```
[PATCH] pcmcia: add error handling for pcmcia_enable device
Thouyang Jia <jiazhouyang09@gmail.com>

Mon, Jun 11, 2018 at 1:1

Cc: Zhouyang Jia sjiazhouyang09@gmail.com>, "James E.J. Bottomley" <jejb@linux.vnet.ibm.com>, "Martin K. Petersen" <martin.petersen@oracle.com>, Johannes Thumshirn <jthumshirn@suse.de>, Hannes Reinecke <hare@suse.com>, linux-scsi@vger.kemel.org, linux-kemel@vger.kemel.org

Thouyang Jia sjiazhouyang09@gmail.com>, "James E.J. Bottomley" <jejb@linux.vnet.ibm.com>, "Martin K. Petersen" <martin.petersen@oracle.com>, Johannes Thumshirn <jthumshirn@suse.de>, Hannes Reinecke <hare@suse.com>, linux-scsi@vger.kemel.org, linux-kemel@vger.kemel.org

Thouyang Jia sjiazhouyang09@gmail.com>
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             Mon, Jun 11, 2018 at 1:15 AM
       When pcmcia_enable_device fails, the lack of error-handling code may cause unexpected results
       This patch adds error-handling code after calling pcmcia_enable_device
       Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.co
         drivers/scsi/pcmcia/qlogic_stub.c | 8 +++++++
1 file changed, 7 insertions(+), 1 deletion(-)
       diff –git a/drivers/scsi/pcmcia/qlogic_stub.c b/drivers/scsi/pcmcia/qlogic_stub.c index 0556054.9287d52 100644
       scsi_info_t *info = link->priv;
                      ret = pcmcia_enable_device(link);
if (ret) {
                                    pcmcia_disable_device(link);
return -ENODEV;
                     pcmcia_enable_device(link);
if ((info->manf_id == MANFID_MACNICA) ||
(info->manf_id == MANFID_PIONEER) ||
(info->manf_id == 0x0098)) {
Johannes Thumshirn | Johannes T
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             Mon. Jun 11 2018 at 3:21 AM
     On Mon, Jun 11, 2018 at 01:15:50PM +0800, Zhouyang Jia wrote: 
> When pcmcia_enable_device fails, the lack of error-handling code may 
> cause unexpected results.
         > This patch adds error-handling code after calling pcmcia_enable_device
          > Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com
       > drivers/scsi/pcmcia/qlogic_stub.c | 8 +++++++
> 1 file changed, 7 insertions(+), 1 deletion(-)
        > diff –git a'drivers/scsi/pcmcia/qlogic_stub.c b'drivers/scsi/pcmcia/qlogic_stub.c 
> index 0556054.9287d52 100644 
> --- a'drivers/scsi/pcmcia/qlogic_stub.c 
> +++ b'drivers/scsi/pcmcia/qlogic_stub.c 
> @2-254.8 +254,14 @ @3-stalic void qlogic_release(struct pcmcia_device *link) 
> staltic int qlogic_resume(struct pcmcia_device *link)
       > static int qlogic_resume(struct p
> {
> scsi_info_t *info = link->priv;
> + int ret;
     pcmcia_enable_device() can fail for three reasons:
1) the socket is not present
2) the configuration is locked
3) setting the socket's power control fails
       In all three cases I think it's actually an error to call pcmcia_disable_device().
     Imagine the following scenario:
permia_enable_device() falled because the device configuration is
locked by another driver, then you call porncia_disable_device() which
calls permia_release_configuration(), permia_release_configuration()
checks if the device is locked, decrements the lock counter, clears
the CONFIG_LOCKED bit in the PCMCIA config and sets the voltage to 0
without the first driver every noticing it.
     Byte,
Johannes
         --
Johannes Thumshi
    Johannes Thumshirn Storage

jthumshirn@suse.de +49 911 74053 689

SUSE LINUX GmbH, Maxfeldstr. 5, 90409 Nurbrebrg

GF: Felix Imendorffer, Jane Smithard, Graham Norton

HRB 21284 (AG Nümberg)

Key fingerprint = EC38 9CAB C2C4 F25D 8600 DDD0 0393 969D 2D76 0850
Thouyang Jia <jiazhouyang09@gmail.com>
To: Johannes Thumshirn <jthumshirn@suse.de>
Co: "almes E.J. Bottomley" <jejb@linux.vnet.ibm.com>, "Martin K. Petersen" <martin.petersen@oracle.com>, Hannes Reinecke <hare@suse.com>, linux-scsi@vger.kernel.org, linux-kernel@vger.kernel.org
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        Wed, Jun 13, 2018 at 11:37 PM
       I reported this bug since more than 90% callsites of pcmcia_enable_device are well handled in kernel, and many of them use pcmcia_disable_device.
     I'll remove pcmcia_disable_device in v2 if necessary
     Thanks for your kind reply.
     Best,
Zhouyang
[Quoted text h
 Zhouyang Jia <jiazhouyang09@gmail.com>
Thu, Jun 14, 2018 at 5:23 AM
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, "James E.J. Bottomley" <jejb@linux.vnet.ibm.com>, "Martin K. Petersen" <martin.petersen@oracle.com>, Johannes Thumshirm@suse.de>, Hannes Reinecke <hare@suse.com>, linux-scsi@vger.kernel.org, linux-kernel@vger.kernel.org
       When pcmcia_enable_device fails, the lack of error-handling code may
     This patch adds error-handling code after calling pcmcia enable device
       Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.co
       v1->v2:
- Remove pcmcia_disable_device.
       diff --git a/drivers/scsi/pcmcia/qlogic_stub.c b/drivers/scsi/pcmcia/qlogic_stub.c index 0556054..eb2c8bf 100644
       ### BURN U000U9. ### BURN U000U U000
                     scsi_info_t *info = link->priv;
int ret;
     .
+ ret = pcmcia_enable_device(link);
+ if (ret)
+ return -ENODEV;
[Quoted text hidden]
Andy Shevchenko <andy.shevchenko@gmail.com>
Thu, Jun 14, 2018
To: Zhouyang Jia <a href="mailto:size-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-went-linux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibilinux-escis/eligibili
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             Thu, Jun 14, 2018 at 5:39 AM
```

