

---

Subject: [PATCH 13/16] net: Make the device list and device lookups per namespace.

Posted by [ebiederm](#) on Sat, 08 Sep 2007 21:35:46 GMT

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

---

This patch makes most of the generic device layer network namespace safe. This patch makes dev\_base\_head a network namespace variable, and then it picks up a few associated variables. The functions:

```
dev_getbyhwaddr
dev_getfirsthwbytype
dev_get_by_flags
dev_get_by_name
__dev_get_by_name
dev_get_by_index
__dev_get_by_index
dev_ioctl
dev_ethtool
dev_load
wireless_process_ioctl
```

were modified to take a network namespace argument, and deal with it.

vlan\_ioctl\_set and briocli\_set were modified so their hooks will receive a network namespace argument.

So basically anything in the core of the network stack that was affected to by the change of dev\_base was modified to handle multiple network namespaces. The rest of the network stack was simply modified to explicitly use &init\_net the initial network namespace. This can be fixed when those components of the network stack are modified to handle multiple network namespaces.

For now the ifindex generator is left global.

Fundametally ifindex numbers are per namespace, or else we will have corner case problems with migration when we get that far.

At the same time there are assumptions in the network stack that the ifindex of a network device won't change. Making the ifindex number global seems a good compromise until the network stack can cope with ifindex changes when you change namespaces, and the like.

Signed-off-by: Eric W. Biederman <[ebiederm@xmission.com](mailto:ebiederm@xmission.com)>

---

```

arch/s390/appldata/appldata_net_sum.c | 3 +-
arch/sparc64/solaris/ioctl.c           | 3 +-
drivers/atm/idt77252.c                 | 2 +-
drivers/block/aoe/aoecmd.c             | 3 +-
drivers/infiniband/hw/cxgb3/cxio_hal.c | 3 +-
drivers/net/bonding/bond_main.c         | 2 +-
drivers/net/bonding/bond_sysfs.c        | 3 +-
drivers/net/eql.c                      | 9 +-
drivers/net/ifb.c                      | 3 +-
drivers/net/macvlan.c                  | 2 +-
drivers/net/pppoe.c                    | 4 +-
drivers/net/shaper.c                   | 3 +-
drivers/net/tun.c                      | 3 +-
drivers/net/veth.c                     | 2 +-
drivers/net/wan/dlci.c                 | 4 +-
drivers/net/wan/sbni.c                 | 3 +-
drivers/net/wireless/strip.c           | 2 +-
drivers/parisc/led.c                   | 2 +-
fs/afs/netdevices.c                   | 5 +-
include/linux/if_bridge.h              | 2 +-
include/linux/if_vlan.h                | 2 +-
include/linux/netdevice.h              | 66 +++++-----
include/net/net_namespace.h            | 4 +
include/net/pkt_cls.h                  | 3 +-
include/net/rtnetlink.h                | 2 +-
include/net/wext.h                     | 15 ++-
net/802/tr.c                           | 2 +-
net/8021q/vlan.c                       | 6 +-
net/8021q/vlan_netlink.c               | 3 +-
net/8021q/vlanproc.c                   | 6 +-
net/appletalk/ddp.c                    | 6 +-
net/atm/mpc.c                          | 2 +-
net/ax25/af_ax25.c                     | 2 +-
net/bridge/br_if.c                     | 4 +-
net/bridge/br_ioctl.c                  | 7 +-
net/bridge/br_netlink.c                | 5 +-
net/bridge/br_private.h                | 2 +-
net/core/dev.c                         | 271 ++++++-----
net/core/dev_mcast.c                   | 41 +++++-
net/core/ethtool.c                     | 4 +-
net/core/fib_rules.c                   | 4 +-
net/core/neighbour.c                   | 6 +-
net/core/netpoll.c                     | 2 +-
net/core/pktgen.c                      | 2 +-
net/core/rtnetlink.c                   | 35 +++--
net/core/sock.c                        | 3 +-
net/decnet/af_decnet.c                  | 2 +-
net/decnet/dn_dev.c                    | 20 ++--

```

net/deccnet/dn_fib.c		8 +-
net/deccnet/dn_route.c		6 +-
net/deccnet/sysctl_net_deccnet.c		4 +-
net/econet/af_econet.c		2 +-
net/ipv4/arp.c		4 +-
net/ipv4/devinet.c		18 +-
net/ipv4/fib_frontend.c		2 +-
net/ipv4/fib_semantics.c		4 +-
net/ipv4/icmp.c		2 +-
net/ipv4/igmp.c		4 +-
net/ipv4/ip_fragment.c		2 +-
net/ipv4/ip_gre.c		4 +-
net/ipv4/ip_sockglue.c		2 +-
net/ipv4/ipconfig.c		2 +-
net/ipv4/ipip.c		4 +-
net/ipv4/ipmr.c		4 +-
net/ipv4/ipvs/ip_vs_sync.c		10 +-
net/ipv4/netfilter/ipt_CLUSTERIP.c		2 +-
net/ipv4/route.c		4 +-
net/ipv6/addrconf.c		28 +++
net/ipv6/af_inet6.c		2 +-
net/ipv6/anycast.c		12 +-
net/ipv6/datagram.c		2 +-
net/ipv6/ip6_tunnel.c		6 +-
net/ipv6/ip6_sockglue.c		2 +-
net/ipv6/mcast.c		12 +-
net/ipv6/raw.c		2 +-
net/ipv6/reassembly.c		2 +-
net/ipv6/route.c		4 +-
net/ipv6/sit.c		4 +-
net/ipx/af_ipx.c		6 +-
net/irda/irnetlink.c		9 +-
net/lc/af_llc.c		4 +-
net/lc/llc_core.c		3 +-
net/mac80211/ieee80211.c		1 +
net/mac80211/ieee80211_cfg.c		3 +-
net/mac80211/tx.c		9 +-
net/mac80211/util.c		7 +-
net/netrom/nr_route.c		6 +-
net/packet/af_packet.c		18 +-
net/rose/rose_route.c		8 +-
net/sched/act_mirred.c		3 +-
net/sched/cls_api.c		4 +-
net/sched/em_meta.c		2 +-
net/sched/sch_api.c		10 +-
net/sctp/ipv6.c		4 +-
net/sctp/protocol.c		2 +-
net/socket.c		22 +-

```

net/tipc/eth_media.c      |  2 +-
net/wireless/wext.c       | 38 +++++
net/x25/x25_route.c      |  2 +-
99 files changed, 555 insertions(+), 362 deletions(-)

```

```

diff --git a/arch/s390/appldata/appldata_net_sum.c b/arch/s390/appldata/appldata_net_sum.c
index 2180ac1..6c1815a 100644

```

```

--- a/arch/s390/appldata/appldata_net_sum.c
+++ b/arch/s390/appldata/appldata_net_sum.c

```

```

@@ -16,6 +16,7 @@

```

```

#include <linux/errno.h>
#include <linux/kernel_stat.h>
#include <linux/netdevice.h>
+#include <net/net_namespace.h>

```

```

#include "appldata.h"

```

```

@@ -107,7 +108,7 @@ static void appldata_get_net_sum_data(void *data)

```

```

    tx_dropped = 0;
    collisions = 0;
    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    stats = dev->get_stats(dev);
    rx_packets += stats->rx_packets;
    tx_packets += stats->tx_packets;

```

```

diff --git a/arch/sparc64/solaris/ioctl.c b/arch/sparc64/solaris/ioctl.c

```

```

index 18352a4..8ad10a6 100644

```

```

--- a/arch/sparc64/solaris/ioctl.c

```

```

+++ b/arch/sparc64/solaris/ioctl.c

```

```

@@ -28,6 +28,7 @@

```

```

#include <linux/compat.h>

```

```

#include <net/sock.h>
+#include <net/net_namespace.h>

```

```

#include <asm/uaccess.h>

```

```

#include <asm/termios.h>

```

```

@@ -686,7 +687,7 @@ static inline int solaris_i(unsigned int fd, unsigned int cmd, u32 arg)
    int i = 0;

```

```

    read_lock_bh(&dev_base_lock);
- for_each_netdev(d)
+ for_each_netdev(&init_net, d)
    i++;
    read_unlock_bh(&dev_base_lock);

```

```

diff --git a/drivers/atm/idt77252.c b/drivers/atm/idt77252.c

```

```

index f8b1700..eee54c0 100644
--- a/drivers/atm/idt77252.c
+++ b/drivers/atm/idt77252.c
@@ -3576,7 +3576,7 @@ init_card(struct atm_dev *dev)
    * XXX: <hack>
    */
    sprintf(tname, "eth%d", card->index);
- tmp = dev_get_by_name(tname); /* jhs: was "tmp = dev_get(tname);" */
+ tmp = dev_get_by_name(&init_net, tname); /* jhs: was "tmp = dev_get(tname);" */
    if (tmp) {
        memcpy(card->atmdev->esi, tmp->dev_addr, 6);

diff --git a/drivers/block/aoe/aoecmd.c b/drivers/block/aoe/aoecmd.c
index 01fbdd3..30394f7 100644
--- a/drivers/block/aoe/aoecmd.c
+++ b/drivers/block/aoe/aoecmd.c
@@ -9,6 +9,7 @@
#include <linux/skbuff.h>
#include <linux/netdevice.h>
#include <linux/genhd.h>
+#include <net/net_namespace.h>
#include <asm/unaligned.h>
#include "aoe.h"

@@ -194,7 +195,7 @@ aoecmd_cfg_pkts(ushort aoemajor, unsigned char aoeminor, struct
sk_buff **tail)
    sl = sl_tail = NULL;

    read_lock(&dev_base_lock);
- for_each_netdev(ifp) {
+ for_each_netdev(&init_net, ifp) {
    dev_hold(ifp);
    if (!is_aoe_netif(ifp))
        goto cont;

diff --git a/drivers/infiniband/hw/cxgb3/cxio_hal.c b/drivers/infiniband/hw/cxgb3/cxio_hal.c
index 1518b41..7b92b7d 100644
--- a/drivers/infiniband/hw/cxgb3/cxio_hal.c
+++ b/drivers/infiniband/hw/cxgb3/cxio_hal.c
@@ -37,6 +37,7 @@
#include <linux/spinlock.h>
#include <linux/pci.h>
#include <linux/dma-mapping.h>
+#include <net/net_namespace.h>

#include "cxio_resource.h"
#include "cxio_hal.h"
@@ -894,7 +895,7 @@ int cxio_rdev_open(struct cxio_rdev *rdev_p)
    if (cxio_hal_find_rdev_by_name(rdev_p->dev_name)) {

```

```

    return -EBUSY;
}
- netdev_p = dev_get_by_name(rdev_p->dev_name);
+ netdev_p = dev_get_by_name(&init_net, rdev_p->dev_name);
if (!netdev_p) {
    return -EINVAL;
}
diff --git a/drivers/net/bonding/bond_main.c b/drivers/net/bonding/bond_main.c
index cf97d8a..559fe94 100644
--- a/drivers/net/bonding/bond_main.c
+++ b/drivers/net/bonding/bond_main.c
@@ -3719,7 +3719,7 @@ static int bond_do_ioctl(struct net_device *bond_dev, struct ifreq *ifr,
int cmd
}

    down_write(&(bonding_rwsem));
- slave_dev = dev_get_by_name(ifr->ifr_slave);
+ slave_dev = dev_get_by_name(&init_net, ifr->ifr_slave);

    dprintk("slave_dev=%p: \n", slave_dev);

diff --git a/drivers/net/bonding/bond_sysfs.c b/drivers/net/bonding/bond_sysfs.c
index 60cccf2..8289e27 100644
--- a/drivers/net/bonding/bond_sysfs.c
+++ b/drivers/net/bonding/bond_sysfs.c
@@ -35,6 +35,7 @@
#include <linux/ctype.h>
#include <linux/inet.h>
#include <linux/rtnetlink.h>
+#include <net/net_namespace.h>

/* #define BONDING_DEBUG 1 */
#include "bonding.h"
@@ -299,7 +300,7 @@ static ssize_t bonding_store_slaves(struct device *d,
    read_unlock_bh(&bond->lock);
    printk(KERN_INFO DRV_NAME ": %s: Adding slave %s.\n",
           bond->dev->name, ifname);
- dev = dev_get_by_name(ifname);
+ dev = dev_get_by_name(&init_net, ifname);
if (!dev) {
    printk(KERN_INFO DRV_NAME
           ": %s: Interface %s does not exist!\n",
diff --git a/drivers/net/eql.c b/drivers/net/eql.c
index 102218c..f1cc66d 100644
--- a/drivers/net/eql.c
+++ b/drivers/net/eql.c
@@ -116,6 +116,7 @@
#include <linux/init.h>

```

```

#include <linux/timer.h>
#include <linux/netdevice.h>
+#include <net/net_namespace.h>

#include <linux/if.h>
#include <linux/if_arp.h>
@@ -412,7 +413,7 @@ static int eql_enslave(struct net_device *master_dev, slaving_request_t
__user *
    if (copy_from_user(&srq, srqp, sizeof (slaving_request_t)))
        return -EFAULT;

- slave_dev = dev_get_by_name(srq.slave_name);
+ slave_dev = dev_get_by_name(&init_net, srq.slave_name);
    if (slave_dev) {
        if ((master_dev->flags & IFF_UP) == IFF_UP) {
            /* slave is not a master & not already a slave: */
@@ -460,7 +461,7 @@ static int eql_emancipate(struct net_device *master_dev,
slaving_request_t __use
    if (copy_from_user(&srq, srqp, sizeof (slaving_request_t)))
        return -EFAULT;

- slave_dev = dev_get_by_name(srq.slave_name);
+ slave_dev = dev_get_by_name(&init_net, srq.slave_name);
    ret = -EINVAL;
    if (slave_dev) {
        spin_lock_bh(&eql->queue.lock);
@@ -493,7 +494,7 @@ static int eql_g_slave_cfg(struct net_device *dev, slave_config_t __user
*scp)
    if (copy_from_user(&sc, scp, sizeof (slave_config_t)))
        return -EFAULT;

- slave_dev = dev_get_by_name(sc.slave_name);
+ slave_dev = dev_get_by_name(&init_net, sc.slave_name);
    if (!slave_dev)
        return -ENODEV;

@@ -528,7 +529,7 @@ static int eql_s_slave_cfg(struct net_device *dev, slave_config_t __user
*scp)
    if (copy_from_user(&sc, scp, sizeof (slave_config_t)))
        return -EFAULT;

- slave_dev = dev_get_by_name(sc.slave_name);
+ slave_dev = dev_get_by_name(&init_net, sc.slave_name);
    if (!slave_dev)
        return -ENODEV;

diff --git a/drivers/net/ibf.c b/drivers/net/ibf.c
index f5c3598..b06c6db 100644

```

```

--- a/drivers/net/ifb.c
+++ b/drivers/net/ifb.c
@@ -34,6 +34,7 @@
#include <linux/init.h>
#include <linux/moduleparam.h>
#include <net/pkt_sched.h>
+#include <net/net_namespace.h>

#define TX_TIMEOUT (2*HZ)

@@ -97,7 +98,7 @@ static void ri_tasklet(unsigned long dev)
    stats->tx_packets++;
    stats->tx_bytes +=skb->len;

- skb->dev = __dev_get_by_index(skb->iif);
+ skb->dev = __dev_get_by_index(&init_net, skb->iif);
    if (!skb->dev) {
        dev_kfree_skb(skb);
        stats->tx_dropped++;
diff --git a/drivers/net/macvlan.c b/drivers/net/macvlan.c
index dc74d00..2de073d 100644
--- a/drivers/net/macvlan.c
+++ b/drivers/net/macvlan.c
@@ -376,7 +376,7 @@ static int macvlan_newlink(struct net_device *dev,
    if (!tb[IFLA_LINK])
        return -EINVAL;

- lowerdev = __dev_get_by_index(nla_get_u32(tb[IFLA_LINK]));
+ lowerdev = __dev_get_by_index(dev->nd_net, nla_get_u32(tb[IFLA_LINK]));
    if (lowerdev == NULL)
        return -ENODEV;

diff --git a/drivers/net/pppoe.c b/drivers/net/pppoe.c
index b4ba584..5f5f124 100644
--- a/drivers/net/pppoe.c
+++ b/drivers/net/pppoe.c
@@ -216,7 +216,7 @@ static inline struct pppox_sock *get_item_by_addr(struct sockaddr_pppox
*sp)
    struct net_device *dev;
    int ifindex;

- dev = dev_get_by_name(sp->sa_addr.pppoe.dev);
+ dev = dev_get_by_name(&init_net, sp->sa_addr.pppoe.dev);
    if (!dev)
        return NULL;
    ifindex = dev->ifindex;
@@ -603,7 +603,7 @@ static int pppoe_connect(struct socket *sock, struct sockaddr *uservaddr,

