[PATCH] staging: rtl8192u: add error handling for usb_alloc_urb

13 messages

Zhouyang Jia <jiazhouyang09@gmail.com> Mon, Jun 11, 2018 at 4:31 AM
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Colin Ian King <colin.king@canonical.com>, Shreeya Patel <shreeya.patel23498@gmail.com>, Kees Cook <keescook@chromium.org>, Jia-Ju Bai <baijiaju1990@gmail.com>, devel@driverdev.osuosl.org, linux-kernel@vger.kernel.org

When usb_alloc_urb fails, the lack of error-handling code may
cause unexpected results.

This patch adds error-handling code after calling usb_alloc_urb.

Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
---
drivers/staging/rtl8192u/r8192U_core.c | 3 +++
1 file changed, 3 insertions(+)

```
diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
index 7a0dbc0..3f09615 100644
--- a/drivers/staging/rtl8192u/r8192U_core.c
+++ b/drivers/staging/rtl8192u/r8192U_core.c
@@ -1666,6 +1666,9 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
                void *oldaddr, *newaddr;

                priv->rx_urb[16] = usb_alloc_urb(0, GFP_KERNEL);
+               if (!priv->rx_urb[16])
+                       return -ENOMEM;
+
                priv->oldaddr = kmalloc(16, GFP_KERNEL);
                if (!priv->oldaddr)
                        return -ENOMEM;
--
2.7.4
```

Greg Kroah-Hartman <gregkh@linuxfoundation.org> Mon, Jun 11, 2018 at 3:21 PM
To: Zhouyang Jia <jiazhouyang09@gmail.com>

[Quoted text hidden]

Hi,

This is the friendly semi-automated patch-bot of Greg Kroah-Hartman.
You have sent him a patch that has triggered this response.

Right now, the development tree you have sent a patch for is "closed"
due to the timing of the merge window. Don't worry, the patch(es) you
have sent are not lost, and will be looked at after the merge window is
over (after the -rc1 kernel is released by Linus).

So thank you for your patience and your patches will be reviewed at this
later time, you do not have to do anything further, this is just a short
note to let you know the patch status and so you don't worry they didn't
make it through.

thanks,

greg k-h's patch email bot

Greg Kroah-Hartman <gregkh@linuxfoundation.org> Fri, Jun 15, 2018 at 7:15 AM
To: Zhouyang Jia <jiazhouyang09@gmail.com>
Cc: devel@driverdev.osuosl.org, Kees Cook <keescook@chromium.org>, linux-kernel@vger.kernel.org, Jia-Ju Bai <baijiaju1990@gmail.com>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>

On Mon, Jun 11, 2018 at 04:31:11PM +0800, Zhouyang Jia wrote:
> When usb_alloc_urb fails, the lack of error-handling code may
> cause unexpected results.
>
> This patch adds error-handling code after calling usb_alloc_urb.
>
> Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
> ---
> drivers/staging/rtl8192u/r8192U_core.c | 3 +++
> 1 file changed, 3 insertions(+)
>
> diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
> index 7a0dbc0..3f09615 100644
> --- a/drivers/staging/rtl8192u/r8192U_core.c
> +++ b/drivers/staging/rtl8192u/r8192U_core.c
> @@ -1666,6 +1666,9 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
>   void *oldaddr, *newaddr;
>
>   priv->rx_urb[16] = usb_alloc_urb(0, GFP_KERNEL);
> +   if (!priv->rx_urb[16])
> +   return -ENOMEM;
> +

You just leaked memory :(

Well, this whole function leaks memory on the error paths, like here:

>   priv->oldaddr = kmalloc(16, GFP_KERNEL);
>   if (!priv->oldaddr)
>   return -ENOMEM;

So can you fix this all up at the same time?

thanks,

greg k-h

Zhouyang Jia <jiazhouyang09@gmail.com> Fri, Jun 15, 2018 at 12:25 PM
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Kees Cook <keescook@chromium.org>, Jia-Ju Bai <baijiaju1990@gmail.com>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>, devel@driverdev.osuosl.org, linux-kernel@vger.kernel.org

When usb_alloc_urb fails, the lack of error-handling code may
cause unexpected results.

This patch adds error-handling code after calling usb_alloc_urb.

Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
---
v1->v2:
- Fix memory leak.
---
drivers/staging/rtl8192u/r8192U_core.c | 18 ++++++++++++++---
1 file changed, 15 insertions(+), 3 deletions(-)

```
diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
index 7a0dbc0..6afab4e 100644
--- a/drivers/staging/rtl8192u/r8192U_core.c
+++ b/drivers/staging/rtl8192u/r8192U_core.c
@@ -1648,13 +1648,17 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
 #ifndef JACKSON_NEW_RX
        for (i = 0; i < (MAX_RX_URB + 1); i++) {
                priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
-               if (!priv->rx_urb[i])
+               if (!priv->rx_urb[i]) {
+                       kfree(priv->rx_urb);
                        return -ENOMEM;
+               }

                priv->rx_urb[i]->transfer_buffer =
                        kmalloc(RX_URB_SIZE, GFP_KERNEL);
-               if (!priv->rx_urb[i]->transfer_buffer)
+               if (!priv->rx_urb[i]->transfer_buffer) {
+                       kfree(priv->rx_urb);
                        return -ENOMEM;
+               }

                priv->rx_urb[i]->transfer_buffer_length = RX_URB_SIZE;
        }
@@ -1666,9 +1670,17 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
                void *oldaddr, *newaddr;

                priv->rx_urb[16] = usb_alloc_urb(0, GFP_KERNEL);
+               if (!priv->rx_urb[16]) {
+                       kfree(priv->rx_urb);
+                       return -ENOMEM;
+               }
+
                priv->oldaddr = kmalloc(16, GFP_KERNEL);
-               if (!priv->oldaddr)
+               if (!priv->oldaddr) {
+                       kfree(priv->rx_urb);
                        return -ENOMEM;
+               }
+
                oldaddr = priv->oldaddr;
```

```
                align = ((long)oldaddr) & 3;
                if (align) {
--
2.7.4
```

Greg Kroah-Hartman <gregkh@linuxfoundation.org> Fri, Jun 15, 2018 at 12:33 PM
To: Zhouyang Jia <jiazhouyang09@gmail.com>
Cc: devel@driverdev.osuosl.org, Kees Cook <keescook@chromium.org>, linux-kernel@vger.kernel.org, Jia-Ju Bai <baijiaju1990@gmail.com>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>

> On Sat, Jun 16, 2018 at 12:25:23AM +0800, Zhouyang Jia wrote:
> When usb_alloc_urb fails, the lack of error-handling code may
> cause unexpected results.
>
> This patch adds error-handling code after calling usb_alloc_urb.
>
> Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
> ---
> v1->v2:
> - Fix memory leak.
> ---
> drivers/staging/rtl8192u/r8192U_core.c | 18 +++++++++++++++---
> 1 file changed, 15 insertions(+), 3 deletions(-)
>
> diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
> index 7a0dbc0..6afab4e 100644
> --- a/drivers/staging/rtl8192u/r8192U_core.c
> +++ b/drivers/staging/rtl8192u/r8192U_core.c
> @@ -1648,13 +1648,17 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
> #ifndef JACKSON_NEW_RX
>   for (i = 0; i < (MAX_RX_URB + 1); i++) {
>   priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
> -   if (!priv->rx_urb[i])
> +   if (!priv->rx_urb[i]) {
> +   kfree(priv->rx_urb);
>   return -ENOMEM;
> +   }

{sigh}

No, you are still leaking memory on all of these changes that you just
made :(

greg k-h

Kees Cook <keescook@chromium.org> Fri, Jun 15, 2018 at 12:47 PM
To: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, devel@driverdev.osuosl.org, LKML <linux-kernel@vger.kernel.org>, Jia-Ju Bai <baijiaju1990@gmail.com>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>

On Fri, Jun 15, 2018 at 9:33 AM, Greg Kroah-Hartman
<gregkh@linuxfoundation.org> wrote:
> On Sat, Jun 16, 2018 at 12:25:23AM +0800, Zhouyang Jia wrote:
>> When usb_alloc_urb fails, the lack of error-handling code may
>> cause unexpected results.
>>
>> This patch adds error-handling code after calling usb_alloc_urb.
>>
>> Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
>> ---
>> v1->v2:
>> - Fix memory leak.
>> ---
>> drivers/staging/rtl8192u/r8192U_core.c | 18 +++++++++++++++---
>> 1 file changed, 15 insertions(+), 3 deletions(-)
>>
>> diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
>> index 7a0dbc0..6afab4e 100644
>> --- a/drivers/staging/rtl8192u/r8192U_core.c
>> +++ b/drivers/staging/rtl8192u/r8192U_core.c
>> @@ -1648,13 +1648,17 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
>> #ifndef JACKSON_NEW_RX
>>   for (i = 0; i < (MAX_RX_URB + 1); i++) {
>>   priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
>> -   if (!priv->rx_urb[i])
>> +   if (!priv->rx_urb[i]) {
>> +   kfree(priv->rx_urb);

You're freeing rx_urb, which holds all the pointers to allocated
memory. You'll need to free each item of the array before freeing the
array itself:

for (i = 0; i < (MAX_RX_URB + 1); i++)
    kfree(priv->rx_urb[i]);
kfree(priv->rx_urb);

I think you need some kind of helper to do this, and you can call into
it from your error paths...