c: "James E.J. Bottomley^{*} <|e|b@llinux.vnet.ibm.com>, "Martin K. Petersen" <martin.petersen@oracle.com>, Johannes Thumshim <||thumshim@suse.de>, Hannes Reinecke <hare@suse.com>, ||inux-scsi@vger.kernel.org>, Linux Kernel Mailing List <||inux-kernel@vger.kernel.org>

On Thu, Jun 14, 2018 at 12:23 PM, Zhouyang Jia <||iax|ouyang09@gmail.com> wrote:
> When pomoia_enable_device fails, the lack of error-handling code may

> cause unexpected results.
> cause unexpected results.

> cause unexpected results.

What results?
You need to elaborate this.,

```
Thu, Jun 14, 2018 at 7.41 PM
Cv. Zhouyang Jia -sjachouyangüßggmail com>. "James E.J. Bottomley" -sjelb@linux.vnet.lbm.com>. "Martin K. Petersen" -smartin.petersen@oracle.com>, Hannes Reinecke -share@suse.com>, Johannes Thumshim -sjm.mshim@suse.de>, linux-sca@yeer.kemel.org. linux-ternel@yeer.kemel.org.
When pornia_e.abable_device fails, the lack of error-handling code may cause unexpected results.

This patch adds error-handling code after calling pornia_enable_device.

Signed-off-by: Zhouyang Jia -sjachouyangüßggmail.com>

VI -v2:
- Remove pornia_disable_device.

V2-v3:
- Change the return value.

- drifversiscas/pornia/displog_c, stub.c | 6 +++++-|
- file changed_S. delf-sise.

- disploy-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-general-gen
```

> Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>

On Fri, Jun 15, 2018 at 07:41:38AM +0800, Zhouyang Jia wrote: > When pcmcia_enable_device fails, the lack of error-handling code may > cause unexpected results. > > This patch adds error-handling code after calling pcmcia_enable_device.

> This patch adds error-handling code after calling pcmcia_enable_device

> + int ret; > + > + ret = pcmcia_enable_device(link); > + if (ret) > + return -ENODEV;

And why not to return ret?

-With Best Regards,
Andy Shevchenko

Applied to pcmcia-next.

Thanks, Dominik