```



```

/* Don't re-bind if sid==0 */
if (sp->sa_addr.pppoe.sid != 0) {
- dev = dev_get_by_name(sp->sa_addr.pppoe.dev);
+ dev = dev_get_by_name(&init_net, sp->sa_addr.pppoe.dev);

    error = -ENODEV;
    if (!dev)
diff --git a/drivers/net/shaper.c b/drivers/net/shaper.c
index 4c3d98f..3773b38 100644
--- a/drivers/net/shaper.c
+++ b/drivers/net/shaper.c
@@ -86,6 +86,7 @@

#include <net/dst.h>
#include <net/arp.h>
+#include <net/net_namespace.h>

struct shaper_cb {
    unsigned long shapeclock; /* Time it should go out */
@@ -488,7 +489,7 @@ static int shaper_ioctl(struct net_device *dev, struct ifreq *ifr, int cmd)
{
    case SHAPER_SET_DEV:
    {
- struct net_device *them=__dev_get_by_name(ss->ss_name);
+ struct net_device *them=__dev_get_by_name(&init_net, ss->ss_name);
        if(them==NULL)
            return -ENODEV;
        if(sh->dev)
diff --git a/drivers/net/tun.c b/drivers/net/tun.c
index 62b2b30..691d264 100644
--- a/drivers/net/tun.c
+++ b/drivers/net/tun.c
@@ -62,6 +62,7 @@

#include <linux/if_ether.h>
#include <linux/if_tun.h>
#include <linux/crc32.h>
+#include <net/net_namespace.h>

#include <asm/system.h>
#include <asm/uaccess.h>
@@ -475,7 +476,7 @@ static int tun_set_iff(struct file *file, struct ifreq *ifr)
    !capable(CAP_NET_ADMIN))
    return -EPERM;
}
- else if (__dev_get_by_name(ifr->ifr_name))
+ else if (__dev_get_by_name(&init_net, ifr->ifr_name))
    return -EINVAL;
    else {

```

```

char *name;
diff --git a/drivers/net/veth.c b/drivers/net/veth.c
index 743fd62..9e6a746 100644
--- a/drivers/net/veth.c
+++ b/drivers/net/veth.c
@@ -346,7 +346,7 @@ static int veth_newlink(struct net_device *dev,
else
    snprintf(iframe, IFNAMSIZ, DRV_NAME "%d");

- peer = rtnl_create_link(iframe, &veth_link_ops, tbp);
+ peer = rtnl_create_link(dev->nd_net, iframe, &veth_link_ops, tbp);
if (IS_ERR(peer))
    return PTR_ERR(peer);

```

```

diff --git a/drivers/net/wan/dlci.c b/drivers/net/wan/dlci.c
index 61041d5..bc12810 100644
--- a/drivers/net/wan/dlci.c
+++ b/drivers/net/wan/dlci.c
@@ -361,7 +361,7 @@ static int dlci_add(struct dlci_add *dlci)

```

```

/* validate slave device */
- slave = dev_get_by_name(dlci->devname);
+ slave = dev_get_by_name(&init_net, dlci->devname);
if (!slave)
    return -ENODEV;

@@ -427,7 +427,7 @@ static int dlci_del(struct dlci_add *dlci)
int err;

```

```

/* validate slave device */
- master = __dev_get_by_name(dlci->devname);
+ master = __dev_get_by_name(&init_net, dlci->devname);
if (!master)
    return(-ENODEV);

```

```

diff --git a/drivers/net/wan/sbni.c b/drivers/net/wan/sbni.c
index 1cc18e7..8d7e01e 100644
--- a/drivers/net/wan/sbni.c
+++ b/drivers/net/wan/sbni.c
@@ -54,6 +54,7 @@
#include <linux/init.h>
#include <linux/delay.h>

+#include <net/net_namespace.h>
#include <net/arp.h>

#include <asm/io.h>

```

```

@@ -1361,7 +1362,7 @@ sbni_ioctl( struct net_device *dev, struct ifreq *ifr, int cmd )

    if (copy_from_user( slave_name, ifr->ifr_data, sizeof slave_name ))
        return -EFAULT;
- slave_dev = dev_get_by_name( slave_name );
+ slave_dev = dev_get_by_name(&init_net, slave_name );
    if( !slave_dev || !(slave_dev->flags & IFF_UP) ) {
        printk( KERN_ERR "%s: trying to enslave non-active "
            "device %s\n", dev->name, slave_name );
diff --git a/drivers/net/wireless/strip.c b/drivers/net/wireless/strip.c
index edb214e..904e548 100644
--- a/drivers/net/wireless/strip.c
+++ b/drivers/net/wireless/strip.c
@@ -1972,7 +1972,7 @@ static struct net_device *get_strip_dev(struct strip *strip_info)
    sizeof(zero_address))) {
    struct net_device *dev;
    read_lock_bh(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (dev->type == strip_info->dev->type &&
        !memcmp(dev->dev_addr,
            &strip_info->true_dev_addr,
diff --git a/drivers/parisc/led.c b/drivers/parisc/led.c
index e5d7ed9..a6d6b24 100644
--- a/drivers/parisc/led.c
+++ b/drivers/parisc/led.c
@@ -359,7 +359,7 @@ static __inline__ int led_get_net_activity(void)
    * for reading should be OK */
    read_lock(&dev_base_lock);
    rcu_read_lock();
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
        struct net_device_stats *stats;
        struct in_device *in_dev = __in_dev_get_rcu(dev);
        if (!in_dev || !in_dev->ifa_list)
diff --git a/fs/afs/netdevices.c b/fs/afs/netdevices.c
index fc27d4b..49f1894 100644
--- a/fs/afs/netdevices.c
+++ b/fs/afs/netdevices.c
@@ -8,6 +8,7 @@
#include <linux/inetdevice.h>
#include <linux/netdevice.h>
#include <linux/if_arp.h>
+#include <net/net_namespace.h>
#include "internal.h"

/*
@@ -23,7 +24,7 @@ int afs_get_MAC_address(u8 *mac, size_t maclen)

```

```

BUG();

rtnl_lock();
- dev = __dev_getfirstbyhwtype(ARPHRD_ETHER);
+ dev = __dev_getfirstbyhwtype(&init_net, ARPHRD_ETHER);
if (dev) {
    memcpy(mac, dev->dev_addr, maclen);
    ret = 0;
@@ -47,7 +48,7 @@ int afs_get_ipv4_interfaces(struct afs_interface *bufs, size_t maxbufs,
    ASSERT(maxbufs > 0);

rtnl_lock();
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (dev->type == ARPHRD_LOOPBACK && !wantloopback)
        continue;
    idev = __in_dev_get_rtnl(dev);
diff --git a/include/linux/if_bridge.h b/include/linux/if_bridge.h
index 4ff211d..99e3a1a 100644
--- a/include/linux/if_bridge.h
+++ b/include/linux/if_bridge.h
@@ -104,7 +104,7 @@ struct __fdb_entry

#include <linux/netdevice.h>

-extern void brioctl_set(int (*ioctl_hook)(unsigned int, void __user *));
+extern void brioctl_set(int (*ioctl_hook)(struct net *, unsigned int, void __user *));
extern struct sk_buff *(*br_handle_frame_hook)(struct net_bridge_port *p,
        struct sk_buff *skb);
extern int (*br_should_route_hook)(struct sk_buff **pskb);
diff --git a/include/linux/if_vlan.h b/include/linux/if_vlan.h
index f8443fd..976d4b1 100644
--- a/include/linux/if_vlan.h
+++ b/include/linux/if_vlan.h
@@ -62,7 +62,7 @@ struct vlan_hdr {
#define VLAN_VID_MASK 0xffff

/* found in socket.c */
-extern void vlan_ioctl_set(int (*hook)(void __user *));
+extern void vlan_ioctl_set(int (*hook)(struct net *, void __user *));

#define VLAN_NAME "vlan"

diff --git a/include/linux/netdevice.h b/include/linux/netdevice.h
index 8eeced0..ec90d1a 100644
--- a/include/linux/netdevice.h
+++ b/include/linux/netdevice.h
@@ -727,44 +727,48 @@ struct packet_type {

```

```
#include <linux/notifier.h>
```

```
extern struct net_device loopback_dev; /* The loopback */
```

```
-extern struct list_head dev_base_head; /* All devices */
```

```
extern rwlock_t dev_base_lock; /* Device list lock */
```

```
-#define for_each_netdev(d) \
```

```
- list_for_each_entry(d, &dev_base_head, dev_list)
```

```
-#define for_each_netdev_safe(d, n) \
```

```
- list_for_each_entry_safe(d, n, &dev_base_head, dev_list)
```

```
-#define for_each_netdev_continue(d) \
```

```
- list_for_each_entry_continue(d, &dev_base_head, dev_list)
```

```
-#define net_device_entry(lh) list_entry(lh, struct net_device, dev_list)
```

```
-
```

```
-static inline struct net_device *next_net_device(struct net_device *dev)
```

```
-{
```

```
- struct list_head *lh;
```

```
- lh = dev->dev_list.next;
```

```
- return lh == &dev_base_head ? NULL : net_device_entry(lh);
```

```
-}
```

```
+#define for_each_netdev(net, d) \
```

```
+ list_for_each_entry(d, &(net)->dev_base_head, dev_list)
```

```
+#define for_each_netdev_safe(net, d, n) \
```

```
+ list_for_each_entry_safe(d, n, &(net)->dev_base_head, dev_list)
```

```
+#define for_each_netdev_continue(net, d) \
```

```
+ list_for_each_entry_continue(d, &(net)->dev_base_head, dev_list)
```

```
+#define net_device_entry(lh) list_entry(lh, struct net_device, dev_list)
```

```
-static inline struct net_device *first_net_device(void)
```

```
-{
```

```
- return list_empty(&dev_base_head) ? NULL :
```

```
- net_device_entry(dev_base_head.next);
```

```
-}
```

```
+#define next_net_device(d) \
```

```
+({ \
```

```
+ struct net_device *dev = d; \
```

```
+ struct list_head *lh; \
```

```
+ struct net *net; \
```

```
+ \
```

```
+ net = dev->nd_net; \
```

```
+ lh = dev->dev_list.next; \
```

```
+ lh == &net->dev_base_head ? NULL : net_device_entry(lh); \
```

```
+})
```

```
+
```

```
+#define first_net_device(N) \
```

```
+({ \
```

```
+ struct net *NET = (N); \
```

```

+ list_empty(&NET->dev_base_head) ? NULL : \
+ net_device_entry(NET->dev_base_head.next); \
+))

extern int  netdev_boot_setup_check(struct net_device *dev);
extern unsigned long netdev_boot_base(const char *prefix, int unit);
-extern struct net_device *dev_getbyhwaddr(unsigned short type, char *hwaddr);
-extern struct net_device *dev_getfirstbyhwtype(unsigned short type);
-extern struct net_device *__dev_getfirstbyhwtype(unsigned short type);
+extern struct net_device *dev_getbyhwaddr(struct net *net, unsigned short type, char *hwaddr);
+extern struct net_device *dev_getfirstbyhwtype(struct net *net, unsigned short type);
+extern struct net_device *__dev_getfirstbyhwtype(struct net *net, unsigned short type);
extern void dev_add_pack(struct packet_type *pt);
extern void dev_remove_pack(struct packet_type *pt);
extern void __dev_remove_pack(struct packet_type *pt);

-extern struct net_device *dev_get_by_flags(unsigned short flags,
+extern struct net_device *dev_get_by_flags(struct net *net, unsigned short flags,
      unsigned short mask);
-extern struct net_device *dev_get_by_name(const char *name);
-extern struct net_device *__dev_get_by_name(const char *name);
+extern struct net_device *dev_get_by_name(struct net *net, const char *name);
+extern struct net_device *__dev_get_by_name(struct net *net, const char *name);
extern int dev_alloc_name(struct net_device *dev, const char *name);
extern int dev_open(struct net_device *dev);
extern int dev_close(struct net_device *dev);
@@ -776,8 +780,8 @@ extern void synchronize_net(void);
extern int register_netdevice_notifier(struct notifier_block *nb);
extern int unregister_netdevice_notifier(struct notifier_block *nb);
extern int call_netdevice_notifiers(unsigned long val, void *v);
-extern struct net_device *dev_get_by_index(int ifindex);
-extern struct net_device *__dev_get_by_index(int ifindex);
+extern struct net_device *dev_get_by_index(struct net *net, int ifindex);
+extern struct net_device *__dev_get_by_index(struct net *net, int ifindex);
extern int dev_restart(struct net_device *dev);
#ifdef CONFIG_NETPOLL_TRAP
extern int netpoll_trap(void);
@@ -993,8 +997,8 @@ extern int netif_rx_ni(struct sk_buff *skb);
#define HAVE_NETIF_RECEIVE_SKB 1
extern int netif_receive_skb(struct sk_buff *skb);
extern int dev_valid_name(const char *name);
-extern int dev_ioctl(unsigned int cmd, void __user *);
-extern int dev_ethtool(struct ifreq *);
+extern int dev_ioctl(struct net *net, unsigned int cmd, void __user *);
+extern int dev_ethtool(struct net *net, struct ifreq *);
extern unsigned dev_get_flags(const struct net_device *);
extern int dev_change_flags(struct net_device *, unsigned);
extern int dev_change_name(struct net_device *, char *);

```

```

@@ -1320,7 +1324,7 @@ extern void dev_set_allmulti(struct net_device *dev, int inc);
extern void netdev_state_change(struct net_device *dev);
extern void netdev_features_change(struct net_device *dev);
/* Load a device via the kmod */
-extern void dev_load(const char *name);
+extern void dev_load(struct net *net, const char *name);
extern void dev_mcast_init(void);
extern int netdev_max_backlog;
extern int weight_p;
diff --git a/include/net/net_namespace.h b/include/net/net_namespace.h
index 5472476..fac42db 100644
--- a/include/net/net_namespace.h
+++ b/include/net/net_namespace.h
@@ -22,6 +22,10 @@ struct net {
    struct proc_dir_entry *proc_net;
    struct proc_dir_entry *proc_net_stat;
    struct proc_dir_entry *proc_net_root;
+
+ struct list_head dev_base_head;
+ struct hlist_head *dev_name_head;
+ struct hlist_head *dev_index_head;
};

extern struct net init_net;
diff --git a/include/net/pkt_cls.h b/include/net/pkt_cls.h
index 7968b1d..f285de6 100644
--- a/include/net/pkt_cls.h
+++ b/include/net/pkt_cls.h
@@ -2,6 +2,7 @@
#define __NET_PKT_CLS_H

#include <linux/pkt_cls.h>
+#include <net/net_namespace.h>
#include <net/sch_generic.h>
#include <net/act_api.h>

@@ -351,7 +352,7 @@ tcf_match_indev(struct sk_buff *skb, char *indev)
    if (indev[0]) {
        if (!skb->iif)
            return 0;
-    dev = __dev_get_by_index(skb->iif);
+    dev = __dev_get_by_index(&init_net, skb->iif);
        if (!dev || strcmp(indev, dev->name))
            return 0;
    }
diff --git a/include/net/rtnetlink.h b/include/net/rtnetlink.h
index 8218288..793863e 100644
--- a/include/net/rtnetlink.h

```

```

+++ b/include/net/rtnetlink.h
@@ -78,7 +78,7 @@ extern void __rtnl_link_unregister(struct rtnl_link_ops *ops);
extern int rtnl_link_register(struct rtnl_link_ops *ops);
extern void rtnl_link_unregister(struct rtnl_link_ops *ops);

-extern struct net_device *rtnl_create_link(char *ifname,
+extern struct net_device *rtnl_create_link(struct net *net, char *ifname,
      const struct rtnl_link_ops *ops, struct nlattr *tb[]);
extern const struct nla_policy ifla_policy[IFLA_MAX+1];

diff --git a/include/net/wext.h b/include/net/wext.h
index c02b8de..80b31d8 100644
--- a/include/net/wext.h
+++ b/include/net/wext.h
@@ -5,16 +5,23 @@
 * wireless extensions interface to the core code
 */

+struct net;
+
+#ifdef CONFIG_WIRELESS_EXT
-extern int wext_proc_init(void);
-extern int wext_handle_ioctl(struct ifreq *ifr, unsigned int cmd,
+extern int wext_proc_init(struct net *net);
+extern void wext_proc_exit(struct net *net);
+extern int wext_handle_ioctl(struct net *net, struct ifreq *ifr, unsigned int cmd,
      void __user *arg);
#else
-static inline int wext_proc_init(void)
+static inline int wext_proc_init(struct net *net)
{
    return 0;
}
-static inline int wext_handle_ioctl(struct ifreq *ifr, unsigned int cmd,
+static inline void wext_proc_exit(struct net *net)
+{
+    return;
+}
+static inline int wext_handle_ioctl(struct net *net, struct ifreq *ifr, unsigned int cmd,
      void __user *arg)
{
    return -EINVAL;
}
diff --git a/net/802/tr.c b/net/802/tr.c
index 032c31e..55c76d7 100644
--- a/net/802/tr.c
+++ b/net/802/tr.c
@@ -533,7 +533,7 @@ static int rif_seq_show(struct seq_file *seq, void *v)
    seq_puts(seq,