-Kees
[Quoted text hidden]
--
Kees Cook
Pixel Security

Kees Cook <keescook@chromium.org> Fri, Jun 15, 2018 at 12:47 PM
To: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, devel@driverdev.osuosl.org, LKML <linux-kernel@vger.kernel.org>, Jia-Ju Bai <baijiaju1990@gmail.com>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>

[Quoted text hidden]
(Though if you do this, rx_urb must be zero-initialized, so change the
kmalloc_array() to kcalloc()...)

-Kees
[Quoted text hidden]

Zhouyang Jia <jiazhouyang09@gmail.com> Fri, Jun 15, 2018 at 1:28 PM
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Colin Ian King <colin.king@canonical.com>, Jia-Ju Bai <baijiaju1990@gmail.com>, Shreeya Patel <shreeya.patel23498@gmail.com>, Kees Cook <keescook@chromium.org>, devel@driverdev.osuosl.org, linux-kernel@vger.kernel.org

When usb_alloc_urb fails, the lack of error-handling code may
cause unexpected results.

This patch adds error-handling code after calling usb_alloc_urb.

Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
---
v1->v2:
- Fix memory leak.
v2->v3:
- Release memory in error path.
---
 drivers/staging/rtl8192u/r8192U_core.c | 24 +++++++++++++++---------
 1 file changed, 15 insertions(+), 9 deletions(-)

```
diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
index 7a0dbc0..1c980e9 100644
--- a/drivers/staging/rtl8192u/r8192U_core.c
+++ b/drivers/staging/rtl8192u/r8192U_core.c
@@ -1649,12 +1649,12 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
        for (i = 0; i < (MAX_RX_URB + 1); i++) {
                priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
                if (!priv->rx_urb[i])
-                       return -ENOMEM;
+                       goto out_release_mem;

                priv->rx_urb[i]->transfer_buffer =
                        kmalloc(RX_URB_SIZE, GFP_KERNEL);
                if (!priv->rx_urb[i]->transfer_buffer)
-                       return -ENOMEM;
+                       goto out_release_mem;

                priv->rx_urb[i]->transfer_buffer_length = RX_URB_SIZE;
        }
@@ -1666,9 +1666,13 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
        void *oldaddr, *newaddr;

        priv->rx_urb[16] = usb_alloc_urb(0, GFP_KERNEL);
+       if (!priv->rx_urb[16])
+               goto out_release_mem;
+
        priv->oldaddr = kmalloc(16, GFP_KERNEL);
        if (!priv->oldaddr)
-               return -ENOMEM;
+               goto out_release_mem;
+
        oldaddr = priv->oldaddr;
        align = ((long)oldaddr) & 3;
        if (align) {
@@ -1686,17 +1690,19 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
        priv->pp_rxskb = kcalloc(MAX_RX_URB, sizeof(struct sk_buff *),
```

```
                                GFP_KERNEL);
        if (!priv->pp_rxskb) {
-               kfree(priv->rx_urb);
-
-               priv->pp_rxskb = NULL;
-               priv->rx_urb = NULL;
-
                DMESGE("Endpoint Alloc Failure");
-               return -ENOMEM;
+               goto out_release_mem;
        }

        netdev_dbg(dev, "End of initendpoints\n");
        return 0;
+
+out_release_mem:
+       for (i = 0; i < (MAX_RX_URB + 1); i++)
+               kfree(priv->rx_urb[i]);
+       kfree(priv->rx_urb);
+       priv->rx_urb = NULL;
+       return -ENOMEM;
 }

 #ifdef THOMAS_BEACON
--
2.7.4
```

---

kbuild test robot <lkp@intel.com> Fri, Jun 15, 2018 at 2:35 PM
To: linux-kernel-owner@vger.kernel.org
Cc: kbuild-all@01.org, Zhouyang Jia <jiazhouyang09@gmail.com>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Colin Ian King <colin.king@canonical.com>, Jia-Ju Bai <baijiaju1990@gmail.com>, Shreeya Patel <shreeya.patel23498@gmail.com>, Kees Cook <keescook@chromium.org>, devel@driverdev.osuosl.org, linux-kernel@vger.kernel.org

