New issue

Jump to bottom

Lack hardware wake-up support checking #78

Closed **TheSilentDawn** opened this issue on Oct 14, 2020 · 5 comments

Assignees

Labels enhancement internal bug tracker mw usb

Projects stm32cube-mcu-fw-dashb...

Milestone v1.10.0

TheSilentDawn commented on Oct 14, 2020 • edited

Describe the set-up

- Software:
  - STM32Cube MCU & MPU Packages
- Version:
  - STM32Cube_FW_H7_V1.8.0
- Verification Hardware Platform:
  - STM32H7B3

Describe the bug

- Function:
  - static void USBH_ParseCfgDesc(USBH_CfgDescTypeDef *cfg_desc, uint8_t *buf, uint16_t length)
- Location:
  - STM32CubeH7/Middlewares/ST/STM32_USB_Host_Library/Core/Src/usbh_ctlreq.c
    Line 399 in 79196b0

```
399     cfg_desc->bmAttributes        = *(uint8_t *)(buf + 7);
```

- Type:
  - Denial-of-Service.
- Result:
  - The system will hang when trying to set a remote wake-up feature.
- Description:
  - The function USBH_ParseCfgDesc() parses the configuration descriptor, interface descriptor, and endpoint descriptor by input data from a USB device.
  - And it set the variable cfg_desc->bmAttributes by the input data from the USB device. This variable will be used as part of a judgment in the function USBH_Process() as shown in

    STM32CubeH7/Middlewares/ST/STM32_USB_Host_Library/Core/Src/usbh_core.c
    Line 643 in 79196b0

```
643     if ((phost->device.CfgDesc.bmAttributes) & (1U << 5))
```

    . With a malformed value, the remote wakeup may be enabled as shown in

    STM32CubeH7/Middlewares/ST/STM32_USB_Host_Library/Core/Src/usbh_core.c
    Line 643 in 79196b0

```
643     if ((phost->device.CfgDesc.bmAttributes) & (1U << 5))
```

    . If the hardware doesn't support this feature, the system will hang due to a FAIL return value by the function USBH_HandleControl().

How To Reproduce

1. Running MSC_Standalone application on the STM32H7B3I platform
2. Plug a USB disk
3. Use the attached Bug4.txt to replace the USB device packet. Bug4.txt

Additional context

- To patch it, the program should check if the hardware supports a remote wake-up feature.

ALABSTM added this to **To do** in **stm32cube-mcu-fw-dashboard** on Oct 15, 2020

RKOUSTM assigned **ALABSTM** on Nov 18, 2020

ALABSTM commented on Nov 24, 2020 Contributor

Hi **@TheSilentDawn**,

Thank you for this other report. If I understood the idea we should ensure that the software embedded within the *host* checks whether the value *bmAttributes* provided by the *device* is correct? Which reference should the *host* base itself on to perform such a check? Should we not assume the data provided by the *device* is correct?

Otherwise, when to stop checking? In case the *host* has a reference to base itself on, then would it not also make sense to check the validity of this reference too?

I hope my questions make sense. My goal is to be sure I understood correctly your point before reporting it to our development teams. I am looking forward to reading your reply. Thank you again.

With regards,

TheSilentDawn commented on Nov 25, 2020 • edited — Author

Hi **@ALABSTM**,
No, I think the software embedded within the host should check if the hardware or the board supports or enables the wake-up feature. If not, the firmware shouldn't check bmAttributes as the current logic which will lead the whole system to hang. And to patch it, the wake-up feature checking codes should be added before the bmAttributes is accessed in the function and position reported upper. If anything not clear, please let me know. Thanks for your help.^_^

ALABSTM commented on Dec 2, 2020 — Contributor

Hi **@TheSilentDawn**,

Please allow me this couple of questions:

- Is there any way for the host to check whether the device supports the wake-up feature other than the `bmAttributes` read from the device descriptor?
- You mentioned *"the wake-up feature checking codes"*. Are you referring to some piece of code already present in our source files that should be moved before the reading of the `bmAttributes` ?

Thank you,

ALABSTM moved this from **To do** to **Assigned** in **stm32cube-mcu-fw-dashboard** on Dec 2, 2020

ALABSTM added the **mw** label on Dec 2, 2020

ALABSTM added the **enhancement** label on Dec 15, 2020

ALABSTM added the **usb** label on Jan 18, 2021

ALABSTM moved this from **Assigned** to **In progress** in **stm32cube-mcu-fw-dashboard** on Jan 18, 2021

ALABSTM added the **internal bug tracker** label on Jan 18, 2021

ALABSTM commented on Jan 18, 2021 — Contributor

ST Internal Reference: 99173

ALABSTM added this to the **v1.10.0** milestone on Feb 22, 2021

ALABSTM moved this from **In progress** to **To release** in **stm32cube-mcu-fw-dashboard** on Feb 22, 2021

TheSilentDawn mentioned this issue on May 31, 2021

No validity chekcing on the variable dev_desc->bMaxPacketSize #75

Closed

ALABSTM commented on Mar 14 — Contributor

Hi **@TheSilentDawn**,

Hope you're fine. Just to inform you the fix has been published in the frame of **v1.10.0** release.

With regards,

ALABSTM closed this as completed on Mar 14

stm32cube-mcu-fw-dashboard automation moved this from **To release** to **Done** on Mar 14

Assignees

ALABSTM

Labels

enhancement **internal bug tracker** **mw** **usb**

Projects

stm32cube-mcu-fw-dashboard

Done

Milestone

v1.10.0

Development

No branches or pull requests

2 participants