```



```

    "if    TR address    TTL    rcf    routing segments\n");
else {
- struct net_device *dev = dev_get_by_index(entry->iface);
+ struct net_device *dev = dev_get_by_index(&init_net, entry->iface);
    long ttl = (long) (entry->last_used + sysctl_tr_rif_timeout)
        - (long) jiffies;

diff --git a/net/8021q/vlan.c b/net/8021q/vlan.c
index 55f2dbb..546f6db 100644
--- a/net/8021q/vlan.c
+++ b/net/8021q/vlan.c
@@ -51,7 +51,7 @@ static char vlan_copyright[] = "Ben Greear <greearb@candelatech.com>";
static char vlan_buggyright[] = "David S. Miller <davem@redhat.com>";

static int vlan_device_event(struct notifier_block *, unsigned long, void *);
-static int vlan_ioctl_handler(void __user *);
+static int vlan_ioctl_handler(struct net *net, void __user *);
static int unregister_vlan_dev(struct net_device *, unsigned short );

static struct notifier_block vlan_notifier_block = {
@@ -699,7 +699,7 @@ out:
    * o execute requested action or pass command to the device driver
    * arg is really a struct vlan_ioctl_args __user *.
    */
-static int vlan_ioctl_handler(void __user *arg)
+static int vlan_ioctl_handler(struct net *net, void __user *arg)
{
    int err;
    unsigned short vid = 0;
@@ -728,7 +728,7 @@ static int vlan_ioctl_handler(void __user *arg)
    case GET_VLAN_REALDEV_NAME_CMD:
    case GET_VLAN_VID_CMD:
        err = -ENODEV;
-    dev = __dev_get_by_name(args.device1);
+    dev = __dev_get_by_name(&init_net, args.device1);
    if (!dev)
        goto out;

diff --git a/net/8021q/vlan_netlink.c b/net/8021q/vlan_netlink.c
index 6cdd1e0..0996185 100644
--- a/net/8021q/vlan_netlink.c
+++ b/net/8021q/vlan_netlink.c
@@ -11,6 +11,7 @@
#include <linux/kernel.h>
#include <linux/netdevice.h>
#include <linux/if_vlan.h>
+#include <net/net_namespace.h>
#include <net/netlink.h>

```

```

#include <net/rtnetlink.h>
#include "vlan.h"
@@ -112,7 +113,7 @@ static int vlan_newlink(struct net_device *dev,

    if (!tb[IFLA_LINK])
        return -EINVAL;
- real_dev = __dev_get_by_index(nla_get_u32(tb[IFLA_LINK]));
+ real_dev = __dev_get_by_index(&init_net, nla_get_u32(tb[IFLA_LINK]));
    if (!real_dev)
        return -ENODEV;

diff --git a/net/8021q/vlanproc.c b/net/8021q/vlanproc.c
index ac80e6b..6cefd8 100644
--- a/net/8021q/vlanproc.c
+++ b/net/8021q/vlanproc.c
@@ -254,7 +254,7 @@ static void *vlan_seq_start(struct seq_file *seq, loff_t *pos)
    if (*pos == 0)
        return SEQ_START_TOKEN;

- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (!is_vlan_dev(dev))
        continue;

@@ -273,9 +273,9 @@ static void *vlan_seq_next(struct seq_file *seq, void *v, loff_t *pos)

    dev = (struct net_device *)v;
    if (v == SEQ_START_TOKEN)
- dev = net_device_entry(&dev_base_head);
+ dev = net_device_entry(&init_net.dev_base_head);

- for_each_netdev_continue(dev) {
+ for_each_netdev_continue(&init_net, dev) {
    if (!is_vlan_dev(dev))
        continue;

diff --git a/net/appletalk/ddp.c b/net/appletalk/ddp.c
index 36fcdbf..7c0b515 100644
--- a/net/appletalk/ddp.c
+++ b/net/appletalk/ddp.c
@@ -677,7 +677,7 @@ static int atif_ioctl(int cmd, void __user *arg)
    if (copy_from_user(&atreq, arg, sizeof(atreq)))
        return -EFAULT;

- dev = __dev_get_by_name(atreq.ifr_name);
+ dev = __dev_get_by_name(&init_net, atreq.ifr_name);
    if (!dev)
        return -ENODEV;

```

```

@@ -901,7 +901,7 @@ static int atrtr_ioctl(unsigned int cmd, void __user *arg)
    if (copy_from_user(name, rt.rt_dev, IFNAMSIZ-1))
        return -EFAULT;
    name[IFNAMSIZ-1] = '\0';
-   dev = __dev_get_by_name(name);
+   dev = __dev_get_by_name(&init_net, name);
    if (!dev)
        return -ENODEV;
}
@@ -1273,7 +1273,7 @@ static __inline__ int is_ip_over_ddp(struct sk_buff *skb)

static int handle_ip_over_ddp(struct sk_buff *skb)
{
- struct net_device *dev = __dev_get_by_name("ipddp0");
+ struct net_device *dev = __dev_get_by_name(&init_net, "ipddp0");
  struct net_device_stats *stats;

  /* This needs to be able to handle ipddp"N" devices */
diff --git a/net/atm/mpc.c b/net/atm/mpc.c
index 0968430..2086396 100644
--- a/net/atm/mpc.c
+++ b/net/atm/mpc.c
@@ -244,7 +244,7 @@ static struct net_device *find_lec_by_itfnum(int itf)
    char name[IFNAMSIZ];

    sprintf(name, "lec%d", itf);
-   dev = dev_get_by_name(name);
+   dev = dev_get_by_name(&init_net, name);

    return dev;
}
diff --git a/net/ax25/af_ax25.c b/net/ax25/af_ax25.c
index 8d13a8b..993e5c7 100644
--- a/net/ax25/af_ax25.c
+++ b/net/ax25/af_ax25.c
@@ -631,7 +631,7 @@ static int ax25_setsockopt(struct socket *sock, int level, int optname,
    break;
}

-   dev = dev_get_by_name(devname);
+   dev = dev_get_by_name(&init_net, devname);
    if (dev == NULL) {
        res = -ENODEV;
        break;
diff --git a/net/bridge/br_if.c b/net/bridge/br_if.c
index 749f0e8..f4847ab 100644
--- a/net/bridge/br_if.c

```

```

+++ b/net/bridge/br_if.c
@@ -303,7 +303,7 @@ int br_del_bridge(const char *name)
    int ret = 0;

    rtnl_lock();
- dev = __dev_get_by_name(name);
+ dev = __dev_get_by_name(&init_net, name);
    if (dev == NULL)
        ret = -ENXIO; /* Could not find device */

@@ -444,7 +444,7 @@ void __exit br_cleanup_bridges(void)
    struct net_device *dev, *nxt;

    rtnl_lock();
- for_each_netdev_safe(dev, nxt)
+ for_each_netdev_safe(&init_net, dev, nxt)
    if (dev->priv_flags & IFF_EBRIDGE)
        del_br(dev->priv);
    rtnl_unlock();
diff --git a/net/bridge/br_ioctl.c b/net/bridge/br_ioctl.c
index bb15e9e..0655a5f 100644
--- a/net/bridge/br_ioctl.c
+++ b/net/bridge/br_ioctl.c
@@ -18,6 +18,7 @@
#include <linux/if_bridge.h>
#include <linux/netdevice.h>
#include <linux/times.h>
+#include <net/net_namespace.h>
#include <asm/uaccess.h>
#include "br_private.h"

@@ -27,7 +28,7 @@ static int get_bridge_ifindices(int *indices, int num)
    struct net_device *dev;
    int i = 0;

- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (i >= num)
        break;
    if (dev->priv_flags & IFF_EBRIDGE)
@@ -90,7 +91,7 @@ static int add_del_if(struct net_bridge *br, int ifindex, int isadd)
    if (!capable(CAP_NET_ADMIN))
        return -EPERM;

- dev = dev_get_by_index(ifindex);
+ dev = dev_get_by_index(&init_net, ifindex);
    if (dev == NULL)
        return -EINVAL;

```

```

@@ -364,7 +365,7 @@ static int old_deviceless(void __user *uarg)
    return -EOPNOTSUPP;
}

-int br_ioctl_deviceless_stub(unsigned int cmd, void __user *uarg)
+int br_ioctl_deviceless_stub(struct net *net, unsigned int cmd, void __user *uarg)
{
    switch (cmd) {
    case SIOCGIFBR:
diff --git a/net/bridge/br_netlink.c b/net/bridge/br_netlink.c
index 0fcf6f0..53ab8e0 100644
--- a/net/bridge/br_netlink.c
+++ b/net/bridge/br_netlink.c
@@ -12,6 +12,7 @@

#include <linux/kernel.h>
#include <net/rtnetlink.h>
+#include <net/net_namespace.h>
#include "br_private.h"

static inline size_t br_nlmsg_size(void)
@@ -110,7 +111,7 @@ static int br_dump_ifinfo(struct sk_buff *skb, struct netlink_callback *cb)
    int idx;

    idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    /* not a bridge port */
    if (dev->br_port == NULL || idx < cb->args[0])
        goto skip;
@@ -155,7 +156,7 @@ static int br_rtm_setlink(struct sk_buff *skb, struct nlmsghdr *nlh, void
*arg)
    if (new_state > BR_STATE_BLOCKING)
        return -EINVAL;

- dev = __dev_get_by_index(ifm->ifi_index);
+ dev = __dev_get_by_index(&init_net, ifm->ifi_index);
    if (!dev)
        return -ENODEV;

diff --git a/net/bridge/br_private.h b/net/bridge/br_private.h
index 21bf3a9..5ca6bd7 100644
--- a/net/bridge/br_private.h
+++ b/net/bridge/br_private.h
@@ -196,7 +196,7 @@ extern struct sk_buff *br_handle_frame(struct net_bridge_port *p,

/* br_ioctl.c */

```

```

extern int br_dev_ioctl(struct net_device *dev, struct ifreq *rq, int cmd);
-extern int br_ioctl_deviceless_stub(unsigned int cmd, void __user *arg);
+extern int br_ioctl_deviceless_stub(struct net *net, unsigned int cmd, void __user *arg);

/* br_netfilter.c */
#ifdef CONFIG_BRIDGE_NETFILTER
diff --git a/net/core/dev.c b/net/core/dev.c
index 316043b..c51cf40 100644
--- a/net/core/dev.c
+++ b/net/core/dev.c
@@ -190,25 +190,22 @@ static struct net_dma net_dma = {
 * unregister_netdevice(), which must be called with the rtnl
 * semaphore held.
 */
-__LIST_HEAD(dev_base_head);
-DEFINE_RWLOCK(dev_base_lock);

-EXPORT_SYMBOL(dev_base_head);
EXPORT_SYMBOL(dev_base_lock);

#define NETDEV_HASHBITS 8
-static struct hlist_head dev_name_head[1<<NETDEV_HASHBITS];
-static struct hlist_head dev_index_head[1<<NETDEV_HASHBITS];
+#define NETDEV_HASHENTRIES (1 << NETDEV_HASHBITS)

-static inline struct hlist_head *dev_name_hash(const char *name)
+static inline struct hlist_head *dev_name_hash(struct net *net, const char *name)
{
    unsigned hash = full_name_hash(name, strlen(name), IFNAMSIZ);
-    return &dev_name_head[hash & ((1<<NETDEV_HASHBITS)-1)];
+    return &net->dev_name_head[hash & ((1 << NETDEV_HASHBITS) - 1)];
}

-static inline struct hlist_head *dev_index_hash(int ifindex)
+static inline struct hlist_head *dev_index_hash(struct net *net, int ifindex)
{
-    return &dev_index_head[ifindex & ((1<<NETDEV_HASHBITS)-1)];
+    return &net->dev_index_head[ifindex & ((1 << NETDEV_HASHBITS) - 1)];
}

/*
@@ -492,7 +489,7 @@ unsigned long netdev_boot_base(const char *prefix, int unit)
 * If device already registered then return base of 1
 * to indicate not to probe for this interface
 */
- if (__dev_get_by_name(name))
+ if (__dev_get_by_name(&init_net, name))
    return 1;

```

```

    for (i = 0; i < NETDEV_BOOT_SETUP_MAX; i++)
@@ -547,11 +544,11 @@ __setup("netdev=", netdev_boot_setup);
    * careful with locks.
    */

-struct net_device *__dev_get_by_name(const char *name)
+struct net_device *__dev_get_by_name(struct net *net, const char *name)
{
    struct hlist_node *p;

- hlist_for_each(p, dev_name_hash(name)) {
+ hlist_for_each(p, dev_name_hash(net, name)) {
    struct net_device *dev
        = hlist_entry(p, struct net_device, name_hlist);
    if (!strcmp(dev->name, name, IFNAMSIZ))
@@ -571,12 +568,12 @@ struct net_device *__dev_get_by_name(const char *name)
    * matching device is found.
    */

-struct net_device *dev_get_by_name(const char *name)
+struct net_device *dev_get_by_name(struct net *net, const char *name)
{
    struct net_device *dev;

    read_lock(&dev_base_lock);
- dev = __dev_get_by_name(name);
+ dev = __dev_get_by_name(net, name);
    if (dev)
        dev_hold(dev);
    read_unlock(&dev_base_lock);
@@ -594,11 +591,11 @@ struct net_device *dev_get_by_name(const char *name)
    * or @dev_base_lock.
    */

-struct net_device *__dev_get_by_index(int ifindex)
+struct net_device *__dev_get_by_index(struct net *net, int ifindex)
{
    struct hlist_node *p;

- hlist_for_each(p, dev_index_hash(ifindex)) {
+ hlist_for_each(p, dev_index_hash(net, ifindex)) {
    struct net_device *dev
        = hlist_entry(p, struct net_device, index_hlist);
    if (dev->ifindex == ifindex)
@@ -618,12 +615,12 @@ struct net_device *__dev_get_by_index(int ifindex)
    * dev_put to indicate they have finished with it.
    */

```

```

-struct net_device *dev_get_by_index(int ifindex)
+struct net_device *dev_get_by_index(struct net *net, int ifindex)
{
    struct net_device *dev;

    read_lock(&dev_base_lock);
- dev = __dev_get_by_index(ifindex);
+ dev = __dev_get_by_index(net, ifindex);
    if (dev)
        dev_hold(dev);
    read_unlock(&dev_base_lock);
@@ -644,13 +641,13 @@ struct net_device *dev_get_by_index(int ifindex)
    * If the API was consistent this would be __dev_get_by_hwaddr
    */

-struct net_device *dev_getbyhwaddr(unsigned short type, char *ha)
+struct net_device *dev_getbyhwaddr(struct net *net, unsigned short type, char *ha)
{
    struct net_device *dev;

    ASSERT_RTNL();

- for_each_netdev(dev)
+ for_each_netdev(&init_net, dev)
    if (dev->type == type &&
        !memcmp(dev->dev_addr, ha, dev->addr_len))
        return dev;
@@ -660,12 +657,12 @@ struct net_device *dev_getbyhwaddr(unsigned short type, char *ha)

EXPORT_SYMBOL(dev_getbyhwaddr);

-struct net_device *__dev_getfirstbyhwtype(unsigned short type)
+struct net_device *__dev_getfirstbyhwtype(struct net *net, unsigned short type)
{
    struct net_device *dev;

    ASSERT_RTNL();
- for_each_netdev(dev)
+ for_each_netdev(net, dev)
    if (dev->type == type)
        return dev;

@@ -674,12 +671,12 @@ struct net_device *__dev_getfirstbyhwtype(unsigned short type)

EXPORT_SYMBOL(__dev_getfirstbyhwtype);

-struct net_device *dev_getfirstbyhwtype(unsigned short type)

```



```

+struct net_device *dev_getfirstbyhwtype(struct net *net, unsigned short type)
{
    struct net_device *dev;

    rtnl_lock();
- dev = __dev_getfirstbyhwtype(type);
+ dev = __dev_getfirstbyhwtype(net, type);
    if (dev)
        dev_hold(dev);
    rtnl_unlock();
@@ -699,13 +696,13 @@ EXPORT_SYMBOL(dev_getfirstbyhwtype);
    * dev_put to indicate they have finished with it.
    */

-struct net_device * dev_get_by_flags(unsigned short if_flags, unsigned short mask)
+struct net_device * dev_get_by_flags(struct net *net, unsigned short if_flags, unsigned short
mask)
{
    struct net_device *dev, *ret;

    ret = NULL;
    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(net, dev) {
    if (((dev->flags ^ if_flags) & mask) == 0) {
        dev_hold(dev);
        ret = dev;
@@ -763,6 +760,10 @@ int dev_alloc_name(struct net_device *dev, const char *name)
    const int max_netdevices = 8*PAGE_SIZE;
    long *inuse;
    struct net_device *d;
+ struct net *net;
+
+ BUG_ON(!dev->nd_net);
+ net = dev->nd_net;

    p = strchr(name, IFNAMSIZ-1, '%');
    if (p) {
@@ -779,7 +780,7 @@ int dev_alloc_name(struct net_device *dev, const char *name)
    if (!inuse)
        return -ENOMEM;

- for_each_netdev(d) {
+ for_each_netdev(net, d) {
    if (!sscanf(d->name, name, &i))
        continue;
    if (i < 0 || i >= max_netdevices)
@@ -796,7 +797,7 @@ int dev_alloc_name(struct net_device *dev, const char *name)

