Subject: [PATCH] Copy mac_len in skb_clone() as well
Posted by Alexey Dobriyan on Wed, 14 Mar 2007 13:00:04 GMT
View Forum Message <> Reply to Message

ANK says: "It is rarely used, that's wy it was not noticed.
But in the places, where it is used, it should be disaster."

Signed-off-by: Alexey Dobriyan <adobriyan@sw.ru>
---

 net/core/skbuff.c | 1 +
 1 file changed, 1 insertion(+)

--- a/net/core/skbuff.c
+++ b/net/core/skbuff.c
@@ -463,6 +463,7 @@ #endif
  memcpy(n->cb, skb->cb, sizeof(skb->cb));
  C(len);
  C(data_len);
+ C(mac_len);
  C(csum);
  C(local_df);
  n->cloned = 1;

Subject: Re: [PATCH] Copy mac_len in skb_clone() as well
Posted by davem on Thu, 15 Mar 2007 10:02:35 GMT
View Forum Message <> Reply to Message

From: Alexey Dobriyan <adobriyan@sw.ru>
Date: Wed, 14 Mar 2007 16:07:11 +0300

> ANK says: "It is rarely used, that's wy it was not noticed.
> But in the places, where it is used, it should be disaster."
>
> Signed-off-by: Alexey Dobriyan <adobriyan@sw.ru>

Applied.

What bug triggered that helped you discover this? Or is it
merely from a code audit?

Thanks.

Subject: Re: Re: [PATCH] Copy mac_len in skb_clone() as well
Posted by dev on Thu, 15 Mar 2007 10:19:00 GMT

David Miller wrote:
> From: Alexey Dobriyan <adobriyan@sw.ru>
> Date: Wed, 14 Mar 2007 16:07:11 +0300
>
>
>>ANK says: "It is rarely used, that's wy it was not noticed.
>>But in the places, where it is used, it should be disaster."
>>
>>Signed-off-by: Alexey Dobriyan <adobriyan@sw.ru>
>
>
> Applied.
>
> What bug triggered that helped you discover this? Or is it
> merely from a code audit?
Ohhh, it is a fairy-tale to tell the truth :)
We had some unexplainable problems with java application in OpenVZ kernel.
It didn't work sometimes, but worked fine (!) with CONFIG_SLAB_DEBUG.
Alexey blamed java :), but ...
Then we found that poising one of the bits in slab cache was curing it.
After that we found that the problem is related to fclone cache.
And then we found that not all the fields are initialized during cloning.
The bug was related to our own skb->field we introduced,
but we analyzed the code and found this as well.

Thanks,
Kirill

Subject: Re: [PATCH] Copy mac_len in skb_clone() as well
Posted by Alexey Kuznetsov on Thu, 15 Mar 2007 16:04:36 GMT

Hello!

> What bug triggered that helped you discover this? Or is it
> merely from a code audit?

I asked the same question. :-)

openvz added some another fields to skbuff and when it was found
that they are lost while clone, he tried to figure out how all this works
and looked for another examples of this kind.

As I understand, the problem can be seen only in xfrmX_tunnel_input.
If uninitialized mac_len obtained from slab is more than current head room

it could corrupt memory.

Also, it looks like the fix is incomplete. copy_skb_header() also does not
copy this field. But it will be initialized to 0 by alloc_skb in this case
and xfrmX_tunnel_input() just will not copy mac header.

Alexey

Subject: Re: Re: [PATCH] Copy mac_len in skb_clone() as well
Posted by davem on Fri, 16 Mar 2007 01:08:34 GMT
View Forum Message <> Reply to Message

From: Kirill Korotaev <dev@sw.ru>
Date: Thu, 15 Mar 2007 13:33:12 +0300

> David Miller wrote:
> > From: Alexey Dobriyan <adobriyan@sw.ru>
> > Date: Wed, 14 Mar 2007 16:07:11 +0300
> >
> >
> > >>ANK says: "It is rarely used, that's wy it was not noticed.
> > >>But in the places, where it is used, it should be disaster."
> > >>
> > >>Signed-off-by: Alexey Dobriyan <adobriyan@sw.ru>
> >
> >
> > Applied.
> >
> > What bug triggered that helped you discover this? Or is it
> > merely from a code audit?
> Ohhh, it is a fairy-tale to tell the truth :)
> We had some unexplainable problems with java application in OpenVZ kernel.
> It didn't work sometimes, but worked fine (!) with CONFIG_SLAB_DEBUG.
> Alexey blamed java :), but ...
> Then we found that poising one of the bits in slab cache was curing it.
> After that we found that the problem is related to fclone cache.
> And then we found that not all the fields are initialized during cloning.
> The bug was related to our own skb->field we introduced,
> but we analyzed the code and found this as well.

Thanks for the detailed information.
