Re: [RFC,PATCH] loopback: calls netif_receive_skb() instead of netif_rx()

Source: http://linux.derkeiler.com/Mailing−Lists/Kernel/2008−03/msg11931.html

- *From*: Ingo Molnar <mingo@xxxxxxx>
- *Date*: Mon, 31 Mar 2008 12:12:37 +0200

* Eric Dumazet <dada1@xxxxxxxxxxxxx> wrote:

> Problem is to check available space :
>
> It depends on stack growing UP or DOWN, and depends on caller running
> on process stack, or softirq stack, or even hardirq stack.

ok − i wish such threads were on lkml so that everyone not just the
netdev kabal can read it. It's quite ugly, but if we want to check stack
free space i'd suggest for you to put a stack_can_recurse() call into
arch/x86/kernel/process.c and offer a default __weak implementation in
kernel/fork.c that always returns 0.

the rule on x86 should be something like this: on 4K stacks and 64−bit
[which have irqstacks] free stack space can go as low as 25%. On 8K
stacks [which doesnt have irqstacks but nests irqs] it should not go
below 50% before falling back to the explicitly queued packet branch.

this way other pieces of kernel code code can choose between on−stack
fast recursion and explicit iterators. Although i'm not sure i like the
whole concept to begin with ...

Ingo
−−
To unsubscribe from this list: send the line "unsubscribe linux−kernel" in
the body of a message to majordomo@xxxxxxxxxxxxxxx
More majordomo info at http://vger.kernel.org/majordomo−info.html
Please read the FAQ at http://www.tux.org/lkml/