```

```

}

snprintf(buf, sizeof(buf), name, i);
- if (!__dev_get_by_name(buf)) {
+ if (!__dev_get_by_name(net, buf)) {
    strcpy(dev->name, buf, IFNAMSIZ);
    return i;
}
@@ -822,9 +823,12 @@ int dev_change_name(struct net_device *dev, char *newname)
    char oldname[IFNAMSIZ];
    int err = 0;
    int ret;
+ struct net *net;

    ASSERT_RTNL();
+ BUG_ON(!dev->nd_net);

+ net = dev->nd_net;
    if (dev->flags & IFF_UP)
        return -EBUSY;

@@ -839,7 +843,7 @@ int dev_change_name(struct net_device *dev, char *newname)
    return err;
    strcpy(newname, dev->name);
}
- else if (__dev_get_by_name(newname))
+ else if (__dev_get_by_name(net, newname))
    return -EEXIST;
else
    strcpy(dev->name, newname, IFNAMSIZ);
@@ -849,7 +853,7 @@ rollback:

    write_lock_bh(&dev_base_lock);
    hlist_del(&dev->name_hlist);
- hlist_add_head(&dev->name_hlist, dev_name_hash(dev->name));
+ hlist_add_head(&dev->name_hlist, dev_name_hash(net, dev->name));
    write_unlock_bh(&dev_base_lock);

    ret = raw_notifier_call_chain(&netdev_chain, NETDEV_CHANGENAME, dev);
@@ -908,12 +912,12 @@ void netdev_state_change(struct net_device *dev)
    * available in this kernel then it becomes a nop.
    */

-void dev_load(const char *name)
+void dev_load(struct net *net, const char *name)
{
    struct net_device *dev;

```

```

    read_lock(&dev_base_lock);
- dev = __dev_get_by_name(name);
+ dev = __dev_get_by_name(net, name);
    read_unlock(&dev_base_lock);

    if (!dev && capable(CAP_SYS_MODULE))
@@ -1052,6 +1056,8 @@ int dev_close(struct net_device *dev)
}

+static int dev_boot_phase = 1;
+
+/*
+ * Device change register/unregister. These are not inline or static
+ * as we export them to the world.
@@ -1075,23 +1081,27 @@ int register_netdevice_notifier(struct notifier_block *nb)
{
    struct net_device *dev;
    struct net_device *last;
+ struct net *net;
    int err;

    rtnl_lock();
    err = raw_notifier_chain_register(&netdev_chain, nb);
    if (err)
        goto unlock;
+ if (dev_boot_phase)
+     goto unlock;
+ for_each_net(net) {
+     for_each_netdev(net, dev) {
+         err = nb->notifier_call(nb, NETDEV_REGISTER, dev);
+         err = notifier_to_errno(err);
+         if (err)
+             goto rollback;
+
+         if (!(dev->flags & IFF_UP))
+             continue;

- for_each_netdev(dev) {
-     err = nb->notifier_call(nb, NETDEV_REGISTER, dev);
-     err = notifier_to_errno(err);
-     if (err)
-         goto rollback;
-
-     if (!(dev->flags & IFF_UP))
-         continue;
-
-     nb->notifier_call(nb, NETDEV_UP, dev);

```

```

+ nb->notifier_call(nb, NETDEV_UP, dev);
+ }
}

unlock:
@@ -1100,15 +1110,17 @@ unlock:

rollback:
    last = dev;
- for_each_netdev(dev) {
-     if (dev == last)
-         break;
+ for_each_net(net) {
+     for_each_netdev(net, dev) {
+         if (dev == last)
+             break;

-     if (dev->flags & IFF_UP) {
-         nb->notifier_call(nb, NETDEV_GOING_DOWN, dev);
-         nb->notifier_call(nb, NETDEV_DOWN, dev);
+     if (dev->flags & IFF_UP) {
+         nb->notifier_call(nb, NETDEV_GOING_DOWN, dev);
+         nb->notifier_call(nb, NETDEV_DOWN, dev);
+     }
+     nb->notifier_call(nb, NETDEV_UNREGISTER, dev);
    }
-     nb->notifier_call(nb, NETDEV_UNREGISTER, dev);
    }
    goto unlock;
}
@@ -2169,7 +2181,7 @@ int register_gifconf(unsigned int family, gifconf_func_t * gifconf)
    * match. --pb
    */

-static int dev_ifname(struct ifreq __user *arg)
+static int dev_ifname(struct net *net, struct ifreq __user *arg)
{
    struct net_device *dev;
    struct ifreq ifr;
@@ -2182,7 +2194,7 @@ static int dev_ifname(struct ifreq __user *arg)
    return -EFAULT;

    read_lock(&dev_base_lock);
- dev = __dev_get_by_index(ifr.ifr_ifindex);
+ dev = __dev_get_by_index(net, ifr.ifr_ifindex);
    if (!dev) {
        read_unlock(&dev_base_lock);
        return -ENODEV;

```

```

@@ -2202,7 +2214,7 @@ static int dev_ifname(struct ifreq __user *arg)
 * Thus we will need a 'compatibility mode'.
 */

-static int dev_ifconf(char __user *arg)
+static int dev_ifconf(struct net *net, char __user *arg)
{
    struct ifconf ifc;
    struct net_device *dev;
@@ -2226,7 +2238,7 @@ static int dev_ifconf(char __user *arg)
 */

    total = 0;
- for_each_netdev(dev) {
+ for_each_netdev(net, dev) {
    for (i = 0; i < NPROTO; i++) {
        if (gifconf_list[i]) {
            int done;
@@ -2260,6 +2272,7 @@ static int dev_ifconf(char __user *arg)
 */
void *dev_seq_start(struct seq_file *seq, loff_t *pos)
{
+ struct net *net = seq->private;
    loff_t off;
    struct net_device *dev;

@@ -2268,7 +2281,7 @@ void *dev_seq_start(struct seq_file *seq, loff_t *pos)
    return SEQ_START_TOKEN;

    off = 1;
- for_each_netdev(dev)
+ for_each_netdev(net, dev)
    if (off++ == *pos)
        return dev;

@@ -2277,9 +2290,10 @@ void *dev_seq_start(struct seq_file *seq, loff_t *pos)

void *dev_seq_next(struct seq_file *seq, void *v, loff_t *pos)
{
+ struct net *net = seq->private;
    ++*pos;
    return v == SEQ_START_TOKEN ?
- first_net_device() : next_net_device((struct net_device *)v);
+ first_net_device(net) : next_net_device((struct net_device *)v);
}

void dev_seq_stop(struct seq_file *seq, void *v)
@@ -2375,7 +2389,22 @@ static const struct seq_operations dev_seq_ops = {

```

```

static int dev_seq_open(struct inode *inode, struct file *file)
{
- return seq_open(file, &dev_seq_ops);
+ struct seq_file *seq;
+ int res;
+ res = seq_open(file, &dev_seq_ops);
+ if (!res) {
+ seq = file->private_data;
+ seq->private = get_net(PROC_NET(inode));
+ }
+ return res;
+}
+
+static int dev_seq_release(struct inode *inode, struct file *file)
+{
+ struct seq_file *seq = file->private_data;
+ struct net *net = seq->private;
+ put_net(net);
+ return seq_release(inode, file);
+}

static const struct file_operations dev_seq_fops = {
@@ -2383,7 +2412,7 @@ static const struct file_operations dev_seq_fops = {
    .open    = dev_seq_open,
    .read    = seq_read,
    .llseek  = seq_lseek,
- .release  = seq_release,
+ .release  = dev_seq_release,
};

static const struct seq_operations softnet_seq_ops = {
@@ -2535,30 +2564,49 @@ static const struct file_operations ptype_seq_fops = {
};

-static int __init dev_proc_init(void)
+static int dev_proc_net_init(struct net *net)
{
    int rc = -ENOMEM;

- if (!proc_net_fops_create(&init_net, "dev", S_IRUGO, &dev_seq_fops))
+ if (!proc_net_fops_create(net, "dev", S_IRUGO, &dev_seq_fops))
    goto out;
- if (!proc_net_fops_create(&init_net, "softnet_stat", S_IRUGO, &softnet_seq_fops))
+ if (!proc_net_fops_create(net, "softnet_stat", S_IRUGO, &softnet_seq_fops))
    goto out_dev;
- if (!proc_net_fops_create(&init_net, "ptype", S_IRUGO, &ptype_seq_fops))

```

```

+ if (!proc_net_fops_create(net, "ptype", S_IRUGO, &ptype_seq_fops))
    goto out_softnet;

- if (wext_proc_init())
+ if (wext_proc_init(net))
    goto out_ptype;
    rc = 0;
out:
    return rc;
out_ptype:
- proc_net_remove(&init_net, "ptype");
+ proc_net_remove(net, "ptype");
out_softnet:
- proc_net_remove(&init_net, "softnet_stat");
+ proc_net_remove(net, "softnet_stat");
out_dev:
- proc_net_remove(&init_net, "dev");
+ proc_net_remove(net, "dev");
    goto out;
}
+
+static void dev_proc_net_exit(struct net *net)
+{
+ wext_proc_exit(net);
+
+ proc_net_remove(net, "ptype");
+ proc_net_remove(net, "softnet_stat");
+ proc_net_remove(net, "dev");
+}
+
+static struct pernet_operations dev_proc_ops = {
+ .init = dev_proc_net_init,
+ .exit = dev_proc_net_exit,
+};
+
+static int __init dev_proc_init(void)
+{
+ return register_pernet_subsys(&dev_proc_ops);
+}
+
+ #else
+ #define dev_proc_init() 0
+ #endif /* CONFIG_PROC_FS */
@@ -2993,10 +3041,10 @@ int dev_set_mac_address(struct net_device *dev, struct sockaddr
*sa)
/*
 * Perform the SIOCxIFxxx calls.
 */
-static int dev_ifsioc(struct ifreq *ifr, unsigned int cmd)

```

```

+static int dev_ifsioc(struct net *net, struct ifreq *ifr, unsigned int cmd)
{
    int err;
- struct net_device *dev = __dev_get_by_name(ifr->ifr_name);
+ struct net_device *dev = __dev_get_by_name(net, ifr->ifr_name);

    if (!dev)
        return -ENODEV;
@@ -3149,7 +3197,7 @@ static int dev_ifsioc(struct ifreq *ifr, unsigned int cmd)
    * positive or a negative errno code on error.
    */

-int dev_ioctl(unsigned int cmd, void __user *arg)
+int dev_ioctl(struct net *net, unsigned int cmd, void __user *arg)
{
    struct ifreq ifr;
    int ret;
@@ -3162,12 +3210,12 @@ int dev_ioctl(unsigned int cmd, void __user *arg)

    if (cmd == SIOCGIFCONF) {
        rtnl_lock();
- ret = dev_ifconf((char __user *) arg);
+ ret = dev_ifconf(net, (char __user *) arg);
        rtnl_unlock();
        return ret;
    }
    if (cmd == SIOCGIFNAME)
- return dev_ifname((struct ifreq __user *)arg);
+ return dev_ifname(net, (struct ifreq __user *)arg);

    if (copy_from_user(&ifr, arg, sizeof(struct ifreq)))
        return -EFAULT;
@@ -3197,9 +3245,9 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
    case SIOCGIFMAP:
    case SIOCGIFINDEX:
    case SIOCGIFTXQLEN:
- dev_load(ifr.ifr_name);
+ dev_load(net, ifr.ifr_name);
    read_lock(&dev_base_lock);
- ret = dev_ifsioc(&ifr, cmd);
+ ret = dev_ifsioc(net, &ifr, cmd);
    read_unlock(&dev_base_lock);
    if (!ret) {
        if (colon)
@@ -3211,9 +3259,9 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
        return ret;

    case SIOCETHTOOL:

```



```

- dev_load(ifr.ifr_name);
+ dev_load(net, ifr.ifr_name);
  rtnl_lock();
- ret = dev_ethtool(&ifr);
+ ret = dev_ethtool(net, &ifr);
  rtnl_unlock();
  if (!ret) {
    if (colon)
@@ -3235,9 +3283,9 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
  case SIOCSIFNAME:
    if (!capable(CAP_NET_ADMIN))
      return -EPERM;
- dev_load(ifr.ifr_name);
+ dev_load(net, ifr.ifr_name);
  rtnl_lock();
- ret = dev_ifsioc(&ifr, cmd);
+ ret = dev_ifsioc(net, &ifr, cmd);
  rtnl_unlock();
  if (!ret) {
    if (colon)
@@ -3276,9 +3324,9 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
  /* fall through */
  case SIOCBONDSLAVEINFOQUERY:
  case SIOCBONDINFOQUERY:
- dev_load(ifr.ifr_name);
+ dev_load(net, ifr.ifr_name);
  rtnl_lock();
- ret = dev_ifsioc(&ifr, cmd);
+ ret = dev_ifsioc(net, &ifr, cmd);
  rtnl_unlock();
  return ret;

@@ -3298,9 +3346,9 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
  if (cmd == SIOCWANDEV ||
      (cmd >= SIOCDEVPRIVATE &&
       cmd <= SIOCDEVPRIVATE + 15)) {
- dev_load(ifr.ifr_name);
+ dev_load(net, ifr.ifr_name);
  rtnl_lock();
- ret = dev_ifsioc(&ifr, cmd);
+ ret = dev_ifsioc(net, &ifr, cmd);
  rtnl_unlock();
  if (!ret && copy_to_user(arg, &ifr,
      sizeof(struct ifreq)))
@@ -3309,7 +3357,7 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
  }
  /* Take care of Wireless Extensions */
  if (cmd >= SIOCIWFIRST && cmd <= SIOCIWLAST)

```

```

- return wext_handle_ioctl(&ifr, cmd, arg);
+ return wext_handle_ioctl(net, &ifr, cmd, arg);
  return -EINVAL;
}
}
@@ -3322,19 +3370,17 @@ int dev_ioctl(unsigned int cmd, void __user *arg)
 * number. The caller must hold the rtnl semaphore or the
 * dev_base_lock to be sure it remains unique.
 */
-static int dev_new_index(void)
+static int dev_new_index(struct net *net)
{
  static int ifindex;
  for (;;) {
    if (++ifindex <= 0)
      ifindex = 1;
- if (!__dev_get_by_index(ifindex))
+ if (!__dev_get_by_index(net, ifindex))
    return ifindex;
  }
}

-static int dev_boot_phase = 1;
-
/* Delayed registration/unregistration */
static DEFINE_SPINLOCK(net_todo_list_lock);
static struct list_head net_todo_list = LIST_HEAD_INIT(net_todo_list);
@@ -3368,6 +3414,7 @@ int register_netdevice(struct net_device *dev)
  struct hlist_head *head;
  struct hlist_node *p;
  int ret;
+ struct net *net;

  BUG_ON(dev_boot_phase);
  ASSERT_RTNL();
@@ -3376,6 +3423,8 @@ int register_netdevice(struct net_device *dev)

  /* When net_device's are persistent, this will be fatal. */
  BUG_ON(dev->reg_state != NETREG_UNINITIALIZED);
+ BUG_ON(!dev->nd_net);
+ net = dev->nd_net;

  spin_lock_init(&dev->queue_lock);
  spin_lock_init(&dev->xmit_lock);
@@ -3400,12 +3449,12 @@ int register_netdevice(struct net_device *dev)
  goto err_uninit;
}

```

```

- dev->ifindex = dev_new_index();
+ dev->ifindex = dev_new_index(net);
  if (dev->iflink == -1)
    dev->iflink = dev->ifindex;

  /* Check for existence of name */
- head = dev_name_hash(dev->name);
+ head = dev_name_hash(net, dev->name);
  hlist_for_each(p, head) {
    struct net_device *d
      = hlist_entry(p, struct net_device, name_hlist);
@@ -3483,9 +3532,9 @@ int register_netdevice(struct net_device *dev)

  dev_init_scheduler(dev);
  write_lock_bh(&dev_base_lock);
- list_add_tail(&dev->dev_list, &dev_base_head);
+ list_add_tail(&dev->dev_list, &net->dev_base_head);
  hlist_add_head(&dev->name_hlist, head);
- hlist_add_head(&dev->index_hlist, dev_index_hash(dev->ifindex));
+ hlist_add_head(&dev->index_hlist, dev_index_hash(net, dev->ifindex));
  dev_hold(dev);
  write_unlock_bh(&dev_base_lock);

@@ -4049,6 +4098,45 @@ int netdev_compute_features(unsigned long all, unsigned long one)
}
EXPORT_SYMBOL(netdev_compute_features);

+/* Initialize per network namespace state */
+static int netdev_init(struct net *net)
+{
+  int i;
+  INIT_LIST_HEAD(&net->dev_base_head);
+  rwlock_init(&dev_base_lock);
+
+  net->dev_name_head = kmalloc(
+    sizeof(*net->dev_name_head)*NETDEV_HASHENTRIES, GFP_KERNEL);
+  if (!net->dev_name_head)
+    return -ENOMEM;
+
+  net->dev_index_head = kmalloc(
+    sizeof(*net->dev_index_head)*NETDEV_HASHENTRIES, GFP_KERNEL);
+  if (!net->dev_index_head) {
+    kfree(net->dev_name_head);
+    return -ENOMEM;
+  }
+
+  for (i = 0; i < NETDEV_HASHENTRIES; i++)
+    INIT_HLIST_HEAD(&net->dev_name_head[i]);

