# Bug 254764 - GPT 'bootme' flag is not respected on AARCH64

| | |
|---|---|
| **Status:** Closed Not A Bug | **Reported:** 2021-04-04 14:13 UTC by Slawomir Wojciech Wojtczak |
| | **Modified:** 2021-04-04 16:08 UTC (History) |
| **Alias:** None | **CC List:** 1 user (show) |
| **Product:** Base System | **See Also:** |
| **Component:** arm (show other bugs) | |
| **Version:** 13.0-STABLE | |
| **Hardware:** arm64 Any | |
| **Importance:** --- Affects Only Me | |
| **Assignee:** freebsd-arm (Nobody) | |
| **URL:** | |
| **Keywords:** | |
| **Depends on:** | |
| **Blocks:** | |

---

| **Attachments** |
|---|
| Add an attachment (proposed patch, testcase, etc.) |

Note

You need to log in before you can comment on or make changes to this bug.

Slawomir Wojciech Wojtczak 2021-04-04 14:13:57 UTC Description

VMDISK fetched from here:
https://download.freebsd.org/ftp/releases/VM-IMAGES/13.0-RC4/aarch64/Latest/FreeBSD-13.0-RC4-arm64-aarch64.raw.xz

```
host % export VMDISK=FreeBSD-13.0-RC4-arm64-aarch64.raw
host % qemu-system-aarch64 \
    -m 4096M \
    -cpu cortex-a57 \
    -M virt \
    -bios /usr/local/share/u-boot/u-boot-qemu-arm64/u-boot.bin \
    -serial telnet::4444,server \
    -nographic \
    -drive if=none,file=${VMDISK},format=raw,id=hd0 \
    -device virtio-blk-device,drive=hd0 \
    -device virtio-net-device,netdev=net0 \
    -netdev user,id=net0
```

The GPT bootme flag is set on p4 partition:

```
root@freebsd:~ # gpart show
=>       3  29426709  vtbd0  GPT  (14G)
         3     66584      1  efi  (33M)
     66587   2097152      2  freebsd-swap  (1.0G)
   2163739   8388608      3  freebsd-ufs  (4.0G)
  10552347   8388608      4  freebsd-ufs  [bootme]  (4.0G)
  18940955  10485757         - free -  (5.0G)
```

Yet FreeBSD boots from p3 partition:

```
root@freebsd:~ # df -m
Filesystem      1M-blocks Used Avail Capacity  Mounted on
/dev/vtbd0p3         3958 2709   931    74%    /
devfs                   0    0     0   100%    /dev
/dev/vtbd0p4         3958 2706   935    74%    /ufsbe/4
/dev/gpt/efiesp        31    1    30     4%    /boot/efi
```

What am I doing wrong here?

Why it does not boot from p4 where 'bootme' flag is set?

Regards.

Slawomir Wojciech Wojtczak 2021-04-04 14:22:21 UTC Comment 1

The upper method was with U-BOOT.

The UEFI boot has the same problem - does not respect 'bootme' GPT flag.

```
host % export VMDISK=FreeBSD-13.0-RC4-arm64-aarch64.raw
host % qemu-system-aarch64 \
    -m 4096M \
    -cpu cortex-a57 \
    -M virt \
    -bios edk2-aarch64-code.fd \
    -serial telnet::4444,server \
    -nographic \
    -drive if=none,file=${VMDISK},format=raw,id=hd0 \
    -device virtio-blk-device,drive=hd0 \
    -device virtio-net-device,netdev=net0 \
    -netdev user,id=net0
```

Regards.

Warner Losh 2021-04-04 14:26:16 UTC Comment 2

The 'bootme' flag is a FreeBSD-specific extension. Nobody else uses it. And it was for BIOS booting only until recently.

You must use gptboot.efi in place of boot1.efi or loader.efi for the non-standard, FreeBSD-specific bootme flag to be honored. there's no installer option to create this, and the images we ship do not have this setup. IT's considered obsolete with UEFI because of the UEFI boot manager. However, there are far too many UEFI implementations where it's not possible any other way.

Your qemu invocation doesn't give you the right environment for efibootmgr to work properly.

So you'll need copy gptboot.efi to \EFI\BOOT\BOOTAA64.EFI to get this functionality.

The UEFI direct method might work with efibootmgr, but you need to use the EDK2 build with detached variables, not the normal one.

Slawomir Wojciech Wojtczak 2021-04-04 15:55:38 UTC Comment 3

(In reply to Warner Losh from comment #2)

Thank You for explanations.

Recently I wrote Boot Environments manager for UFS (not ZFS).

Here:
https://vermaden.wordpress.com/2021/04/02/ufs-boot-environments/
https://github.com/vermaden/ufsbe

It requires 'bootme' flags support to work.

This is why I asked.

I do not need FreeBSD images to support that but I would like to make possible to use on Raspberry Pi for example.

That was the reason for these tests.

I do not own any modern arm64/aarch64 Raspberry Pi at the moment - only Raspberry Pi 2 which is armv7/arm32 only so I wanted to test that on FreeBSD images and QEMU emulator.

Regards.

Slawomir Wojciech Wojtczak 2021-04-04 16:08:59 UTC Comment 4

This 'fix' solves the problem:

```
# cp /boot/gptboot.efi /boot/efi/EFI/BOOT/bootaa64.efi
```

Now mine UFS Boot Environments manager works like a charm.

Big thanks for fast help :)

Regards.