Hi,

Thank you for the patch! Yet something to improve:

[auto build test ERROR on staging/staging-testing]
[also build test ERROR on v4.17 next-20180615]
[if your patch is applied to the wrong git tree, please drop us a note to help improve the system]

url: https://github.com/0day-ci/linux/commits/linux-kernel-owner-vger-kernel-org/staging-rtl8192u-add-error-handling-for-usb_alloc_urb/20180616-012944
config: xtensa-allyesconfig (attached as .config)
compiler: xtensa-linux-gcc (GCC) 8.1.0
reproduce:
wget https://raw.githubusercontent.com/intel/lkp-tests/master/sbin/make.cross -O ~/bin/make.cross
chmod +x ~/bin/make.cross
# save the attached .config to linux build tree
GCC_VERSION=8.1.0 make.cross ARCH=xtensa

All errors (new ones prefixed by >>):

drivers/staging//rtl8192u/r8192U_core.c: In function 'rtl8192_usb_initendpoints':
>> drivers/staging//rtl8192u/r8192U_core.c:1701:7: error: 'i' undeclared (first use in this function)
for (i = 0; i < (MAX_RX_URB + 1); i++)
^
drivers/staging//rtl8192u/r8192U_core.c:1701:7: note: each undeclared identifier is reported only once for each function it appears in

vim +/i +1701 drivers/staging//rtl8192u/r8192U_core.c

```
 1688
 1689          memset(priv->rx_urb, 0, sizeof(struct urb *) * MAX_RX_URB);
 1690          priv->pp_rxskb = kcalloc(MAX_RX_URB, sizeof(struct sk_buff *),
 1691                                   GFP_KERNEL);
 1692          if (!priv->pp_rxskb) {
 1693                  DMESGE("Endpoint Alloc Failure");
 1694                  goto out_release_mem;
 1695          }
 1696
 1697          netdev_dbg(dev, "End of initendpoints\n");
 1698          return 0;
 1699
 1700  out_release_mem:
> 1701          for (i = 0; i < (MAX_RX_URB + 1); i++)
 1702                  kfree(priv->rx_urb[i]);
 1703          kfree(priv->rx_urb);
 1704          priv->rx_urb = NULL;
 1705          return -ENOMEM;
 1706  }
 1707
```

0-DAY kernel test infrastructure Open Source Technology Center
https://lists.01.org/pipermail/kbuild-all Intel Corporation

.config.gz
53K

Kees Cook <keescook@chromium.org> Fri, Jun 15, 2018 at 7:17 PM
To: Zhouyang Jia <jiazhouyang09@gmail.com>
Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Colin Ian King <colin.king@canonical.com>, Jia-Ju Bai <baijiaju1990@gmail.com>, Shreeya Patel <shreeya.patel23498@gmail.com>, devel@driverdev.osuosl.org, LKML <linux-kernel@vger.kernel.org>

On Fri, Jun 15, 2018 at 10:28 AM, Zhouyang Jia <jiazhouyang09@gmail.com> wrote:
> When usb_alloc_urb fails, the lack of error-handling code may
> cause unexpected results.
>
> This patch adds error-handling code after calling usb_alloc_urb.
>
> Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
> ---
> v1->v2:
> - Fix memory leak.
> v2->v3:
> - Release memory in error path.
> ---
> drivers/staging/rtl8192u/r8192U_core.c | 24 ++++++++++++++---------
> 1 file changed, 15 insertions(+), 9 deletions(-)
>
> diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
> index 7a0dbc0..1c980e9 100644
> --- a/drivers/staging/rtl8192u/r8192U_core.c
> +++ b/drivers/staging/rtl8192u/r8192U_core.c
> @@ -1649,12 +1649,12 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
>   for (i = 0; i < (MAX_RX_URB + 1); i++) {
>   priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
>   if (!priv->rx_urb[i])
> -   return -ENOMEM;
> +   goto out_release_mem;

You need to use kcalloc() above here for priv->rx_urb, otherwise you
may free random garbage. :)

-Kees
[Quoted text hidden]
[Quoted text hidden]

Zhouyang Jia <jiazhouyang09@gmail.com> Fri, Jun 15, 2018 at 10:01 PM
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Jia-Ju Bai <baijiaju1990@gmail.com>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>, Kees Cook <keescook@chromium.org>, devel@driverdev.osuosl.org, linux-kernel@vger.kernel.org

When usb_alloc_urb fails, the lack of error-handling code may
cause unexpected results.