```

```

+
+ for (i = 0; i < NETDEV_HASHENTRIES; i++)
+ INIT_HLIST_HEAD(&net->dev_index_head[i]);
+
+ return 0;
+}
+
+static void netdev_exit(struct net *net)
+{
+ kfree(net->dev_name_head);
+ kfree(net->dev_index_head);
+}
+
+static struct pernet_operations netdev_net_ops = {
+ .init = netdev_init,
+ .exit = netdev_exit,
+};
+
+/*
+ * Initialize the DEV module. At boot time this walks the device list and
+ * unhooks any devices that fail to initialise (normally hardware not
@@ -4076,11 +4164,8 @@ static int __init net_dev_init(void)
for (i = 0; i < 16; i++)
INIT_LIST_HEAD(&ptype_base[i]);

- for (i = 0; i < ARRAY_SIZE(dev_name_head); i++)
- INIT_HLIST_HEAD(&dev_name_head[i]);
-
- for (i = 0; i < ARRAY_SIZE(dev_index_head); i++)
- INIT_HLIST_HEAD(&dev_index_head[i]);
+ if (register_pernet_subsys(&netdev_net_ops))
+ goto out;

/*
* Initialise the packet receive queues.
diff --git a/net/core/dev_mcast.c b/net/core/dev_mcast.c
index 8e069fc..1c4f619 100644
--- a/net/core/dev_mcast.c
+++ b/net/core/dev_mcast.c
@@ -187,11 +187,12 @@ EXPORT_SYMBOL(dev_mc_unsync);
#ifdef CONFIG_PROC_FS
static void *dev_mc_seq_start(struct seq_file *seq, loff_t *pos)
{
+ struct net *net = seq->private;
struct net_device *dev;
loff_t off = 0;

read_lock(&dev_base_lock);

```

```

- for_each_netdev(dev) {
+ for_each_netdev(net, dev) {
    if (off++ == *pos)
        return dev;
    }
@@ -240,7 +241,22 @@ static const struct seq_operations dev_mc_seq_ops = {

static int dev_mc_seq_open(struct inode *inode, struct file *file)
{
- return seq_open(file, &dev_mc_seq_ops);
+ struct seq_file *seq;
+ int res;
+ res = seq_open(file, &dev_mc_seq_ops);
+ if (!res) {
+     seq = file->private_data;
+     seq->private = get_net(PROC_NET(inode));
+ }
+ return res;
+}
+
+static int dev_mc_seq_release(struct inode *inode, struct file *file)
+{
+ struct seq_file *seq = file->private_data;
+ struct net *net = seq->private;
+ put_net(net);
+ return seq_release(inode, file);
+}

static const struct file_operations dev_mc_seq_fops = {
@@ -248,14 +264,31 @@ static const struct file_operations dev_mc_seq_fops = {
    .open    = dev_mc_seq_open,
    .read    = seq_read,
    .llseek  = seq_lseek,
- .release  = seq_release,
+ .release  = dev_mc_seq_release,
};

#endif

+static int dev_mc_net_init(struct net *net)
+{
+ if (!proc_net_fops_create(net, "dev_mcast", 0, &dev_mc_seq_fops))
+ return -ENOMEM;
+ return 0;
+}
+
+static void dev_mc_net_exit(struct net *net)
+{

```

```

+ proc_net_remove(net, "dev_mcast");
+}
+
+static struct pernet_operations dev_mc_net_ops = {
+ .init = dev_mc_net_init,
+ .exit = dev_mc_net_exit,
+};
+
+ void __init dev_mcast_init(void)
+ {
- proc_net_fops_create(&init_net, "dev_mcast", 0, &dev_mc_seq_fops);
+ register_pernet_subsys(&dev_mc_net_ops);
+ }

EXPORT_SYMBOL(dev_mc_add);
diff --git a/net/core/ethtool.c b/net/core/ethtool.c
index 7c43f03..0d0b13c 100644
--- a/net/core/ethtool.c
+++ b/net/core/ethtool.c
@@ -779,9 +779,9 @@ static int ethtool_set_value(struct net_device *dev, char __user
*useraddr,

/* The main entry point in this file. Called from net/core/dev.c */

-int dev_ethtool(struct ifreq *ifr)
+int dev_ethtool(struct net *net, struct ifreq *ifr)
{
- struct net_device *dev = __dev_get_by_name(ifr->ifr_name);
+ struct net_device *dev = __dev_get_by_name(net, ifr->ifr_name);
    void __user *useraddr = ifr->ifr_data;
    u32 ethcmd;
    int rc;
diff --git a/net/core/fib_rules.c b/net/core/fib_rules.c
index 9eabe1a..1ba71ba 100644
--- a/net/core/fib_rules.c
+++ b/net/core/fib_rules.c
@@ -12,6 +12,7 @@
#include <linux/kernel.h>
#include <linux/list.h>
#include <net/net_namespace.h>
+#include <net/sock.h>
#include <net/fib_rules.h>

static LIST_HEAD(rules_ops);
@@ -198,6 +199,7 @@ errout:

static int fib_nl_newrule(struct sk_buff *skb, struct nlmsg_hdr* nlh, void *arg)
{

```

```

+ struct net *net = skb->sk->sk_net;
  struct fib_rule_hdr *frh = nlmsg_data(nlh);
  struct fib_rules_ops *ops = NULL;
  struct fib_rule *rule, *r, *last = NULL;
@@ -235,7 +237,7 @@ static int fib_nl_newrule(struct sk_buff *skb, struct nlmsg_hdr* nlh, void
*arg)

  rule->ifindex = -1;
  nla_strncpy(rule->ifname, tb[FRA_IFNAME], IFNAMSIZ);
- dev = __dev_get_by_name(rule->ifname);
+ dev = __dev_get_by_name(net, rule->ifname);
  if (dev)
    rule->ifindex = dev->ifindex;
}
diff --git a/net/core/neighbour.c b/net/core/neighbour.c
index 5f25f4f..2c6577c 100644
--- a/net/core/neighbour.c
+++ b/net/core/neighbour.c
@@ -1441,6 +1441,7 @@ int neigh_table_clear(struct neigh_table *tbl)

static int neigh_delete(struct sk_buff *skb, struct nlmsg_hdr* nlh, void *arg)
{
+ struct net *net = skb->sk->sk_net;
  struct ndmsg *ndm;
  struct nlattr *dst_attr;
  struct neigh_table *tbl;
@@ -1456,7 +1457,7 @@ static int neigh_delete(struct sk_buff *skb, struct nlmsg_hdr* nlh, void
*arg)

  ndm = nlmsg_data(nlh);
  if (ndm->ndm_ifindex) {
- dev = dev_get_by_index(ndm->ndm_ifindex);
+ dev = dev_get_by_index(net, ndm->ndm_ifindex);
  if (dev == NULL) {
    err = -ENODEV;
    goto out;
@@ -1506,6 +1507,7 @@ out:

static int neigh_add(struct sk_buff *skb, struct nlmsg_hdr* nlh, void *arg)
{
+ struct net *net = skb->sk->sk_net;
  struct ndmsg *ndm;
  struct nlattr *tb[NDA_MAX+1];
  struct neigh_table *tbl;
@@ -1522,7 +1524,7 @@ static int neigh_add(struct sk_buff *skb, struct nlmsg_hdr* nlh, void
*arg)

  ndm = nlmsg_data(nlh);

```

```

    if (ndm->ndm_ifindex) {
-   dev = dev_get_by_index(ndm->ndm_ifindex);
+   dev = dev_get_by_index(net, ndm->ndm_ifindex);
    if (dev == NULL) {
        err = -ENODEV;
        goto out;
diff --git a/net/core/netpoll.c b/net/core/netpoll.c
index 0952f93..bb7523a 100644
--- a/net/core/netpoll.c
+++ b/net/core/netpoll.c
@@ -653,7 +653,7 @@ int netpoll_setup(struct netpoll *np)
    int err;

    if (np->dev_name)
-   ndev = dev_get_by_name(np->dev_name);
+   ndev = dev_get_by_name(&init_net, np->dev_name);
    if (!ndev) {
        printk(KERN_ERR "%s: %s doesn't exist, aborting.\n",
            np->name, np->dev_name);
diff --git a/net/core/pktgen.c b/net/core/pktgen.c
index 21ae955..ba5338a 100644
--- a/net/core/pktgen.c
+++ b/net/core/pktgen.c
@@ -1999,7 +1999,7 @@ static int pktgen_setup_dev(struct pktgen_dev *pkt_dev, const char
*ifname)
    pkt_dev->odev = NULL;
    }

-   odev = dev_get_by_name(ifname);
+   odev = dev_get_by_name(&init_net, ifname);
    if (!odev) {
        printk(KERN_ERR "pktgen: no such netdevice: \"%s\"\n", ifname);
        return -ENODEV;
diff --git a/net/core/rtnetlink.c b/net/core/rtnetlink.c
index 416768d..44f91bb 100644
--- a/net/core/rtnetlink.c
+++ b/net/core/rtnetlink.c
@@ -306,10 +306,13 @@ EXPORT_SYMBOL_GPL(rtnl_link_register);
void __rtnl_link_unregister(struct rtnl_link_ops *ops)
{
    struct net_device *dev, *n;
+   struct net *net;

-   for_each_netdev_safe(dev, n) {
-       if (dev->rtnl_link_ops == ops)
-           ops->dellink(dev);
+   for_each_net(net) {
+       for_each_netdev_safe(net, dev, n) {

```



```

+ if (dev->rtnl_link_ops == ops)
+ ops->dellink(dev);
+ }
}
list_del(&ops->list);
}
@@ -693,12 +696,13 @@ nla_put_failure:

```

```

static int rtnl_dump_ifinfo(struct sk_buff *skb, struct netlink_callback *cb)
{
+ struct net *net = skb->sk->sk_net;
  int idx;
  int s_idx = cb->args[0];
  struct net_device *dev;

  idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(net, dev) {
  if (idx < s_idx)
    goto cont;
  if (rtnl_fill_ifinfo(skb, dev, RTM_NEWLINK,
@@ -858,6 +862,7 @@ errout:

```

```

static int rtnl_setlink(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)
{
+ struct net *net = skb->sk->sk_net;
  struct ifinfomsg *ifm;
  struct net_device *dev;
  int err;
@@ -876,9 +881,9 @@ static int rtnl_setlink(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)
  err = -EINVAL;
  ifm = nlmsg_data(nlh);
  if (ifm->ifi_index > 0)
- dev = dev_get_by_index(ifm->ifi_index);
+ dev = dev_get_by_index(net, ifm->ifi_index);
  else if (tb[IFLA_IFNAME])
- dev = dev_get_by_name(ifname);
+ dev = dev_get_by_name(net, ifname);
  else
    goto errout;

```

```

@@ -904,6 +909,7 @@ errout:

```

```

static int rtnl_dellink(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)
{
+ struct net *net = skb->sk->sk_net;
  const struct rtnl_link_ops *ops;
  struct net_device *dev;

```

```

    struct ifinfomsg *ifm;
@@ -920,9 +926,9 @@ static int rtnl_dellink(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)

    ifm = nlmsg_data(nlh);
    if (ifm->ifi_index > 0)
-   dev = __dev_get_by_index(ifm->ifi_index);
+   dev = __dev_get_by_index(net, ifm->ifi_index);
    else if (tb[IFLA_IFNAME])
-   dev = __dev_get_by_name(ifname);
+   dev = __dev_get_by_name(net, ifname);
    else
        return -EINVAL;

@@ -937,7 +943,7 @@ static int rtnl_dellink(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)
    return 0;
}

-struct net_device *rtnl_create_link(char *ifname,
+struct net_device *rtnl_create_link(struct net *net, char *ifname,
    const struct rtnl_link_ops *ops, struct nlattr *tb[])
{
    int err;
@@ -954,6 +960,7 @@ struct net_device *rtnl_create_link(char *ifname,
    goto err_free;
}

+ dev->nd_net = net;
    dev->rtnl_link_ops = ops;

    if (tb[IFLA_MTU])
@@ -981,6 +988,7 @@ err:

static int rtnl_newlink(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)
{
+ struct net *net = skb->sk->sk_net;
    const struct rtnl_link_ops *ops;
    struct net_device *dev;
    struct ifinfomsg *ifm;
@@ -1004,9 +1012,9 @@ replay:

    ifm = nlmsg_data(nlh);
    if (ifm->ifi_index > 0)
-   dev = __dev_get_by_index(ifm->ifi_index);
+   dev = __dev_get_by_index(net, ifm->ifi_index);
    else if (ifname[0])
-   dev = __dev_get_by_name(ifname);
+   dev = __dev_get_by_name(net, ifname);
    else

```

```

dev = NULL;

@@ -1092,7 +1100,7 @@ replay:
    if (!ifname[0])
        snprintf(ifname, IFNAMSIZ, "%s%%d", ops->kind);

- dev = rtnl_create_link(ifname, ops, tb);
+ dev = rtnl_create_link(net, ifname, ops, tb);

    if (IS_ERR(dev))
        err = PTR_ERR(dev);
@@ -1109,6 +1117,7 @@ replay:

static int rtnl_getlink(struct sk_buff *skb, struct nlmsg_hdr* nlh, void *arg)
{
+ struct net *net = skb->sk->sk_net;
    struct ifinfomsg *ifm;
    struct nlattr *tb[IFLA_MAX+1];
    struct net_device *dev = NULL;
@@ -1121,7 +1130,7 @@ static int rtnl_getlink(struct sk_buff *skb, struct nlmsg_hdr* nlh, void
*arg)

    ifm = nlmsg_data(nlh);
    if (ifm->ifi_index > 0) {
- dev = dev_get_by_index(ifm->ifi_index);
+ dev = dev_get_by_index(net, ifm->ifi_index);
        if (dev == NULL)
            return -ENODEV;
    } else
diff --git a/net/core/sock.c b/net/core/sock.c
index 9feec0f..ff52c83 100644
--- a/net/core/sock.c
+++ b/net/core/sock.c
@@ -372,6 +372,7 @@ int sock_setsockopt(struct socket *sock, int level, int optname,
    char __user *optval, int optlen)
{
    struct sock *sk=sock->sk;
+ struct net *net = sk->sk_net;
    struct sk_filter *filter;
    int val;
    int valbool;
@@ -613,7 +614,7 @@ set_rcvbuf:
    if (devname[0] == '\0') {
        sk->sk_bound_dev_if = 0;
    } else {
- struct net_device *dev = dev_get_by_name(devname);
+ struct net_device *dev = dev_get_by_name(net, devname);
        if (!dev) {

```

```

    ret = -ENODEV;
    break;
diff --git a/net/decnet/af_decnet.c b/net/decnet/af_decnet.c
index 83398da..aabe98d 100644
--- a/net/decnet/af_decnet.c
+++ b/net/decnet/af_decnet.c
@@ -751,7 +751,7 @@ static int dn_bind(struct socket *sock, struct sockaddr *uaddr, int
addr_len)
    if (dn_ntohs(saddr->sdn_nodeaddr)) {
        read_lock(&dev_base_lock);
        ldev = NULL;
-   for_each_netdev(dev) {
+   for_each_netdev(&init_net, dev) {
        if (!dev->dn_ptr)
            continue;
        if (dn_dev_islocal(dev, dn_saddr2dn(saddr))) {
diff --git a/net/decnet/dn_dev.c b/net/decnet/dn_dev.c
index 04f12e2..db25a97 100644
--- a/net/decnet/dn_dev.c
+++ b/net/decnet/dn_dev.c
@@ -513,7 +513,7 @@ int dn_dev_ioctl(unsigned int cmd, void __user *arg)
    ifr->ifr_name[IFNAMSIZ-1] = 0;

#ifdef CONFIG_KMOD
-   dev_load(ifr->ifr_name);
+   dev_load(&init_net, ifr->ifr_name);
#endif

    switch(cmd) {
@@ -531,7 +531,7 @@ int dn_dev_ioctl(unsigned int cmd, void __user *arg)

    rtnl_lock();

-   if ((dev = __dev_get_by_name(ifr->ifr_name)) == NULL) {
+   if ((dev = __dev_get_by_name(&init_net, ifr->ifr_name)) == NULL) {
        ret = -ENODEV;
        goto done;
    }
@@ -629,7 +629,7 @@ static struct dn_dev *dn_dev_by_index(int ifindex)
{
    struct net_device *dev;
    struct dn_dev *dn_dev = NULL;
-   dev = dev_get_by_index(ifindex);
+   dev = dev_get_by_index(&init_net, ifindex);
    if (dev) {
        dn_dev = dev->dn_ptr;
        dev_put(dev);
@@ -694,7 +694,7 @@ static int dn_nl_newaddr(struct sk_buff *skb, struct nlmsghdr *nlh, void

