
Subject: Re: [PATCH 5/12] L2 network namespace: IPv4 routing

Posted by [Mishin Dmitry](#) on Thu, 07 Dec 2006 10:05:41 GMT

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

On Thursday 07 December 2006 12:58, Daniel Lezcano wrote:

> Dmitry Mishin wrote:

> > On Thursday 07 December 2006 01:40, Daniel Lezcano wrote:

> > > Dmitry Mishin wrote:

> > > > Make FIBs per-namespace and adds additional key (net namespace) to lookups in

> > > > routing cache.

> > > >

> > > > Signed-off-by: Dmitry Mishin <dim@openvz.org>

> > > >

> > > > ---

> > > > include/linux/net_namespace.h | 12 ++++

> > > > include/net/flow.h | 3 +

> > > > include/net/ip_fib.h | 48 ++++++++-----

> > > > net/core/fib_rules.c | 43 ++++++++---

> > > > net/core/net_namespace.c | 9 +++

> > > > net/ipv4/fib_frontend.c | 121 ++++++++-----

> > > > net/ipv4/fib_hash.c | 13 +++-

> > > > net/ipv4/fib_rules.c | 86 ++++++++-----

> > > > net/ipv4/fib_semantics.c | 100 ++++++++-----

> > > > net/ipv4/fib_trie.c | 18 ++++-

> > > > net/ipv4/route.c | 29 ++++++++-

> > > > 11 files changed, 391 insertions(+), 91 deletions(-)

> > > >

> > > > --- linux-2.6.19-rc6-mm2.orig/include/linux/net_namespace.h

> > > > +++ linux-2.6.19-rc6-mm2/include/linux/net_namespace.h

> > > > @@ -11,6 +11,18 @@ struct net_namespace {

> > > > struct net_device *dev_base_p, **dev_tail_p;

> > > > struct net_device *loopback_dev_p;

> > > > struct pcpu_lstats *pcpu_lstats_p;

> > > > + #ifndef CONFIG_IP_MULTIPLE_TABLES

> > > > + struct fib_table *fib4_local_table, *fib4_main_table;

> > > > + #else

> > > > + struct list_head fib_rules_ops_list;

> > > > + struct fib_rules_ops *fib4_rules_ops;

> > > > + #endif

> > > > + struct hlist_head *fib4_tables;

> > > > + struct hlist_head *fib4_hash, *fib4_laddrhash;

> > > > + unsigned fib4_hash_size, fib4_info_cnt;

> > > > + #ifdef CONFIG_IP_FIB_TRIE

> > > > + int fib4_trie_last_dflt;

> > > > + #endif

> > > > unsigned int hash;

> > > > };

> > > >

```

> >>> --- linux-2.6.19-rc6-mm2.orig/include/net/flow.h
> >>> +++ linux-2.6.19-rc6-mm2/include/net/flow.h
> >>> @@ -82,6 +82,9 @@ struct flowi {
> >>> #define fl_mh_type uli_u.mht.type
> >>> #endif
> >>> __u32 secid; /* used by xfrm; see secid.txt */
> >>> #ifdef CONFIG_NET_NS
> >>> + struct net_namespace *net_ns;
> >>> #endif
> >>> } __attribute__((__aligned__(BITS_PER_LONG/8)));
> >>>
> >>> #define FLOW_DIR_IN 0
> >>> --- linux-2.6.19-rc6-mm2.orig/include/net/ip_fib.h
> >>> +++ linux-2.6.19-rc6-mm2/include/net/ip_fib.h
> >>> @@ -18,6 +18,7 @@
> >>>
> >>> #include <net/flow.h>
> >>> #include <linux/seq_file.h>
> >>> #include <linux/net_namespace.h>
> >>> #include <net/fib_rules.h>
> >>>
> >>> struct fib_config {
> >>> @@ -171,14 +172,21 @@ struct fib_table {
> >>>
> >>> #ifndef CONFIG_IP_MULTIPLE_TABLES
> >>>
> >>> -extern struct fib_table *ip_fib_local_table;
> >>> -extern struct fib_table *ip_fib_main_table;
> >>> #ifdef CONFIG_NET_NS
> >>> +extern struct fib_table *ip_fib_local_table_static;
> >>> +extern struct fib_table *ip_fib_main_table_static;
> >>> #define ip_fib_local_table_ns() ip_fib_local_table_static
> >>> #define ip_fib_main_table_ns() ip_fib_main_table_static
> >>> #else
> >>> #define ip_fib_local_table_ns() (current_net_ns->fib4_local_table)
> >>> #define ip_fib_main_table_ns() (current_net_ns->fib4_main_table)
> >>> #endif
> >> Why don't you use :
> >> #ifndef CONFIG_NET_NS
> >> extern struct fib_table *ip_fib_local_table
> >> extern struct fib_table *ip_fib_main_table
> >> #else
> >> #define ip_fib_local_table (current_net_ns->fib4_local_table)
> >> #define ip_fib_main_table (current_net_ns->fib4_main_table)
> >> #endif
> >>
> >> You don't need to replace all ip_fib_local_table by
> >> ip_fib_local_table_ns. It will reduce a lot the code impact, no ?

```

> > But this also make code more foggy. All developers will continue to think,
> > that ip_fib_local_table IS static variable, while after such change it is
> > not true. So, rename it for more clear code.

>

> In other way, if net_ns is disabled, all developpers will continue to
> think that ip_fib_local_table_ns is non-static :)

No, because it is new entity. As it is unknown, they should to check what is it
and remember, that this is complex macros, which changes it's value depending
on enabled CONFIG_NET_NS or not.

It's just our experience in debugging virtualization code, that if you rename
reloaded entity, than it is easier to debug code with it, while if you don't
rename, it is easier to port to new kernel versions.

--

Thanks,
Dmitry.

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>