This patch adds error-handling code after calling usb_alloc_urb.

Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
---
v1->v2:
- Fix memory leak.
v2->v3:
- Release memory in error path.
v3->v4:
- Use kcalloc instead of kmalloc_array.
---
 drivers/staging/rtl8192u/r8192U_core.c | 27 +++++++++++++++++----------
 1 file changed, 17 insertions(+), 10 deletions(-)

diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
index 7a0dbc0..d15ee4f 100644
--- a/drivers/staging/rtl8192u/r8192U_core.c
+++ b/drivers/staging/rtl8192u/r8192U_core.c
@@ -1639,8 +1639,9 @@ short rtl8192_tx(struct net_device *dev, struct sk_buff *skb)
 static short rtl8192_usb_initendpoints(struct net_device *dev)
 {
        struct r8192_priv *priv = ieee80211_priv(dev);
+   int i;

-       priv->rx_urb = kmalloc(sizeof(struct urb *) * (MAX_RX_URB + 1),
+   priv->rx_urb = kcalloc(MAX_RX_URB + 1, sizeof(struct urb *),
                               GFP_KERNEL);
        if (!priv->rx_urb)
```
```

```
                return -ENOMEM;
@@ -1649,12 +1650,12 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
        for (i = 0; i < (MAX_RX_URB + 1); i++) {
                priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
                if (!priv->rx_urb[i])
-                       return -ENOMEM;
+                       goto out_release_mem;

                priv->rx_urb[i]->transfer_buffer =
                        kmalloc(RX_URB_SIZE, GFP_KERNEL);
                if (!priv->rx_urb[i]->transfer_buffer)
-                       return -ENOMEM;
+                       goto out_release_mem;

                priv->rx_urb[i]->transfer_buffer_length = RX_URB_SIZE;
        }
@@ -1666,9 +1667,13 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
                void *oldaddr, *newaddr;

                priv->rx_urb[16] = usb_alloc_urb(0, GFP_KERNEL);
+               if (!priv->rx_urb[16])
+                       goto out_release_mem;
+
                priv->oldaddr = kmalloc(16, GFP_KERNEL);
                if (!priv->oldaddr)
-                       return -ENOMEM;
+                       goto out_release_mem;
+
                oldaddr = priv->oldaddr;
                align = ((long)oldaddr) & 3;
                if (align) {
@@ -1686,17 +1691,19 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
[Quoted text hidden]
```

Dan Carpenter <dan.carpenter@oracle.com> Sat, Jun 16, 2018 at 4:11 AM
To: Zhouyang Jia <jiazhouyang09@gmail.com>
Cc: devel@driverdev.osuosl.org, Kees Cook <keescook@chromium.org>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, linux-kernel@vger.kernel.org, Jia-Ju Bai <baijiaju1990@gmail.com>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>

I was actually OK with v1 on the theory that everything else leaked and
so this didn't really introduce anything new... :P

On Sat, Jun 16, 2018 at 10:01:22AM +0800, Zhouyang Jia wrote:
> --- a/drivers/staging/rtl8192u/r8192U_core.c
> +++ b/drivers/staging/rtl8192u/r8192U_core.c
> @@ -1639,8 +1639,9 @@ short rtl8192_tx(struct net_device *dev, struct sk_buff *skb)
> static short rtl8192_usb_initendpoints(struct net_device *dev)
> {
>   struct r8192_priv *priv = ieee80211_priv(dev);
> +   int i;
>
> -   priv->rx_urb = kmalloc(sizeof(struct urb *) * (MAX_RX_URB + 1),
> +   priv->rx_urb = kcalloc(MAX_RX_URB + 1, sizeof(struct urb *),
>   GFP_KERNEL);
>   if (!priv->rx_urb)
>   return -ENOMEM;
> @@ -1649,12 +1650,12 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
>   for (i = 0; i < (MAX_RX_URB + 1); i++) {
>   priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
>   if (!priv->rx_urb[i])
> -   return -ENOMEM;
> +   goto out_release_mem;
>
>   priv->rx_urb[i]->transfer_buffer =
>   kmalloc(RX_URB_SIZE, GFP_KERNEL);

You need to free priv->rx_urb[i]->transfer_buffer as well and there are
several other resources which are also not freed.

