



author Conor Dooley <conor.dooley@microchip.com> 2022-11-07 15:15:25 +0000
committer Greg Kroah-Hartman <gregkh@linuxfoundation.org> 2022-11-16 10:04:04 +0100
commit [93598deb101540c4f9e7de15099ea8255b965fc2](#) (patch)
tree [ed792f861669ad21db607f9190f8c46350b32f1c](#)
parent [3822d92737cddbba0af5e7c45294c95b15e668e61](#) (diff)
download [linux-93598deb101540c4f9e7de15099ea8255b965fc2.tar.gz](#)

diff options

context:

3 ▼

space:

include ▼

mode:

unified ▼

riscv: fix reserved memory setup

[Upstream commit 50e63dd8ed92045eb70a72d7ec725488320fb68b]

Currently, RISC-V sets up reserved memory using the "early" copy of the device tree. As a result, when trying to get a reserved memory region using `of_reserved_mem_lookup()`, the pointer to reserved memory regions is using the early, pre-virtual-memory address which causes a kernel panic when trying to use the buffer's name:

```
Unable to handle kernel paging request at virtual address 00000000401c31ac
Oops [#1]
```

```
Modules linked in:
```

```
CPU: 0 PID: 0 Comm: swapper Not tainted 6.0.0-rc1-00001-g0d9d6953d834 #1
```

```
Hardware name: Microchip PolarFire-SoC Icicle Kit (DT)
```

```
epc : string+0x4a/0xea
```

```
ra : vsnprintf+0x1e4/0x336
```

```
epc : ffffffff80335ea0 ra : ffffffff80338936 sp : ffffffff81203be0
```

```
gp : ffffffff812e0a98 tp : ffffffff8120de40 t0 : 0000000000000000
```

```
t1 : ffffffff81203e28 t2 : 7265736572203a46 s0 : ffffffff81203c20
```

```
s1 : ffffffff81203e28 a0 : ffffffff81203d22 a1 : 0000000000000000
```

```
a2 : ffffffff81203d08 a3 : 0000000081203d21 a4 : ffffffff81203d21
```

```
a5 : 00000000401c31ac a6 : ffff0a00ffffff04 a7 : ffffffff81203d21
```

```
s2 : ffffffff81203d08 s3 : ffffffff81203d00 s4 : 0000000000000000
```

```
s5 : ffffffff000000ff s6 : 000000000000ffff s7 : 00000000fffff000
```

```
s8 : ffffffff80d9821a s9 : ffffffff81203d22 s10: 0000000000000002
```

```
s11: ffffffff80d9821c t3 : ffffffff812f3617 t4 : ffffffff812f3617
```

```
t5 : ffffffff812f3618 t6 : ffffffff81203d08
```

```
status: 0000000200000100 badaddr: 00000000401c31ac cause: 0000000000000000
```

```
[<fffffff80338936>] vsnprintf+0x1e4/0x336
```

```
[<fffffff80055ae2>] vprintk_store+0xf6/0x344
```

```
[<fffffff80055d86>] vprintk_emit+0x56/0x192
```

```
[<fffffff80055ed8>] vprintk_default+0x16/0x1e
```

```
[<fffffff800563d2>] vprintk+0x72/0x80
```

```
[<fffffff806813b2>] _printk+0x36/0x50
```

```
[<fffffff8068af48>] print_reserved_mem+0x1c/0x24
```

```
[<fffffff808057ec>] paging_init+0x528/0x5bc
```

```
[<fffffff808031ae>] setup_arch+0xd0/0x592
```

```
[<fffffff8080070e>] start_kernel+0x82/0x73c
```

`early_init_fdt_scan_reserved_mem()` takes no arguments as it operates on `initial_boot_params`, which is populated by `early_init_dt_verify()`. On RISC-V, `early_init_dt_verify()` is called twice. Once, directly, in

setup_arch() if CONFIG_BUILTIN_DTB is not enabled and once indirectly, very early in the boot process, by parse_dtb() when it calls early_init_dt_scan_nodes().

This first call uses dtb_early_va to set initial_boot_params, which is not usable later in the boot process when early_init_fdt_scan_reserved_mem() is called. On arm64 for example, the corresponding call to early_init_dt_scan_nodes() uses fixmap addresses and doesn't suffer the same fate.

Move early_init_fdt_scan_reserved_mem() further along the boot sequence, after the direct call to early_init_dt_verify() in setup_arch() so that the names use the correct virtual memory addresses. The above supposed that CONFIG_BUILTIN_DTB was not set, but should work equally in the case where it is - unflattened_and_copy_device_tree() also updates initial_boot_params.

Reported-by: Valentina Fernandez <valentina.fernandezalanis@microchip.com>

Reported-by: Evgenii Shatokhin <e.shatokhin@yadro.com>

Link: <https://lore.kernel.org/linux-riscv/f8e67f82-103d-156c-deb0-d6d6e2756f5e@microchip.com/>

Fixes: 922b0375fc93 ("riscv: Fix memblock reservation for device tree blob")

Signed-off-by: Conor Dooley <conor.dooley@microchip.com>

Tested-by: Evgenii Shatokhin <e.shatokhin@yadro.com>

Link: <https://lore.kernel.org/r/20221107151524.3941467-1-conor.dooley@microchip.com>

Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>

Signed-off-by: Sasha Levin <sashal@kernel.org>

Diffstat

-rw-r--r--	arch/riscv/kernel/setup.c	1
-rw-r--r--	arch/riscv/mm/init.c	1

2 files changed, 1 insertions, 1 deletions

diff --git a/arch/riscv/kernel/setup.c b/arch/riscv/kernel/setup.c

index ad76bb59b0590d..67ec1fadcf248 100644

--- a/arch/riscv/kernel/setup.c

+++ b/arch/riscv/kernel/setup.c

@@ -283,6 +283,7 @@ void __init setup_arch(char **cmdline_p)
 else

pr_err("No DTB found in kernel mappings\n");

#endif

+ early_init_fdt_scan_reserved_mem();
 misc_mem_init();

init_resources();

diff --git a/arch/riscv/mm/init.c b/arch/riscv/mm/init.c

index b56a0a75533fe3..50a1b6edd49182 100644

--- a/arch/riscv/mm/init.c

+++ b/arch/riscv/mm/init.c

@@ -262,7 +262,6 @@ static void __init setup_bootmem(void)
 memblock_reserve(dtb_early_pa, fdt_totalsize(dtb_early_va));
 }

- early_init_fdt_scan_reserved_mem();
 dma_contiguous_reserve(dma32_phys_limit);
 if (IS_ENABLED(CONFIG_64BIT))
 hugetlb_cma_reserve(PUD_SHIFT - PAGE_SHIFT);