 크래시 유발코드 (C 언어)

```
long int* boom = (long int*)0xcafebabe;
```

```
*boom = (본인의 학번을 16진수로 변환 후 0x00FFFFFF 와 AND 한 값);
```

과제 진행절차

■ 아래 순서로 진행

- ✓ 1. Ubuntu 환경준비 (20.04, 22.04 권장)
- ✓ 2. 리눅스 커널소스 다운로드 <https://www.kernel.org/>
 - ✓ 커널버전: 6.2.8
- ✓ 3. QEMU-ARM 에뮬레이터 설치:
 - ✓ `sudo apt install qemu-system-arm`
- ✓ 4. ARM 크로스컴파일러 및 개발보조도구 설치:
 - ✓ `sudo apt install gcc-arm-linux-gnueabi make flex bison`
- ✓ 5. ARM 리눅스 커널컴파일 (아래 링크 등 참고)
 - ✓ https://jasonblog.github.io/note/arm_emulation/compiling_linux_kernel_for_qemu_arm_emulator.html
 - ✓ <https://github.com/surajx/qemu-arm-linux>
 - ✓ QEMU 지원 config 로 (versatilepb 등) 컴파일 해야함 유의.
 - ✓ 최신커널은 DTB파일 추가해야함 유의.
 - ✓ <https://community.arm.com/oss-platforms/w/docs/525/device-tree>

과제 진행절차

■ 아래 순서로 진행

- ✓ 6. BusyBox 소스코드 다운후 ARM 크로스컴파일 (아래 링크 등 참고)
 - ✓ https://jasonblog.github.io/note/arm_emulation/busybox_for_arm_on_qemu.html
 - ✓ <https://github.com/surajx/qemu-arm-linux>
 - ✓ static 컴파일 해야함 유의.
- ✓ 7. 에뮬레이터 (qemu-system-arm) 을 이용해서 BusyBox 와 함께 수정 전 OS 구동 후 쉘 획득
 - ✓ 1단계 완성
- ✓ 8. ARM 리눅스 커널 소스코드 분석 후 커널소스코드 수정.
 - ✓ 크래시 유발코드를 삽입할 부분 찾기
- ✓ 9. 수정된 ARM 리눅스 커널을 QEMU 로 부팅시킨 후 커널 크래시 관찰
 - ✓ 2단계 최종 완성

제출물

❖ 다음 파일들을 수업조교에게 email 로 제출 (yuri0329@khu.ac.kr)

- 1단계, 2단계 각각 완료하면 스크린샷
- zImage.back (커널을 수정하기 전 쉘 획득까지 정상실행되는 커널이미지)
- zImage (커널을 수정하여 컴파일한 커널이미지)
- 수정한 커널소스 파일
- rootfs.img.gz (gzip된 BusyBox 파일시스템)
- dtb 파일 (Device Tree Blob)
- start.sh (QEMU 구동 스크립트)
- 1단계/2단계 가 정상적으로 재현/검증 되어야 제출이 인정됨.

❖ 아래 스크린샷 참고

```
daehee@none:~$ ls
a a.c a-dynamic a-static lab os-homework1 snap
daehee@none:~$ cd os-homework1/
daehee@none:~/os-homework1$ ls
busybox-1.36.0 linux-6.2.8 linux-6.2.8.tar
busybox-1.36.0.tar.bz2 linux-6.2.8.patch stage
daehee@none:~/os-homework1$ cd stage
daehee@none:~/os-homework1/stage$ ls
rootfs.img rootfs.img.gz start.sh versatile-pb.dtb zImage zImage.back
daehee@none:~/os-homework1/stage$ cat start.sh
#!/bin/sh
qemu-system-arm -M versatilepb -m 128M -kernel zImage -dtb versatile-pb.dtb -initrd rootfs
.img.gz -append "root=/dev/ram console=ttyAMA0 rdinit=/bin/sh init=/bin/sh" -nographic
daehee@none:~/os-homework1/stage$
```

