

Re: bind() udp behavior 2.6.8.1

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-12/3335.html>

From: Kyle Moffett (mrmacman_g4_at_mac.com)

Date: 12/15/04

Date: Tue, 14 Dec 2004 22:19:28 -0500

To: Adam Denenberg <adam@denberg.org>

On Dec 14, 2004, at 21:23, Adam Denenberg wrote:

> *i think you guys are all right. However there is one concern. Not
> clearing out a UDP connection in a firewall coming from a high port is
> indeed a security risk. Allowing a high numbered udp port to remain
> open for a prolonged period of time would definitely impose a security
> risk which is why the PIX is doing what it does. The linux server is
> "reusing" the same UDP high numbered socket however it is doing so
> exactly as the firewall is clearing its state table (60 ms) from the
> first connection which is what is causing the issue.*
>
> *I think a firewall ought to be aware of such behavior, but at the same
> time be secure enough to not just leave high numbered udp ports wide
> open for attack. I am trying to find out why the PIX chose 60 ms to
> clear out the UDP state table. I think that is a random number and
> probably too short of a span for this to occur however i am still
> researching it.*
>
> *Any other insight would be greatly appreciated.*

60ms is certainly *_way_* too small for most UDP traffic. With something like

that, OpenAFS would die almost immediately. I think the current OpenAFS minimum is like 20 minutes, although somebody patched the OpenAFS source to send a keepalive every 5 minutes, so it could be reduced.

OTOH,

sending a keepalive every 60ms would take a *_massive_* amount of bandwidth even for one client, think about a couple hundred :-D. Heck, I've

even seen pings on a regular basis that take longer than 60ms, which means that even an infinitely fast kerberos server wouldn't respond quickly enough :-D.

Cheers,
Kyle Moffett

-----BEGIN GEEK CODE BLOCK-----

Linux-Kernel: Re: bind() udp behavior 2.6.8.1

Version: 3.12

GCM/CS/IT/U d- s++: a18 C++++>\$ UB/L/X/*++++(+)>\$ P+++(++++)>\$
L++++(++++) E W++(+) N+++((++)) o? K? w--- O? M++ V? PS+() PE+(-) Y+
PGP+++ t+(++++) 5 X R? tv-((-) b++++(++)) DI+ D+ G e->++++\$ h!*()>+\$ r
!y?(-)

-----END GEEK CODE BLOCK-----

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>