regards,
dan carpenter

Zhouyang Jia <jiazhouyang09@gmail.com> Sat, Jun 16, 2018 at 11:47 AM
Cc: Zhouyang Jia <jiazhouyang09@gmail.com>, Greg Kroah-Hartman <gregkh@linuxfoundation.org>, Christophe JAILLET <christophe.jaillet@wanadoo.fr>, Shreeya Patel <shreeya.patel23498@gmail.com>, Colin Ian King <colin.king@canonical.com>, Jia-Ju Bai <baijiaju1990@gmail.com>, Kees Cook <keescook@chromium.org>, devel@driverdev.osuosl.org, linux-kernel@vger.kernel.org

When usb_alloc_urb fails, the lack of error-handling code may
cause unexpected results.

This patch adds error-handling code after calling usb_alloc_urb,
and fixes memory leaks in error paths.

Signed-off-by: Zhouyang Jia <jiazhouyang09@gmail.com>
---
v1->v2:
- Fix memory leak.
v2->v3:
- Release memory in error path.
v3->v4:
- Use kcalloc instead of kmalloc_array.
v4->v5:
- Free priv->rx_urb[i]->transfer_buffer and priv->oldaddr.
---
 drivers/staging/rtl8192u/r8192U_core.c | 34 ++++++++++++++++++++++----------
 1 file changed, 24 insertions(+), 10 deletions(-)

diff --git a/drivers/staging/rtl8192u/r8192U_core.c b/drivers/staging/rtl8192u/r8192U_core.c
index 7a0dbc0..9413f29 100644
--- a/drivers/staging/rtl8192u/r8192U_core.c
+++ b/drivers/staging/rtl8192u/r8192U_core.c
@@ -1639,8 +1639,9 @@ short rtl8192_tx(struct net_device *dev, struct sk_buff *skb)
 static short rtl8192_usb_initendpoints(struct net_device *dev)
 {
        struct r8192_priv *priv = ieee80211_priv(dev);
+   int i;

-       priv->rx_urb = kmalloc(sizeof(struct urb *) * (MAX_RX_URB + 1),
+   priv->rx_urb = kcalloc(MAX_RX_URB + 1, sizeof(struct urb *),
                               GFP_KERNEL);
        if (!priv->rx_urb)
                return -ENOMEM;
@@ -1649,12 +1650,12 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
        for (i = 0; i < (MAX_RX_URB + 1); i++) {
                priv->rx_urb[i] = usb_alloc_urb(0, GFP_KERNEL);
                if (!priv->rx_urb[i])
-                       return -ENOMEM;
+   goto out_release_urb;

                priv->rx_urb[i]->transfer_buffer =
                        kmalloc(RX_URB_SIZE, GFP_KERNEL);
                if (!priv->rx_urb[i]->transfer_buffer)
-                       return -ENOMEM;
+   goto out_release_urb;

                priv->rx_urb[i]->transfer_buffer_length = RX_URB_SIZE;
        }
@@ -1666,9 +1667,13 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
                void *oldaddr, *newaddr;

                priv->rx_urb[16] = usb_alloc_urb(0, GFP_KERNEL);
+   if (!priv->rx_urb[16])
+   goto out_release_urb;
+
                priv->oldaddr = kmalloc(16, GFP_KERNEL);
                if (!priv->oldaddr)
-                       return -ENOMEM;
+   goto out_release_urb;
+
                oldaddr = priv->oldaddr;
                align = ((long)oldaddr) & 3;
                if (align) {
@@ -1686,17 +1691,26 @@ static short rtl8192_usb_initendpoints(struct net_device *dev)
        priv->pp_rxskb = kcalloc(MAX_RX_URB, sizeof(struct sk_buff *),
                                 GFP_KERNEL);
        if (!priv->pp_rxskb) {
-               kfree(priv->rx_urb);
-
-               priv->pp_rxskb = NULL;
-               priv->rx_urb = NULL;
-
                DMESGE("Endpoint Alloc Failure");
-               return -ENOMEM;
+   goto out_release_oldaddr;
        }

        netdev_dbg(dev, "End of initendpoints\n");
        return 0;
+
+out_release_oldaddr:
+   kfree(priv->oldaddr);
+
+out_release_urb:
```
```

```
+	for (i = 0; i < (MAX_RX_URB + 1); i++) {
+		if (priv->rx_urb[i]) {
+			kfree(priv->rx_urb[i]->transfer_buffer);
+			kfree(priv->rx_urb[i]);
+		}
+	}
```
[Quoted text hidden]