결과화면

❖ 과제 1단계 정상 수행시 아래와 유사한 화면을 확인해야함

- QEMU 로 정상 부팅후 BusyBox 마운트한뒤 쉘 획득

```
daehee@none:~/os-homework1/stage$ ./start.sh
pulseaudio: set_sink input volume() failed
pulseaudio: Reason: Invalid argument
pulseaudio: set_sink input mute() failed
pulseaudio: Reason: Invalid argument
vpb_sic_write: Bad register offset 0x2c
Booting Linux on physical CPU 0x0
Linux version 6.2.8 (daehee@none) (arm-linux-gnueabi-gcc (Ubuntu 11.3.0-1ubuntu1) 11.3.0)
r 28 01:37:10 KST 2023
CPU: ARM926EJ-S [41069265] revision 5 (ARMv5TEJ), cr=00093177
CPU: VIVT data cache, VIVT instruction cache
OF: fdt: Machine model: ARM Versatile PB
Memory policy: Data cache writeback
Zone ranges:
  Normal [mem 0x0000000000000000-0x0000000007ffffff]
Movable zone start for each node
Early memory node ranges
  node 0: [mem 0x0000000000000000-0x0000000007ffffff]
Initmem setup node 0 [mem 0x0000000000000000-0x0000000007ffffff]
Built 1 zonelists, mobility grouping on. Total pages: 32512
Kernel command line: root=/dev/ram console=ttyAMA0 rdinit=/bin/sh init=/bin/sh
Dentry cache hash table entries: 16384 (order: 4, 65536 bytes, linear)
Inode-cache hash table entries: 8192 (order: 3, 32768 bytes, linear)
mem auto-init: stack:off, heap alloc:off, heap free:off
Memory: 121816K/131072K available (4944K kernel code, 185K rwddata, 1260K rodata)
NR IRQs: 16, nr_irqs: 16, preallocated_irqs: 16
VTC @(<ptrval>): id 0x00041190, vendor 0x41
```

```
leds-syscon 10000008.6.led: registered LED (null)
leds-syscon 10000008.7.led: registered LED (null)
ledtrig-cpu: registered to indicate activity on CPUs
NET: Registered PF_PACKET protocol family
input: AT Raw Set 2 keyboard as /devices/platform/amba/amba:fpga/10006000.kmi/se
Freeing unused kernel image (initmem) memory: 200K
Kernel memory protection not selected by kernel config.
Run /bin/sh as init process
/bin/sh: can't access tty; job control turned off
~ # input: ImExPS/2 Generic Explorer Mouse as /devices/platform/amba/amba:fpga/1
~ #
~ # id
uid=0 gid=0
~ # ls -al
total 4
drwxrwxr-x 10 1000 1000 0 Mar 28 04:23 .
drwxrwxr-x 10 1000 1000 0 Mar 28 04:23 ..
-rw-r----- 1 0 0 10 Mar 28 04:23 .ash_history
drwxrwxr-x 2 1000 1000 0 Mar 27 14:27 bin
drwxrwxr-x 2 1000 1000 0 Mar 27 12:14 dev
drwxrwxr-x 3 1000 1000 0 Mar 27 12:14 etc
lrwxrwxrwx 1 1000 1000 11 Mar 27 14:27 linuxrc -> bin/busybox
drwxrwxr-x 2 1000 1000 0 Mar 27 12:14 proc
drwx-r----- 2 0 0 0 Mar 27 12:33 root
drwxrwxr-x 2 1000 1000 0 Mar 27 14:27/sbin
drwxrwxr-x 2 1000 1000 0 Mar 27 12:14 sys
drwxrwxr-x 4 1000 1000 0 Mar 27 12:14/usr
~ #
```