```

```

*arg)
    return -EINVAL;

    ifm = nlmsg_data(nlh);
- if ((dev = __dev_get_by_index(ifm->ifa_index)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, ifm->ifa_index)) == NULL)
    return -ENODEV;

    if ((dn_db = dev->dn_ptr) == NULL) {
@@ -800,7 +800,7 @@ static int dn_nl_dump_ifaddr(struct sk_buff *skb, struct netlink_callback
*cb)
    skip_naddr = cb->args[1];

    idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (idx < skip_ndevs)
        goto cont;
    else if (idx > skip_ndevs) {
@@ -1297,7 +1297,7 @@ void dn_dev_devices_off(void)
    struct net_device *dev;

    rtnl_lock();
- for_each_netdev(dev)
+ for_each_netdev(&init_net, dev)
    dn_dev_down(dev);
    rtnl_unlock();

@@ -1308,7 +1308,7 @@ void dn_dev_devices_on(void)
    struct net_device *dev;

    rtnl_lock();
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (dev->flags & IFF_UP)
        dn_dev_up(dev);
    }
@@ -1342,7 +1342,7 @@ static void *dn_dev_seq_start(struct seq_file *seq, loff_t *pos)
    return SEQ_START_TOKEN;

    i = 1;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (!is_dn_dev(dev))
        continue;

@@ -1361,9 +1361,9 @@ static void *dn_dev_seq_next(struct seq_file *seq, void *v, loff_t *pos)

```

```

dev = (struct net_device *)v;
if (v == SEQ_START_TOKEN)
- dev = net_device_entry(&dev_base_head);
+ dev = net_device_entry(&init_net.dev_base_head);

- for_each_netdev_continue(dev) {
+ for_each_netdev_continue(&init_net, dev) {
    if (!is_dn_dev(dev))
        continue;

diff --git a/net/decnet/dn_fib.c b/net/decnet/dn_fib.c
index d2bc19d..3760a20 100644
--- a/net/decnet/dn_fib.c
+++ b/net/decnet/dn_fib.c
@@ -212,7 +212,7 @@ static int dn_fib_check_nh(const struct rtmsg *r, struct dn_fib_info *fi,
struct
    return -EINVAL;
    if (dnet_addr_type(nh->nh_gw) != RTN_UNICAST)
        return -EINVAL;
- if ((dev = __dev_get_by_index(nh->nh_oif)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, nh->nh_oif)) == NULL)
    return -ENODEV;
    if (!(dev->flags&IFF_UP))
        return -ENETDOWN;
@@ -255,7 +255,7 @@ out:
    if (nh->nh_flags&(RTNH_F_PERVASIVE|RTNH_F_ONLINK))
        return -EINVAL;

- dev = __dev_get_by_index(nh->nh_oif);
+ dev = __dev_get_by_index(&init_net, nh->nh_oif);
    if (dev == NULL || dev->dn_ptr == NULL)
        return -ENODEV;
    if (!(dev->flags&IFF_UP))
@@ -355,7 +355,7 @@ struct dn_fib_info *dn_fib_create_info(const struct rtmsg *r, struct
dn_kern_rta
    if (nhs != 1 || nh->nh_gw)
        goto err_inval;
    nh->nh_scope = RT_SCOPE_NOWHERE;
- nh->nh_dev = dev_get_by_index(fi->fib_nh->nh_oif);
+ nh->nh_dev = dev_get_by_index(&init_net, fi->fib_nh->nh_oif);
    err = -ENODEV;
    if (nh->nh_dev == NULL)
        goto failure;
@@ -602,7 +602,7 @@ static void dn_fib_del_ifaddr(struct dn_ifaddr *ifa)

/* Scan device list */
read_lock(&dev_base_lock);
- for_each_netdev(dev) {

```

```

+ for_each_netdev(&init_net, dev) {
    dn_db = dev->dn_ptr;
    if (dn_db == NULL)
        continue;
diff --git a/net/decnet/dn_route.c b/net/decnet/dn_route.c
index 580e786..70b1c3f 100644
--- a/net/decnet/dn_route.c
+++ b/net/decnet/dn_route.c
@@ -908,7 +908,7 @@ static int dn_route_output_slow(struct dst_entry **pprt, const struct flowi
*old

    /* If we have an output interface, verify its a DECnet device */
    if (oldflp->oif) {
- dev_out = dev_get_by_index(oldflp->oif);
+ dev_out = dev_get_by_index(&init_net, oldflp->oif);
    err = -ENODEV;
    if (dev_out && dev_out->dn_ptr == NULL) {
        dev_put(dev_out);
@@ -929,7 +929,7 @@ static int dn_route_output_slow(struct dst_entry **pprt, const struct flowi
*old
        goto out;
    }
    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (!dev->dn_ptr)
        continue;
    if (!dn_dev_islocal(dev, oldflp->fld_src))
@@ -1556,7 +1556,7 @@ static int dn_cache_getroute(struct sk_buff *in_skb, struct nlmsg_hdr
*nlh, void

    if (fl.iif) {
        struct net_device *dev;
- if ((dev = dev_get_by_index(fl.iif)) == NULL) {
+ if ((dev = dev_get_by_index(&init_net, fl.iif)) == NULL) {
            kfree_skb(skb);
            return -ENODEV;
        }
diff --git a/net/decnet/sysctl_net_decnet.c b/net/decnet/sysctl_net_decnet.c
index 52e40d7..ae354a4 100644
--- a/net/decnet/sysctl_net_decnet.c
+++ b/net/decnet/sysctl_net_decnet.c
@@ -259,7 +259,7 @@ static int dn_def_dev_strategy(ctl_table *table, int __user *name, int
nlen,

    devname[newlen] = 0;

- dev = dev_get_by_name(devname);

```

```

+ dev = dev_get_by_name(&init_net, devname);
  if (dev == NULL)
    return -ENODEV;

@@ -299,7 +299,7 @@ static int dn_def_dev_handler(ctl_table *table, int write,
  devname[*lenp] = 0;
  strip_it(devname);

- dev = dev_get_by_name(devname);
+ dev = dev_get_by_name(&init_net, devname);
  if (dev == NULL)
    return -ENODEV;

diff --git a/net/econet/af_econet.c b/net/econet/af_econet.c
index f877f3b..9938e76 100644
--- a/net/econet/af_econet.c
+++ b/net/econet/af_econet.c
@@ -662,7 +662,7 @@ static int ec_dev_ioctl(struct socket *sock, unsigned int cmd, void __user
*arg)
  if (copy_from_user(&ifr, arg, sizeof(struct ifreq)))
    return -EFAULT;

- if ((dev = dev_get_by_name(ifr.ifr_name)) == NULL)
+ if ((dev = dev_get_by_name(&init_net, ifr.ifr_name)) == NULL)
    return -ENODEV;

  sec = (struct sockaddr_ec *)&ifr.ifr_addr;
diff --git a/net/ipv4/arp.c b/net/ipv4/arp.c
index a11e7a5..3a68300 100644
--- a/net/ipv4/arp.c
+++ b/net/ipv4/arp.c
@@ -981,7 +981,7 @@ static int arp_req_set(struct arpreq *r, struct net_device * dev)
  if (mask && mask != htonl(0xFFFFFFFF))
    return -EINVAL;
  if (!dev && (r->arp_flags & ATF_COM)) {
- dev = dev_getbyhwaddr(r->arp_ha.sa_family, r->arp_ha.sa_data);
+ dev = dev_getbyhwaddr(&init_net, r->arp_ha.sa_family, r->arp_ha.sa_data);
  if (!dev)
    return -ENODEV;
  }
@@ -1169,7 +1169,7 @@ int arp_ioctl(unsigned int cmd, void __user *arg)
  rtnl_lock();
  if (r.arp_dev[0]) {
    err = -ENODEV;
- if ((dev = __dev_get_by_name(r.arp_dev)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, r.arp_dev)) == NULL)
    goto out;

```



```

/* Mmmm... It is wrong... ARPHRD_NETROM==0 */
diff --git a/net/ipv4/devinet.c b/net/ipv4/devinet.c
index 9996b57..3db853b 100644
--- a/net/ipv4/devinet.c
+++ b/net/ipv4/devinet.c
@@ -420,7 +420,7 @@ struct in_device *inetdev_by_index(int ifindex)
    struct net_device *dev;
    struct in_device *in_dev = NULL;
    read_lock(&dev_base_lock);
- dev = __dev_get_by_index(ifindex);
+ dev = __dev_get_by_index(&init_net, ifindex);
    if (dev)
        in_dev = in_dev_get(dev);
    read_unlock(&dev_base_lock);
@@ -506,7 +506,7 @@ static struct in_ifaddr *rtm_to_ifaddr(struct nlmsg_hdr *nlh)
    goto errout;
}

- dev = __dev_get_by_index(ifm->ifa_index);
+ dev = __dev_get_by_index(&init_net, ifm->ifa_index);
    if (dev == NULL) {
        err = -ENODEV;
        goto errout;
@@ -628,7 +628,7 @@ int devinet_ioctl(unsigned int cmd, void __user *arg)
    *colon = 0;

#ifdef CONFIG_KMOD
- dev_load(ifr.ifr_name);
+ dev_load(&init_net, ifr.ifr_name);
#endif

    switch (cmd) {
@@ -669,7 +669,7 @@ int devinet_ioctl(unsigned int cmd, void __user *arg)
    rtnl_lock();

    ret = -ENODEV;
- if ((dev = __dev_get_by_name(ifr.ifr_name)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, ifr.ifr_name)) == NULL)
        goto done;

    if (colon)
@@ -909,7 +909,7 @@ no_in_dev:
    */
    read_lock(&dev_base_lock);
    rcu_read_lock();
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if ((in_dev = __in_dev_get_rcu(dev)) == NULL)

```

```

continue;

@@ -988,7 +988,7 @@ __be32 inet_confirm_addr(const struct net_device *dev, __be32 dst,
__be32 local,

    read_lock(&dev_base_lock);
    rcu_read_lock();
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if ((in_dev = __in_dev_get_rcu(dev))) {
        addr = confirm_addr_indev(in_dev, dst, local, scope);
        if (addr)
@@ -1185,7 +1185,7 @@ static int inet_dump_ifaddr(struct sk_buff *skb, struct netlink_callback
*cb)

    s_ip_idx = ip_idx = cb->args[1];
    idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (idx < s_idx)
        goto cont;
    if (idx > s_idx)
@@ -1244,7 +1244,7 @@ static void devinet_copy_dflt_conf(int i)
    struct net_device *dev;

    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    struct in_device *in_dev;
    rcu_read_lock();
    in_dev = __in_dev_get_rcu(dev);
@@ -1333,7 +1333,7 @@ void inet_forward_change(void)
    IPV4_DEVCONF_DFLT(FORWARDING) = on;

    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    struct in_device *in_dev;
    rcu_read_lock();
    in_dev = __in_dev_get_rcu(dev);
diff --git a/net/ipv4/fib_frontend.c b/net/ipv4/fib_frontend.c
index 140bf7a..df17aab 100644
--- a/net/ipv4/fib_frontend.c
+++ b/net/ipv4/fib_frontend.c
@@ -334,7 +334,7 @@ static int rtenry_to_fib_config(int cmd, struct rtenry *rt,
    colon = strchr(devname, ':');
    if (colon)
        *colon = 0;

```

```

- dev = __dev_get_by_name(devname);
+ dev = __dev_get_by_name(&init_net, devname);
  if (!dev)
    return -ENODEV;
  cfg->fc_oif = dev->ifindex;
diff --git a/net/ipv4/fib_semantics.c b/net/ipv4/fib_semantics.c
index c434119..d30fb68 100644
--- a/net/ipv4/fib_semantics.c
+++ b/net/ipv4/fib_semantics.c
@@ -533,7 +533,7 @@ static int fib_check_nh(struct fib_config *cfg, struct fib_info *fi,
    return -EINVAL;
    if (inet_addr_type(nh->nh_gw) != RTN_UNICAST)
    return -EINVAL;
- if ((dev = __dev_get_by_index(nh->nh_oif)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, nh->nh_oif)) == NULL)
    return -ENODEV;
    if (!(dev->flags&IFF_UP))
    return -ENETDOWN;
@@ -799,7 +799,7 @@ struct fib_info *fib_create_info(struct fib_config *cfg)
    if (nhs != 1 || nh->nh_gw)
    goto err_inval;
    nh->nh_scope = RT_SCOPE_NOWHERE;
- nh->nh_dev = dev_get_by_index(fi->fib_nh->nh_oif);
+ nh->nh_dev = dev_get_by_index(&init_net, fi->fib_nh->nh_oif);
    err = -ENODEV;
    if (nh->nh_dev == NULL)
    goto failure;
diff --git a/net/ipv4/icmp.c b/net/ipv4/icmp.c
index 02a899b..68a2267 100644
--- a/net/ipv4/icmp.c
+++ b/net/ipv4/icmp.c
@@ -517,7 +517,7 @@ void icmp_send(struct sk_buff *skb_in, int type, int code, __be32 info)
    struct net_device *dev = NULL;

    if (rt->fl.iif && sysctl_icmp_errors_use_inbound_ifaddr)
- dev = dev_get_by_index(rt->fl.iif);
+ dev = dev_get_by_index(&init_net, rt->fl.iif);

    if (dev) {
        saddr = inet_select_addr(dev, 0, RT_SCOPE_LINK);
diff --git a/net/ipv4/igmp.c b/net/ipv4/igmp.c
index d78599a..ad500a4 100644
--- a/net/ipv4/igmp.c
+++ b/net/ipv4/igmp.c
@@ -2292,7 +2292,7 @@ static inline struct ip_mc_list *igmp_mc_get_first(struct seq_file *seq)
    struct igmp_mc_iter_state *state = igmp_mc_seq_private(seq);

    state->in_dev = NULL;

```

```

- for_each_netdev(state->dev) {
+ for_each_netdev(&init_net, state->dev) {
    struct in_device *in_dev;
    in_dev = in_dev_get(state->dev);
    if (!in_dev)
@@ -2454,7 +2454,7 @@ static inline struct ip_sf_list *igmp_mcf_get_first(struct seq_file *seq)

    state->idev = NULL;
    state->im = NULL;
- for_each_netdev(state->dev) {
+ for_each_netdev(&init_net, state->dev) {
    struct in_device *idev;
    idev = in_dev_get(state->dev);
    if (unlikely(idev == NULL))
diff --git a/net/ipv4/ip_fragment.c b/net/ipv4/ip_fragment.c
index 0231bdc..fabb86d 100644
--- a/net/ipv4/ip_fragment.c
+++ b/net/ipv4/ip_fragment.c
@@ -292,7 +292,7 @@ static void ip_expire(unsigned long arg)
    if ((qp->last_in&FIRST_IN) && qp->fragments != NULL) {
        struct sk_buff *head = qp->fragments;
        /* Send an ICMP "Fragment Reassembly Timeout" message. */
- if ((head->dev = dev_get_by_index(qp->iif)) != NULL) {
+ if ((head->dev = dev_get_by_index(&init_net, qp->iif)) != NULL) {
        icmp_send(head, ICMP_TIME_EXCEEDED, ICMP_EXC_FRAGTIME, 0);
        dev_put(head->dev);
    }
diff --git a/net/ipv4/ip_gre.c b/net/ipv4/ip_gre.c
index 5c14ed6..3106225 100644
--- a/net/ipv4/ip_gre.c
+++ b/net/ipv4/ip_gre.c
@@ -262,7 +262,7 @@ static struct ip_tunnel *ipgre_tunnel_locate(struct ip_tunnel_parm
*parms, int
    int i;
    for (i=1; i<100; i++) {
        sprintf(name, "gre%d", i);
- if (__dev_get_by_name(name) == NULL)
+ if (__dev_get_by_name(&init_net, name) == NULL)
        break;
    }
    if (i==100)
@@ -1196,7 +1196,7 @@ static int ipgre_tunnel_init(struct net_device *dev)
}

    if (!tdev && tunnel->parms.link)
- tdev = __dev_get_by_index(tunnel->parms.link);
+ tdev = __dev_get_by_index(&init_net, tunnel->parms.link);

```

```

if (tdev) {
    hlen = tdev->hard_header_len;
diff --git a/net/ipv4/ip_sockglue.c b/net/ipv4/ip_sockglue.c
index 6b420ae..b2b3053 100644
--- a/net/ipv4/ip_sockglue.c
+++ b/net/ipv4/ip_sockglue.c
@@ -602,7 +602,7 @@ static int do_ip_setsockopt(struct sock *sk, int level,
    dev_put(dev);
}
} else
- dev = __dev_get_by_index(mreq.imr_ifindex);
+ dev = __dev_get_by_index(&init_net, mreq.imr_ifindex);

err = -EADDRNOTAVAIL;
diff --git a/net/ipv4/ipconfig.c b/net/ipv4/ipconfig.c
index 08ff623..4303851 100644
--- a/net/ipv4/ipconfig.c
+++ b/net/ipv4/ipconfig.c
@@ -193,7 +193,7 @@ static int __init ic_open_devs(void)
    if (dev_change_flags(&loopback_dev, loopback_dev.flags | IFF_UP) < 0)
        printk(KERN_ERR "IP-Config: Failed to open %s\n", loopback_dev.name);

- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (dev == &loopback_dev)
        continue;
    if (user_dev_name[0] ? !strcmp(dev->name, user_dev_name) :
diff --git a/net/ipv4/ipip.c b/net/ipv4/ipip.c
index 3964372..652bd86 100644
--- a/net/ipv4/ipip.c
+++ b/net/ipv4/ipip.c
@@ -225,7 +225,7 @@ static struct ip_tunnel * ipip_tunnel_locate(struct ip_tunnel_parm *parms,
int c
    int i;
    for (i=1; i<100; i++) {
        sprintf(name, "tunl%d", i);
- if (__dev_get_by_name(name) == NULL)
+ if (__dev_get_by_name(&init_net, name) == NULL)
        break;
    }
    if (i==100)
@@ -822,7 +822,7 @@ static int ipip_tunnel_init(struct net_device *dev)
}

if (!tdev && tunnel->parms.link)
- tdev = __dev_get_by_index(tunnel->parms.link);
+ tdev = __dev_get_by_index(&init_net, tunnel->parms.link);

