
Subject: Re: [PATCH nf-2.6.22] [netfilter] early_drop improvement

Posted by [Eric Dumazet](#) on Sat, 07 Apr 2007 12:08:47 GMT

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

Vasily Averin a e'crit :

> When the number of conntracks is reached nf_conntrack_max limit, early_drop() is
> called and tries to free one of already used conntracks in one of the hash
> buckets. If it does not find any conntracks that may be freed, it
> leads to transmission errors.
> However it is not fair because of current hash bucket may be empty but the
> neighbour ones can have the number of conntracks that can be freed. On the other
> hand the number of checked conntracks is not limited and it can cause a long delay.
> The following patch limits the number of checked conntracks by average number of
> conntracks in one hash bucket and allows to search conntracks in other hash buckets.

Hi Vasily

```
>  
> atomic_inc(&ct->ct_general.use);  
> break;  
> }  
> + if (!--(*cnt)) {  
> +   dropped = 1;  
> +   break;  
> + }
```

```
> + cnt = (nf_conntrack_max/nf_conntrack_htable_size) + 1;
```

I am sorry but this wont help in the case you mentioned in an earlier mail :

If nf_conntrack_max < nf_conntrack_htable_size, cnt will be set to 1.

Then in __early_drop() you endup in breaking the list_for_each_entry_reverse() loop after the first element was tested ! Not what you intended I'm afraid, because you wont event scan the whole chain as before your patch :(

I believe you should not test --cnt in __early_drop() but in the caller.

(That is not counting the number of found cells, but the number of hash chains you tried)