결과화면

❖ 과제 2단계 정상 수행시 아래와 유사한 화면을 확인해야함

- 메모리 접근오류로 커널패닉 확인

```
daehee@none: ~/os-homework1/stage
Register r11 information: NULL pointer
Register r12 information: non-paged memory
Process kworker/u2:0 (pid: 6, stack limit = 0x(ptrval))
Stack: (0xc8825cd4 to 0xc8826000)
5cc0: c00f5b54 00000000 00000000
5ce0: c8825ce4 c002a5a8 c0674b58 c0c17f68 00000040 00000241 00000001 00000000
5d00: 000081ed c0c16340 00000041 c0823740 00000041 c8825d14 c8825d14 c773c866
5d20: c00f6648 00000001 c8825df8 00000001 00000000 00000000 00000000 c4000000
5d40: c096c800 c00f6988 c0803db0 c0c17f68 8d203826 00000003 c08d301b c064a88c
5d60: 00000000 c773c866 c0c16340 00000301 00000000 00000000 00000000 0000001a
5d80: 00000000 00000000 00000000 00000000 c8825d94 c8825e2b 7fffffff 00000000
5da0: c8825e2b 00000007 c0595418 c4000000 c096c800 c04c4730 2160ec00 00000010
5dc0: 00000000 c08d3000 00000000 c773c866 ffffffff9c 000003e8 c09641fd c00f5ec4
5de0: 00000241 c773c866 00000001 c08d3000 000081fd c00e1d48 00000241 000081fd
5e00: 00000002 00000300 00000001 c773c866 c09caa2a c065e8b8 c09caa2a 00008000
5e20: c065e8b8 00000007 c0595418 c063cf84 c065e8b8 00005644 00008000 c065e8b8
5e40: 00000007 c0595418 c4000000 c063c830 c09ca9bc 00005644 00008000 c063c880
5e60: 00008000 c4000000 c09c8000 c063c848 c063c508 c0659bb4 c4000000 c0659e5c
5e80: 00000000 00000000 c05dfef8 c065e8b8 c4000000 0011cdf4 00000000 00000000
5ea0: c0659f1c 00000000 00000000 c0659f48 00000000 00008000 c065e938 c063c508
5ec0: 00000000 c063ca9c 00000000 c065e938 c063c508 c0059100 c05de7a4 c773c866
5ee0: c0966c00 c069b058 c069b054 00000040 c066aad0 00000000 c096ca94 c0805005
5f00: c0804000 c063d5c4 c8825f2c c096ca90 c0813320 c773c866 c0805000 c096ca90
5f20: c0813320 00000040 c0805000 c0043084 c096ca90 c0813320 00000040 c0039294
5f40: c0804000 c0804044 c0813320 c0804000 c0813338 c0804018 c06830c0 c0823740
5f60: 00000088 c0039898 00000000 c0802ce0 c0823740 c003984c c0813320 c0829dc0
5f80: c881lea8 00000000 00000000 c00400b8 c0802ce0 c0040004 00000000 00000000
5fa0: 00000000 00000000 00000000 c0008528 00000000 00000000 00000000 00000000
5fc0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
5fe0: 00000000 00000000 00000000 00000000 00000013 00000000 00000000 00000000
vfs open from path_openat+0xa3c/0xd34
path_openat from do_filp_open+0x48/0xc0
do_filp_open from filp_open+0x12c/0x1f4
filp_open from do_name+0xa4/0x220
do_name from write_buffer+0x24/0x3c
write_buffer from flush_buffer+0x38/0x98
```

제출기한 및 평가

❖제출기한

- 4월 4일 수업시간 전까지 (1차)
 - ✓ 수업시간때 과제 수행방법 추가안내
- 4월 11일 수업시간 전까지 (2차)
 - ✓ 수업시간때 과제 수행방법 추가안내 (그대로 따라하면 되는 수준)
- 4월 18일 수업시간 전까지 (3차)

❖점수: 5점만점 (5% 성적반영)

- 과제1 5%, 과제2 5%. 총합 10% 성적비율

❖평가방식

- 최초 제출자 1인-> 가산점 1점
 - ✓ 중간고사+기말고사 점수가 만점이 아닌경우 +1점 (만점이면 가산점 없음)
- 1차 평가
 - ✓ 1차 기한이내 제출완료 (코드검증 완료) 시 5점
- 2차 평가
 - ✓ 2차 기한이내 제출완료 (코드검증 완료) 시 4점
- 3차 평가
 - ✓ 3차 기한이내 제출완료 (코드검증 완료) 시 3점
 - ✓ 3차 기한이내 제출완료 (코드검증 실패) 시 1점

