

Re: [PATCH] x86: extend e820 ealy_res support 32bit – fix v2

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-05/msg11570.html>

- *From:* "Yinghai Lu" <yhlu.kernel@xxxxxxxx>
 - *Date:* Sat, 24 May 2008 12:57:11 –0700
-

On Sat, May 24, 2008 at 1:54 AM, Jeremy Fitzhardinge <jeremy@xxxxxxxx> wrote:

Yinghai Lu wrote:

[PATCH] x86: extend e820 ealy_res support 32bit – fix v2

use find_e820_area to find address for new RAMDISK, instead of using ram blindly

also print out low ram and bootmap info

v2: remove extra –1 in reaseve_early calling
panic if can not find space for new RAMDISK

OK, I've fixed earlyprintk=xen, so I can finally get some useful debugging information.

With this patch it still crashes, but outputs:

```
(early) Reserving virtual address space above 0xf57fe000
(early) Linux version 2.6.26-rc3-sched-devel.git (jeremy@xxxxxxxxxxxxxxxx)
(gcc version 4.1.2 20070925 (Red Hat 4.1.2-33)) #466 SMP PREEMPT Sat May 24
01:05:41 PDT 2008
(early) ACPI in unprivileged domain disabled
(early) BIOS-provided physical RAM map:
(early) Xen: 0000000000000000 – 000000000009f000 (usable)
(early) Xen: 0000000000100000 – 0000000010000000 (usable)
(early) console [xenboot0] enabled
(early) debug: ignoring loglevel setting.
(early) limit_regions start: 0000000000000000 – 000000000009f000 (early)
(usable)
(early) limit_regions start: 0000000000100000 – 0000000010000000 (early)
(usable)
(early) limit_regions endfunc: 0000000000000000 – 000000000009f000 (early)
(usable)
```

Re: [PATCH] x86: extend e820 ealy_res support 32bit – fix v2

```
(early) limit_regions endfunc: 000000000100000 – 0000000010000000 (early)
(usable)
(early) user-defined physical RAM map:
(early) user: 0000000000000000 – 000000000009f000 (early) (usable)
(early) user: 0000000000100000 – 0000000010000000 (early) (usable)
(early) 0MB HIGHMEM available.
(early) 256MB LOWMEM available.
(early) low ram: 0102c000 – 10000000
(early) bootmap 0102c000 – 0102e000
(early) early res: 0 [0–fff] BIOS data page
(early) early res: 1 [1000–1fff] EX TRAMPOLINE
(early) early res: 2 [6000–6fff] TRAMPOLINE
(early) early res: 3 [102c000–102dfff] BOOTMAP
(early) Scan SMP from c0000000 for 1024 bytes.
(early) Scan SMP from c009fc00 for 1024 bytes.
(early) Scan SMP from c00f0000 for 65536 bytes.
(early) Scan SMP from c00c2c20 for 1024 bytes.
(early) NX (Execute Disable) protection: active
[crash]
```

Hm, I think this is the problem:

```
/*
 * don't need to reserve again, already reserved early
 * in i386_start_kernel
 */
```

A Xen pv boot doesn't presently go via `i386_start_kernel`; it goes directly from `xen_start_kernel` to `start_kernel` (you can see that the "early res" lines are missing important things like the kernel code and pagetables).

I tried making `xen_start_kernel` directly call `i386_start_kernel`, and it nearly works. The problem is that the initial address space layout for a Xen domain is:

```
kernel
ramdisk
init pagetable
```

which means that in

```
reserve_early(__pa_symbol(&_text), __pa_symbol(&_end), "TEXT DATA
BSS");
```

```
#ifdef CONFIG_BLK_DEV_INITRD
/* Reserve INITRD */
if (boot_params.hdr.type_of_loader && boot_params.hdr.ramdisk_image)
{
    u64 ramdisk_image = boot_params.hdr.ramdisk_image;
    u64 ramdisk_size = boot_params.hdr.ramdisk_size;
```

Re: [PATCH] x86: extend e820 ealy_res support 32bit – fix v2

```
u64 ramdisk_end = ramdisk_image + ramdisk_size;
reserve_early(ramdisk_image, ramdisk_end, "RAMDISK");
}
#endif
reserve_early(__pa_symbol(&_end), init_pg_tables_end,
"INIT_PG_TABLE");
```

the INIT_PG_TABLE reserve_early() will panic because it partially overlaps with the RAMDISK reservation.

great. i guess 64bit XEN pv will call x86_64_start_kernel.

INIT_PG_TABLE is right after "TEXT DATA BSS".

So your bootloader will not leave space between kernel and ramdisk?

or need to put INIT_PG_TABLE before end of BSS like 64bit did...

YH

—

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>