```

```

    if (tdev) {
        dev->hard_header_len = tdev->hard_header_len + sizeof(struct iphdr);
diff --git a/net/ipv4/ipmr.c b/net/ipv4/ipmr.c
index 0365988..b8b4b49 100644
--- a/net/ipv4/ipmr.c
+++ b/net/ipv4/ipmr.c
@@ -125,7 +125,7 @@ struct net_device *ipmr_new_tunnel(struct vifctl *v)
{
    struct net_device *dev;

- dev = __dev_get_by_name("tunl0");
+ dev = __dev_get_by_name(&init_net, "tunl0");

    if (dev) {
        int err;
@@ -149,7 +149,7 @@ struct net_device *ipmr_new_tunnel(struct vifctl *v)

    dev = NULL;

- if (err == 0 && (dev = __dev_get_by_name(p.name)) != NULL) {
+ if (err == 0 && (dev = __dev_get_by_name(&init_net, p.name)) != NULL) {
    dev->flags |= IFF_MULTICAST;

    in_dev = __in_dev_get_rtnl(dev);
diff --git a/net/ipv4/ipvs/ip_vs_sync.c b/net/ipv4/ipvs/ip_vs_sync.c
index 356f067..1960747 100644
--- a/net/ipv4/ipvs/ip_vs_sync.c
+++ b/net/ipv4/ipvs/ip_vs_sync.c
@@ -387,7 +387,7 @@ static int set_mcast_if(struct sock *sk, char *ifname)
    struct net_device *dev;
    struct inet_sock *inet = inet_sk(sk);

- if ((dev = __dev_get_by_name(ifname)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, ifname)) == NULL)
    return -ENODEV;

    if (sk->sk_bound_dev_if && dev->ifindex != sk->sk_bound_dev_if)
@@ -412,7 +412,7 @@ static int set_sync_mesg_maxlen(int sync_state)
    int num;

    if (sync_state == IP_VS_STATE_MASTER) {
- if ((dev = __dev_get_by_name(ip_vs_master_mcast_ifn)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, ip_vs_master_mcast_ifn)) == NULL)
    return -ENODEV;

    num = (dev->mtu - sizeof(struct iphdr) -
@@ -423,7 +423,7 @@ static int set_sync_mesg_maxlen(int sync_state)

```

```

IP_VS_DBG(7, "setting the maximum length of sync sending "
    "message %d.\n", sync_send_mesg_maxlen);
} else if (sync_state == IP_VS_STATE_BACKUP) {
- if ((dev = __dev_get_by_name(ip_vs_backup_mcast_ifn)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, ip_vs_backup_mcast_ifn)) == NULL)
    return -ENODEV;

    sync_rcv_mesg_maxlen = dev->mtu -
@@ -451,7 +451,7 @@ join_mcast_group(struct sock *sk, struct in_addr *addr, char *ifname)
    memset(&mreq, 0, sizeof(mreq));
    memcpy(&mreq.imr_multiaddr, addr, sizeof(struct in_addr));

- if ((dev = __dev_get_by_name(ifname)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, ifname)) == NULL)
    return -ENODEV;
    if (sk->sk_bound_dev_if && dev->ifindex != sk->sk_bound_dev_if)
        return -EINVAL;
@@ -472,7 +472,7 @@ static int bind_mcastif_addr(struct socket *sock, char *ifname)
    __be32 addr;
    struct sockaddr_in sin;

- if ((dev = __dev_get_by_name(ifname)) == NULL)
+ if ((dev = __dev_get_by_name(&init_net, ifname)) == NULL)
    return -ENODEV;

    addr = inet_select_addr(dev, 0, RT_SCOPE_UNIVERSE);
diff --git a/net/ipv4/netfilter/ipt_CLUSTERIP.c b/net/ipv4/netfilter/ipt_CLUSTERIP.c
index 50fc9e0..27f14e1 100644
--- a/net/ipv4/netfilter/ipt_CLUSTERIP.c
+++ b/net/ipv4/netfilter/ipt_CLUSTERIP.c
@@ -401,7 +401,7 @@ checkentry(const char *tablename,
    return false;
}

- dev = dev_get_by_name(e->ip.iniface);
+ dev = dev_get_by_name(&init_net, e->ip.iniface);
    if (!dev) {
        printk(KERN_WARNING "CLUSTERIP: no such interface %s\n", e->ip.iniface);
        return false;
diff --git a/net/ipv4/route.c b/net/ipv4/route.c
index efd2a92..396c631 100644
--- a/net/ipv4/route.c
+++ b/net/ipv4/route.c
@@ -2213,7 +2213,7 @@ static int ip_route_output_slow(struct rtable **rp, const struct flowi
*oldflp)

    if (oldflp->oif) {

```

```

- dev_out = dev_get_by_index(oldflp->oif);
+ dev_out = dev_get_by_index(&init_net, oldflp->oif);
  err = -ENODEV;
  if (dev_out == NULL)
    goto out;
@@ -2592,7 +2592,7 @@ static int inet_rtm_getroute(struct sk_buff *in_skb, struct nlmsghdr*
nlh, void
  if (iif) {
    struct net_device *dev;

- dev = __dev_get_by_index(iif);
+ dev = __dev_get_by_index(&init_net, iif);
    if (dev == NULL) {
      err = -ENODEV;
      goto errout_free;
diff --git a/net/ipv6/addrconf.c b/net/ipv6/addrconf.c
index 5b191a3..6de13c1 100644
--- a/net/ipv6/addrconf.c
+++ b/net/ipv6/addrconf.c
@@ -450,7 +450,7 @@ static void addrconf_forward_change(void)
  struct inet6_dev *idev;

  read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
  rcu_read_lock();
  idev = __in6_dev_get(dev);
  if (idev) {
@@ -912,7 +912,7 @@ int ipv6_dev_get_saddr(struct net_device *daddr_dev,
  read_lock(&dev_base_lock);
  rcu_read_lock();

- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
  struct inet6_dev *idev;
  struct inet6_ifaddr *ifa;

@@ -1858,7 +1858,7 @@ int addrconf_set_dstaddr(void __user *arg)
  if (copy_from_user(&ireq, arg, sizeof(struct in6_ifreq)))
    goto err_exit;

- dev = __dev_get_by_index(ireq.ifr6_ifindex);
+ dev = __dev_get_by_index(&init_net, ireq.ifr6_ifindex);

  err = -ENODEV;
  if (dev == NULL)
@@ -1889,7 +1889,7 @@ int addrconf_set_dstaddr(void __user *arg)

```



```

    if (err == 0) {
        err = -ENOBUFS;
-   if ((dev = __dev_get_by_name(p.name)) == NULL)
+   if ((dev = __dev_get_by_name(&init_net, p.name)) == NULL)
        goto err_exit;
        err = dev_open(dev);
    }
@@ -1919,7 +1919,7 @@ static int inet6_addr_add(int ifindex, struct in6_addr *pfx, int plen,
    if (!valid_lft || preferred_lft > valid_lft)
        return -EINVAL;

-   if ((dev = __dev_get_by_index(ifindex)) == NULL)
+   if ((dev = __dev_get_by_index(&init_net, ifindex)) == NULL)
        return -ENODEV;

    if ((idev = addrconf_add_dev(dev)) == NULL)
@@ -1970,7 +1970,7 @@ static int inet6_addr_del(int ifindex, struct in6_addr *pfx, int plen)
    struct inet6_dev *idev;
    struct net_device *dev;

-   if ((dev = __dev_get_by_index(ifindex)) == NULL)
+   if ((dev = __dev_get_by_index(&init_net, ifindex)) == NULL)
        return -ENODEV;

    if ((idev = __in6_dev_get(dev)) == NULL)
@@ -2065,7 +2065,7 @@ static void sit_add_v4_addrs(struct inet6_dev *idev)
    return;
}

-   for_each_netdev(dev) {
+   for_each_netdev(&init_net, dev) {
        struct in_device *in_dev = __in_dev_get_rtnl(dev);
        if (in_dev && (dev->flags & IFF_UP)) {
            struct in_ifaddr *ifa;
@@ -2221,12 +2221,12 @@ static void ip6_tnl_add_linklocal(struct inet6_dev *idev)

    /* first try to inherit the link-local address from the link device */
    if (idev->dev->iflink &&
-   (link_dev = __dev_get_by_index(idev->dev->iflink))) {
+   (link_dev = __dev_get_by_index(&init_net, idev->dev->iflink))) {
        if (!ipv6_inherit_linklocal(idev, link_dev))
            return;
    }
    /* then try to inherit it from any device */
-   for_each_netdev(link_dev) {
+   for_each_netdev(&init_net, link_dev) {
        if (!ipv6_inherit_linklocal(idev, link_dev))
            return;
    }

```

```

}
@@ -3084,7 +3084,7 @@ inet6_rtm_newaddr(struct sk_buff *skb, struct nlmsg_hdr *nlh, void
*arg)
    valid_lft = INFINITY_LIFE_TIME;
}

- dev = __dev_get_by_index(ifm->ifa_index);
+ dev = __dev_get_by_index(&init_net, ifm->ifa_index);
    if (dev == NULL)
        return -ENODEV;

@@ -3268,7 +3268,7 @@ static int inet6_dump_addr(struct sk_buff *skb, struct netlink_callback
*cb,
    s_ip_idx = ip_idx = cb->args[1];

    idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (idx < s_idx)
        goto cont;
    if (idx > s_idx)
@@ -3377,7 +3377,7 @@ static int inet6_rtm_getaddr(struct sk_buff *in_skb, struct nlmsg_hdr*
nlh,

    ifm = nlmsg_data(nlh);
    if (ifm->ifa_index)
- dev = __dev_get_by_index(ifm->ifa_index);
+ dev = __dev_get_by_index(&init_net, ifm->ifa_index);

    if ((ifa = ipv6_get_ifaddr(addr, dev, 1)) == NULL) {
        err = -EADDRNOTAVAIL;
@@ -3589,7 +3589,7 @@ static int inet6_dump_ifinfo(struct sk_buff *skb, struct netlink_callback
*cb)

    read_lock(&dev_base_lock);
    idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (idx < s_idx)
        goto cont;
    if ((idev = in6_dev_get(dev)) == NULL)
@@ -4266,7 +4266,7 @@ void __exit addrconf_cleanup(void)
    * clean dev list.
    */

- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (__in6_dev_get(dev) == NULL)

```

```

    continue;
    addrconf_ifdown(dev, 1);
diff --git a/net/ipv6/af_inet6.c b/net/ipv6/af_inet6.c
index 21931c8..e5c5aad 100644
--- a/net/ipv6/af_inet6.c
+++ b/net/ipv6/af_inet6.c
@@ -302,7 +302,7 @@ int inet6_bind(struct socket *sock, struct sockaddr *uaddr, int addr_len)
    err = -EINVAL;
    goto out;
}
- dev = dev_get_by_index(sk->sk_bound_dev_if);
+ dev = dev_get_by_index(&init_net, sk->sk_bound_dev_if);
if (!dev) {
    err = -ENODEV;
    goto out;
diff --git a/net/ipv6/anycast.c b/net/ipv6/anycast.c
index 0bd6654..d407992 100644
--- a/net/ipv6/anycast.c
+++ b/net/ipv6/anycast.c
@@ -112,10 +112,10 @@ int ipv6_sock_ac_join(struct sock *sk, int ifindex, struct in6_addr *addr)
} else {
    /* router, no matching interface: just pick one */

- dev = dev_get_by_flags(IFF_UP, IFF_UP|IFF_LOOPBACK);
+ dev = dev_get_by_flags(&init_net, IFF_UP, IFF_UP|IFF_LOOPBACK);
}
} else
- dev = dev_get_by_index(ifindex);
+ dev = dev_get_by_index(&init_net, ifindex);

if (dev == NULL) {
    err = -ENODEV;
@@ -196,7 +196,7 @@ int ipv6_sock_ac_drop(struct sock *sk, int ifindex, struct in6_addr *addr)

write_unlock_bh(&ipv6_sk_ac_lock);

- dev = dev_get_by_index(pac->acl_ifindex);
+ dev = dev_get_by_index(&init_net, pac->acl_ifindex);
if (dev) {
    ipv6_dev_ac_dec(dev, &pac->acl_addr);
    dev_put(dev);
@@ -224,7 +224,7 @@ void ipv6_sock_ac_close(struct sock *sk)
if (pac->acl_ifindex != prev_index) {
    if (dev)
        dev_put(dev);
- dev = dev_get_by_index(pac->acl_ifindex);
+ dev = dev_get_by_index(&init_net, pac->acl_ifindex);
    prev_index = pac->acl_ifindex;

```

```

    }
    if (dev)
@@ -429,7 +429,7 @@ int ipv6_chk_acast_addr(struct net_device *dev, struct in6_addr *addr)
    if (dev)
        return ipv6_chk_acast_dev(dev, addr);
    read_lock(&dev_base_lock);
- for_each_netdev(dev)
+ for_each_netdev(&init_net, dev)
    if (ipv6_chk_acast_dev(dev, addr)) {
        found = 1;
        break;
@@ -453,7 +453,7 @@ static inline struct ifacaddr6 *ac6_get_first(struct seq_file *seq)
    struct ac6_iter_state *state = ac6_seq_private(seq);

    state->idev = NULL;
- for_each_netdev(state->dev) {
+ for_each_netdev(&init_net, state->dev) {
    struct inet6_dev *idev;
    idev = in6_dev_get(state->dev);
    if (!idev)
diff --git a/net/ipv6/datagram.c b/net/ipv6/datagram.c
index fe0f490..2ed689a 100644
--- a/net/ipv6/datagram.c
+++ b/net/ipv6/datagram.c
@@ -544,7 +544,7 @@ int datagram_send_ctl(struct msghdr *msg, struct flowi *fl,
    if (!src_info->ipi6_ifindex)
        return -EINVAL;
    else {
- dev = dev_get_by_index(src_info->ipi6_ifindex);
+ dev = dev_get_by_index(&init_net, src_info->ipi6_ifindex);
    if (!dev)
        return -ENODEV;
    }
diff --git a/net/ipv6/ip6_tunnel.c b/net/ipv6/ip6_tunnel.c
index ca774d8..937625e 100644
--- a/net/ipv6/ip6_tunnel.c
+++ b/net/ipv6/ip6_tunnel.c
@@ -235,7 +235,7 @@ static struct ip6_tnl *ip6_tnl_create(struct ip6_tnl_parm *p)
    int i;
    for (i = 1; i < IP6_TNL_MAX; i++) {
        sprintf(name, "ip6tnl%d", i);
- if (__dev_get_by_name(name) == NULL)
+ if (__dev_get_by_name(&init_net, name) == NULL)
        break;
    }
    if (i == IP6_TNL_MAX)
@@ -650,7 +650,7 @@ static inline int ip6_tnl_rcv_ctl(struct ip6_tnl *t)
    struct net_device *ldev = NULL;

```

```

    if (p->link)
-   ldev = dev_get_by_index(p->link);
+   ldev = dev_get_by_index(&init_net, p->link);

    if ((ipv6_addr_is_multicast(&p->laddr) ||
        likely(ipv6_chk_addr(&p->laddr, ldev, 0))) &&
@@ -786,7 +786,7 @@ static inline int ip6_tnl_xmit_ctl(struct ip6_tnl *t)
    struct net_device *ldev = NULL;

    if (p->link)
-   ldev = dev_get_by_index(p->link);
+   ldev = dev_get_by_index(&init_net, p->link);

    if (unlikely(!ipv6_chk_addr(&p->laddr, ldev, 0)))
        printk(KERN_WARNING
diff --git a/net/ipv6/ipv6_sockglue.c b/net/ipv6/ipv6_sockglue.c
index 74254fc..eb330a4 100644
--- a/net/ipv6/ipv6_sockglue.c
+++ b/net/ipv6/ipv6_sockglue.c
@@ -542,7 +542,7 @@ done:
    if (sk->sk_bound_dev_if && sk->sk_bound_dev_if != val)
        goto e_inval;

-   if (__dev_get_by_index(val) == NULL) {
+   if (__dev_get_by_index(&init_net, val) == NULL) {
        retv = -ENODEV;
        break;
    }
diff --git a/net/ipv6/mcast.c b/net/ipv6/mcast.c
index a41d5a0..e2ab43c 100644
--- a/net/ipv6/mcast.c
+++ b/net/ipv6/mcast.c
@@ -215,7 +215,7 @@ int ipv6_sock_mc_join(struct sock *sk, int ifindex, struct in6_addr *addr)
    dst_release(&rt->u.dst);
}
} else
-   dev = dev_get_by_index(ifindex);
+   dev = dev_get_by_index(&init_net, ifindex);

    if (dev == NULL) {
        sock_kfree_s(sk, mc_lst, sizeof(*mc_lst));
@@ -266,7 +266,7 @@ int ipv6_sock_mc_drop(struct sock *sk, int ifindex, struct in6_addr *addr)
    *lnc = mc_lst->next;
    write_unlock_bh(&ipv6_sk_mc_lock);

-   if ((dev = dev_get_by_index(mc_lst->ifindex)) != NULL) {
+   if ((dev = dev_get_by_index(&init_net, mc_lst->ifindex)) != NULL) {

