
Subject: [patch 16/20] [Network namespace] Add loopback isolation.

Posted by [Daniel Lezcano](#) on Sun, 10 Dec 2006 21:58:33 GMT

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

Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>

```
include/linux/net_namespace.h | 7 +++++--
include/linux/skbuff.h        | 5 ++++-
net/core/net_namespace.c      | 13 ++++++++----
net/ipv4/ip_input.c           | 4 ++--
net/ipv4/ip_output.c          | 2 ++
5 files changed, 23 insertions(+), 8 deletions(-)
```

Index: 2.6.19-rc6-mm2/include/linux/skbuff.h

=====

--- 2.6.19-rc6-mm2.orig/include/linux/skbuff.h

+++ 2.6.19-rc6-mm2/include/linux/skbuff.h

```
@@ -227,6 +227,7 @@ enum {
 * @dma_cookie: a cookie to one of several possible DMA operations
 * done by skb DMA functions
 * @secmark: security marking
+ * @net_ns: namespace destination
 */
```

```
struct sk_buff {
@@ -311,7 +312,9 @@ struct sk_buff {
#ifdef CONFIG_NETWORK_SECMARK
__u32 secmark;
#endif
-
+#ifdef CONFIG_NET_NS
+ struct net_namespace *net_ns;
+#endif
__u32 mark;
```

```
/* These elements must be at the end, see alloc_skb() for details. */
```

Index: 2.6.19-rc6-mm2/net/ipv4/ip_input.c

=====

--- 2.6.19-rc6-mm2.orig/net/ipv4/ip_input.c

+++ 2.6.19-rc6-mm2/net/ipv4/ip_input.c

```
@@ -399,9 +399,9 @@ int ip_rcv(struct sk_buff *skb, struct n
 iph = skb->nh.iph;
```

```
#ifdef CONFIG_NET_NS
- dst_net_ns = net_ns_find_from_dest_addr(iph->daddr);
+ dst_net_ns = net_ns_find_from_dest_addr(skb);
```

```

    if (dst_net_ns && net_ns != dst_net_ns)
-   push_net_ns(dst_net_ns, net_ns);
+   net_ns = push_net_ns(dst_net_ns);
#endif
/*
 * RFC1122: 3.1.2.2 MUST silently discard any IP frame that fails the checksum.

```

Index: 2.6.19-rc6-mm2/net/ipv4/ip_output.c

```

=====
--- 2.6.19-rc6-mm2.orig/net/ipv4/ip_output.c
+++ 2.6.19-rc6-mm2/net/ipv4/ip_output.c
@@ -277,9 +277,11 @@ int ip_mc_output(struct sk_buff *skb)
int ip_output(struct sk_buff *skb)
{
    struct net_device *dev = skb->dst->dev;
+   struct net_namespace *net_ns = current_net_ns;

    IP_INC_STATS(IPSTATS_MIB_OUTREQUESTS);

+   skb->net_ns = net_ns;
    skb->dev = dev;
    skb->protocol = htons(ETH_P_IP);

```

Index: 2.6.19-rc6-mm2/include/linux/net_namespace.h

```

=====
--- 2.6.19-rc6-mm2.orig/include/linux/net_namespace.h
+++ 2.6.19-rc6-mm2/include/linux/net_namespace.h
@@ -7,6 +7,7 @@
#include <linux/types.h>

struct in_ifaddr;
+struct sk_buff;

struct net_ns_net_device {
    struct hlist_head *name_head;
@@ -96,7 +97,8 @@ extern int net_ns_check_bind(int addr_ty
extern __be32 net_ns_select_source_address(const struct net_device *dev,
    u32 dst, int scope);

-extern struct net_namespace *net_ns_find_from_dest_addr(u32 daddr);
+extern struct net_namespace
+*net_ns_find_from_dest_addr(const struct sk_buff *skb);

extern int net_ns_ifa_is_visible(const struct in_ifaddr *ifa);

@@ -164,7 +166,8 @@ static inline __be32 net_ns_select_sourc
    return 0;
}

```

```

-static inline struct net_namespace *net_ns_find_from_dest_addr(u32 daddr)
+static inline struct net_namespace
+*net_ns_find_from_dest_addr(const struct sk_buff *skb)
{
    return current_net_ns;
}

```

Index: 2.6.19-rc6-mm2/net/core/net_namespace.c

--- 2.6.19-rc6-mm2.orig/net/core/net_namespace.c

+++ 2.6.19-rc6-mm2/net/core/net_namespace.c

@ @ -14,6 +14,8 @ @

#include <net/ip_fib.h>

#include <linux/inetdevice.h>

#include <linux/in.h>

+#include <linux/skbuff.h>

+#include <linux/ip.h>

```

static spinlock_t net_ns_list_lock = SPIN_LOCK_UNLOCKED;

```

@ @ -376,14 +378,19 @ @ out:

```

    return addr;

```

```

}

```

```

-struct net_namespace *net_ns_find_from_dest_addr(u32 daddr)
+struct net_namespace *net_ns_find_from_dest_addr(const struct sk_buff *skb)
{

```

```

    struct net_namespace *net_ns = NULL;
    struct net_device *dev;
    struct in_device *in_dev;
+ struct iphdr *iph;
+ __be32 daddr;

```

```

- if (LOOPBACK(daddr))
- return current_net_ns;
+ iph = skb->nh.iph;
+ daddr = iph->daddr;
+
+ if (LOOPBACK(daddr))
+ return skb->net_ns;

```

```

    read_lock(&dev_base_lock);
    rcu_read_lock();

```

```

--

```

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>
