
Subject: [PATCH] Fix and allocate less memory for ->priv'less netdevices

Posted by [Alexey Dobriyan](#) on Fri, 18 Apr 2008 15:10:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

This patch effectively reverts commit d0498d9ae1a5cebac363e38907266d5cd2eedf89 aka "[NET]: Do not allocate unneeded memory for dev->priv alignment."
It was found to be buggy because of final unconditional += NETDEV_ALIGN_CONST removal.

For example, for sizeof(struct net_device) being 2048 bytes, "alloc_size" was also 2048 bytes, but allocator with debugging options turned on started giving out !32-byte aligned memory resulting in redzones overwrites.

Patch does small optimization in ->priv'less case: bumping size to next 32-byte boundary was always done to ensure ->priv will also be aligned. But, no ->priv, no need to do that.

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

net/core/dev.c | 15 ++++++++-----
1 file changed, 9 insertions(+), 6 deletions(-)

--- a/net/core/dev.c

+++ b/net/core/dev.c

@@ -3996,12 +3996,15 @@ struct net_device *alloc_netdev_mq(int sizeof_priv, const char *name,

BUG_ON(strlen(name) >= sizeof(dev->name));

- /* ensure 32-byte alignment of both the device and private area */

- alloc_size = (sizeof(*dev) + NETDEV_ALIGN_CONST +

- (sizeof(struct net_device_subqueue) * (queue_count - 1))) &

- ~NETDEV_ALIGN_CONST;

- if (sizeof_priv)

- alloc_size += sizeof_priv + NETDEV_ALIGN_CONST;

+ alloc_size = sizeof(struct net_device) +

+ sizeof(struct net_device_subqueue) * (queue_count - 1);

+ if (sizeof_priv) {

+ /* ensure 32-byte alignment of private area */

+ alloc_size = (alloc_size + NETDEV_ALIGN_CONST) & ~NETDEV_ALIGN_CONST;

+ alloc_size += sizeof_priv;

+ }

+ /* ensure 32-byte alignment of whole construct */

+ alloc_size += NETDEV_ALIGN_CONST;

p = kzalloc(alloc_size, GFP_KERNEL);

if (!p) {
