
Subject: Re: [PATCH] iptables 32bit compat layer
Posted by [Patrick McHardy](#) on Wed, 29 Mar 2006 09:28:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dmitry Mishin wrote:

- > This patch extends current iptables compatibility layer in order to get
- > 32bit iptables to work on 64bit kernel. Current layer is insufficient due to
- > alignment checks both in kernel and user space tools.
- >
- > Patch is for current net-2.6.17 with addition of move of ipt_entry_{match|
- > target} definitions to xt_entry_{match|target}.

Thanks, this looks good. Two small issues so far:

- > diff --git a/net/compat.c b/net/compat.c
- > index 13177a1..6a7028e 100644
- > --- a/net/compat.c
- > +++ b/net/compat.c
- > @@ -476,8 +476,7 @@ asmlinkage long compat_sys_setsockopt(in
- > int err;
- > struct socket *sock;
- >
- > - /* SO_SET_REPLACE seems to be the same in all levels */
- > - if (optname == IPT_SO_SET_REPLACE)
- > + if (level == SOL_IPV6 && optname == IPT_SO_SET_REPLACE)
- > return do_netfilter_replace(fd, level, optname,
- > optval, optlen);

I don't understand the reason for this change. If its not a mistake,
it would make more sense to check for IP6T_SO_SET_REPLACE I guess ..

- > +#ifdef CONFIG_COMPAT
- > +void xt_compat_lock(int af)
- > +{
- > + down(&xt[af].compat_mutex);
- > +}
- > +EXPORT_SYMBOL_GPL(xt_compat_lock);
- > +
- > +void xt_compat_unlock(int af)
- > +{
- > + up(&xt[af].compat_mutex);
- > +}
- > +EXPORT_SYMBOL_GPL(xt_compat_unlock);
- > +#endif

Won't a seperate compat-mutex introduce races between compat- and

non-compatible users? BTW, the up/down calls have been replaced by the new mutex API in Linus' tree, please resend the patch against the current tree.