```

```

struct inet6_dev *idev = in6_dev_get(dev);

(void) ip6_mc_leave_src(sk, mc_lst, idev);
@@ -301,7 +301,7 @@ static struct inet6_dev *ip6_mc_find_dev(struct in6_addr *group, int
ifindex)
    dst_release(&rt->u.dst);
}
} else
- dev = dev_get_by_index(ifindex);
+ dev = dev_get_by_index(&init_net, ifindex);

if (!dev)
    return NULL;
@@ -332,7 +332,7 @@ void ipv6_sock_mc_close(struct sock *sk)
    np->ipv6_mc_list = mc_lst->next;
    write_unlock_bh(&ipv6_sk_mc_lock);

- dev = dev_get_by_index(mc_lst->ifindex);
+ dev = dev_get_by_index(&init_net, mc_lst->ifindex);
    if (dev) {
        struct inet6_dev *idev = in6_dev_get(dev);

@@ -2333,7 +2333,7 @@ static inline struct ifmcaddr6 *igmp6_mc_get_first(struct seq_file *seq)
    struct igmp6_mc_iter_state *state = igmp6_mc_seq_private(seq);

    state->idev = NULL;
- for_each_netdev(state->dev) {
+ for_each_netdev(&init_net, state->dev) {
    struct inet6_dev *idev;
    idev = in6_dev_get(state->dev);
    if (!idev)
@@ -2477,7 +2477,7 @@ static inline struct ip6_sf_list *igmp6_mcf_get_first(struct seq_file *seq)

    state->idev = NULL;
    state->im = NULL;
- for_each_netdev(state->dev) {
+ for_each_netdev(&init_net, state->dev) {
    struct inet6_dev *idev;
    idev = in6_dev_get(state->dev);
    if (unlikely(idev == NULL))
diff --git a/net/ipv6/raw.c b/net/ipv6/raw.c
index 20c1a76..71dcd48 100644
--- a/net/ipv6/raw.c
+++ b/net/ipv6/raw.c
@@ -283,7 +283,7 @@ static int rawv6_bind(struct sock *sk, struct sockaddr *uaddr, int
addr_len)
    if (!sk->sk_bound_dev_if)
        goto out;

```

```

- dev = dev_get_by_index(sk->sk_bound_dev_if);
+ dev = dev_get_by_index(&init_net, sk->sk_bound_dev_if);
  if (!dev) {
    err = -ENODEV;
    goto out;
diff --git a/net/ipv6/reassembly.c b/net/ipv6/reassembly.c
index de795c0..31601c9 100644
--- a/net/ipv6/reassembly.c
+++ b/net/ipv6/reassembly.c
@@ -301,7 +301,7 @@ static void ip6_frag_expire(unsigned long data)

    fq_kill(fq);

- dev = dev_get_by_index(fq->iif);
+ dev = dev_get_by_index(&init_net, fq->iif);
  if (!dev)
    goto out;

diff --git a/net/ipv6/route.c b/net/ipv6/route.c
index f4f0c34..5bdd9d4 100644
--- a/net/ipv6/route.c
+++ b/net/ipv6/route.c
@@ -1130,7 +1130,7 @@ int ip6_route_add(struct fib6_config *cfg)
#endif
  if (cfg->fc_ifindex) {
    err = -ENODEV;
- dev = dev_get_by_index(cfg->fc_ifindex);
+ dev = dev_get_by_index(&init_net, cfg->fc_ifindex);
    if (!dev)
      goto out;
    idev = in6_dev_get(dev);
@@ -2265,7 +2265,7 @@ static int inet6_rtm_getroute(struct sk_buff *in_skb, struct nlmsg_hdr*
nlh, void

  if (iif) {
    struct net_device *dev;
- dev = __dev_get_by_index(iif);
+ dev = __dev_get_by_index(&init_net, iif);
    if (!dev) {
      err = -ENODEV;
      goto errout;
diff --git a/net/ipv6/sit.c b/net/ipv6/sit.c
index eb20bb6..e79f419 100644
--- a/net/ipv6/sit.c
+++ b/net/ipv6/sit.c
@@ -167,7 +167,7 @@ static struct ip_tunnel * ipip6_tunnel_locate(struct ip_tunnel_parm
*parms, int

```

```

int i;
for (i=1; i<100; i++) {
    sprintf(name, "sit%d", i);
-   if (__dev_get_by_name(name) == NULL)
+   if (__dev_get_by_name(&init_net, name) == NULL)
        break;
}
if (i==100)
@@ -761,7 +761,7 @@ static int ipip6_tunnel_init(struct net_device *dev)
}

if (!tdev && tunnel->parms.link)
-   tdev = __dev_get_by_index(tunnel->parms.link);
+   tdev = __dev_get_by_index(&init_net, tunnel->parms.link);

if (tdev) {
    dev->hard_header_len = tdev->hard_header_len + sizeof(struct iphdr);
diff --git a/net/ipx/af_ipx.c b/net/ipx/af_ipx.c
index 24921f1..29b063d 100644
--- a/net/ipx/af_ipx.c
+++ b/net/ipx/af_ipx.c
@@ -989,7 +989,7 @@ static int ipxif_create(struct ipx_interface_definition *idef)
    if (intrfc)
        ipxif_put(intrfc);

-   dev = dev_get_by_name(idef->ipx_device);
+   dev = dev_get_by_name(&init_net, idef->ipx_device);
    rc = -ENODEV;
    if (!dev)
        goto out;
@@ -1097,7 +1097,7 @@ static int ipxif_delete(struct ipx_interface_definition *idef)
    if (!dlink_type)
        goto out;

-   dev = __dev_get_by_name(idef->ipx_device);
+   dev = __dev_get_by_name(&init_net, idef->ipx_device);
    rc = -ENODEV;
    if (!dev)
        goto out;
@@ -1192,7 +1192,7 @@ static int ipxif_ioctl(unsigned int cmd, void __user *arg)
    if (copy_from_user(&ifr, arg, sizeof(ifr)))
        break;
    sipx = (struct sockaddr_ipx *)&ifr.ifr_addr;
-   dev = __dev_get_by_name(ifr.ifr_name);
+   dev = __dev_get_by_name(&init_net, ifr.ifr_name);
    rc = -ENODEV;
    if (!dev)
        break;

```



```

diff --git a/net/irda/irnetlink.c b/net/irda/irnetlink.c
index 1e429c9..cd9ff17 100644
--- a/net/irda/irnetlink.c
+++ b/net/irda/irnetlink.c
@@ -15,6 +15,7 @@

#include <linux/socket.h>
#include <linux/irda.h>
+#include <net/net_namespace.h>
#include <net/sock.h>
#include <net/irda/irda.h>
#include <net/irda/irlap.h>
@@ -30,7 +31,7 @@ static struct genl_family irda_nl_family = {
    .maxattr = IRDA_NL_CMD_MAX,
};

-static struct net_device * ifname_to_netdev(struct genl_info *info)
+static struct net_device * ifname_to_netdev(struct net *net, struct genl_info *info)
{
    char * ifname;

@@ -41,7 +42,7 @@ static struct net_device * ifname_to_netdev(struct genl_info *info)

    IRDA_DEBUG(5, "%s(): Looking for %s\n", __FUNCTION__, ifname);

- return dev_get_by_name(ifname);
+ return dev_get_by_name(net, ifname);
}

static int irda_nl_set_mode(struct sk_buff *skb, struct genl_info *info)
@@ -57,7 +58,7 @@ static int irda_nl_set_mode(struct sk_buff *skb, struct genl_info *info)

    IRDA_DEBUG(5, "%s(): Switching to mode: %d\n", __FUNCTION__, mode);

- dev = ifname_to_netdev(info);
+ dev = ifname_to_netdev(&init_net, info);
    if (!dev)
        return -ENODEV;

@@ -82,7 +83,7 @@ static int irda_nl_get_mode(struct sk_buff *skb, struct genl_info *info)
    void *hdr;
    int ret = -ENOBUFS;

- dev = ifname_to_netdev(info);
+ dev = ifname_to_netdev(&init_net, info);
    if (!dev)
        return -ENODEV;

```

```

diff --git a/net/llc/af_llc.c b/net/llc/af_llc.c
index b482441..49eacba 100644
--- a/net/llc/af_llc.c
+++ b/net/llc/af_llc.c
@@ -252,7 +252,7 @@ static int llc_ui_autobind(struct socket *sock, struct sockaddr_llc *addr)
    if (!sock_flag(sk, SOCK_ZAPPED))
        goto out;
    rc = -ENODEV;
- llc->dev = dev_getfirstbyhwtype(addr->sllc_arphrd);
+ llc->dev = dev_getfirstbyhwtype(&init_net, addr->sllc_arphrd);
    if (!llc->dev)
        goto out;
    rc = -EUSERS;
@@ -303,7 +303,7 @@ static int llc_ui_bind(struct socket *sock, struct sockaddr *uaddr, int
addrlen)
    goto out;
    rc = -ENODEV;
    rtnl_lock();
- llc->dev = dev_getbyhwaddr(addr->sllc_arphrd, addr->sllc_mac);
+ llc->dev = dev_getbyhwaddr(&init_net, addr->sllc_arphrd, addr->sllc_mac);
    rtnl_unlock();
    if (!llc->dev)
        goto out;
diff --git a/net/llc/llc_core.c b/net/llc/llc_core.c
index d4b13a0..248b590 100644
--- a/net/llc/llc_core.c
+++ b/net/llc/llc_core.c
@@ -19,6 +19,6 @@
#include <linux/slab.h>
#include <linux/string.h>
#include <linux/init.h>
+#include <net/net_namespace.h>
#include <net/llc.h>

LIST_HEAD(llc_sap_list);
@@ -162,7 +163,7 @@ static int __init llc_init(void)
{
    struct net_device *dev;

- dev = first_net_device();
+ dev = first_net_device(&init_net);
    if (dev != NULL)
        dev = next_net_device(dev);
diff --git a/net/mac80211/ieee80211.c b/net/mac80211/ieee80211.c
index 56786ff..5ea86f5 100644
--- a/net/mac80211/ieee80211.c
+++ b/net/mac80211/ieee80211.c

```

```

@@ -21,6 +21,7 @@
#include <linux/wireless.h>
#include <linux/rtnetlink.h>
#include <linux/bitmap.h>
+#include <net/net_namespace.h>
#include <net/cfg80211.h>

#include "ieee80211_common.h"
diff --git a/net/mac80211/ieee80211_cfg.c b/net/mac80211/ieee80211_cfg.c
index 509096e..b1c13bc 100644
--- a/net/mac80211/ieee80211_cfg.c
+++ b/net/mac80211/ieee80211_cfg.c
@@ -8,6 +8,7 @@

#include <linux/nl80211.h>
#include <linux/rtnetlink.h>
+#include <net/net_namespace.h>
#include <net/cfg80211.h>
#include "ieee80211_i.h"
#include "ieee80211_cfg.h"
@@ -50,7 +51,7 @@ static int ieee80211_del_iface(struct wiphy *wiphy, int ifindex)
    if (unlikely(local->reg_state != IEEE80211_DEV_REGISTERED))
        return -ENODEV;

- dev = dev_get_by_index(ifindex);
+ dev = dev_get_by_index(&init_net, ifindex);
    if (!dev)
        return 0;

diff --git a/net/mac80211/tx.c b/net/mac80211/tx.c
index b65ff65..9e952e3 100644
--- a/net/mac80211/tx.c
+++ b/net/mac80211/tx.c
@@ -17,6 +17,7 @@
#include <linux/skbuff.h>
#include <linux/etherdevice.h>
#include <linux/bitmap.h>
+#include <net/net_namespace.h>
#include <net/ieee80211_radiotap.h>
#include <net/cfg80211.h>
#include <net/mac80211.h>
@@ -1018,7 +1019,7 @@ static int inline ieee80211_tx_prepare(struct ieee80211_txrx_data *tx,
    struct net_device *dev;

    pkt_data = (struct ieee80211_tx_packet_data *)skb->cb;
- dev = dev_get_by_index(pkt_data->ifindex);
+ dev = dev_get_by_index(&init_net, pkt_data->ifindex);
    if (unlikely(dev && !is_ieee80211_device(dev, mdev))) {

```

```

dev_put(dev);
dev = NULL;
@@ -1226,7 +1227,7 @@ int ieee80211_master_start_xmit(struct sk_buff *skb,
    memset(&control, 0, sizeof(struct ieee80211_tx_control));

    if (pkt_data->ifindex)
-   odev = dev_get_by_index(pkt_data->ifindex);
+   odev = dev_get_by_index(&init_net, pkt_data->ifindex);
    if (unlikely(odev && !is_ieee80211_device(odev, dev))) {
        dev_put(odev);
        odev = NULL;
@@ -1722,7 +1723,7 @@ struct sk_buff *ieee80211_beacon_get(struct ieee80211_hw *hw, int
if_id,
    u8 *b_head, *b_tail;
    int bh_len, bt_len;

-   bdev = dev_get_by_index(if_id);
+   bdev = dev_get_by_index(&init_net, if_id);
    if (bdev) {
        sdata = IEEE80211_DEV_TO_SUB_IF(bdev);
        ap = &sdata->u.ap;
@@ -1836,7 +1837,7 @@ ieee80211_get_buffered_bc(struct ieee80211_hw *hw, int if_id,
    struct ieee80211_sub_if_data *sdata;
    struct ieee80211_if_ap *bss = NULL;

-   bdev = dev_get_by_index(if_id);
+   bdev = dev_get_by_index(&init_net, if_id);
    if (bdev) {
        sdata = IEEE80211_DEV_TO_SUB_IF(bdev);
        bss = &sdata->u.ap;
diff --git a/net/mac80211/util.c b/net/mac80211/util.c
index 07686bd..c970996 100644
--- a/net/mac80211/util.c
+++ b/net/mac80211/util.c
@@ -20,6 +20,7 @@
#include <linux/if_arp.h>
#include <linux/wireless.h>
#include <linux/bitmap.h>
+#include <net/net_namespace.h>
#include <net/cfg80211.h>

#include "ieee80211_i.h"
@@ -318,7 +319,7 @@ __le16 ieee80211_generic_frame_duration(struct ieee80211_hw *hw, int
if_id,
    size_t frame_len, int rate)
{
    struct ieee80211_local *local = hw_to_local(hw);
-   struct net_device *bdev = dev_get_by_index(if_id);

```

```

+ struct net_device *bdev = dev_get_by_index(&init_net, if_id);
  struct ieee80211_sub_if_data *sdata;
  u16 dur;
  int erp;
@@ -342,7 +343,7 @@ __le16 ieee80211_rts_duration(struct ieee80211_hw *hw, int if_id,
{
  struct ieee80211_local *local = hw_to_local(hw);
  struct ieee80211_rate *rate;
- struct net_device *bdev = dev_get_by_index(if_id);
+ struct net_device *bdev = dev_get_by_index(&init_net, if_id);
  struct ieee80211_sub_if_data *sdata;
  int short_preamble;
  int erp;
@@ -378,7 +379,7 @@ __le16 ieee80211_ctstoself_duration(struct ieee80211_hw *hw, int if_id,
{
  struct ieee80211_local *local = hw_to_local(hw);
  struct ieee80211_rate *rate;
- struct net_device *bdev = dev_get_by_index(if_id);
+ struct net_device *bdev = dev_get_by_index(&init_net, if_id);
  struct ieee80211_sub_if_data *sdata;
  int short_preamble;
  int erp;
diff --git a/net/netrom/nr_route.c b/net/netrom/nr_route.c
index 24fe4a6..e943c16 100644
--- a/net/netrom/nr_route.c
+++ b/net/netrom/nr_route.c
@@ -580,7 +580,7 @@ static struct net_device *nr_ax25_dev_get(char *devname)
{
  struct net_device *dev;

- if ((dev = dev_get_by_name(devname)) == NULL)
+ if ((dev = dev_get_by_name(&init_net, devname)) == NULL)
    return NULL;

  if ((dev->flags & IFF_UP) && dev->type == ARPHRD_AX25)
@@ -598,7 +598,7 @@ struct net_device *nr_dev_first(void)
  struct net_device *dev, *first = NULL;

  read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
  if ((dev->flags & IFF_UP) && dev->type == ARPHRD_NETROM)
    if (first == NULL || strcmp(dev->name, first->name, 3) < 0)
      first = dev;
@@ -618,7 +618,7 @@ struct net_device *nr_dev_get(ax25_address *addr)
  struct net_device *dev;

  read_lock(&dev_base_lock);

```

```

- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if ((dev->flags & IFF_UP) && dev->type == ARPHRD_NETROM && ax25cmp(addr,
(ax25_address *)dev->dev_addr) == 0) {
        dev_hold(dev);
        goto out;
diff --git a/net/packet/af_packet.c b/net/packet/af_packet.c
index 30f2143..f14d66b 100644
--- a/net/packet/af_packet.c
+++ b/net/packet/af_packet.c
@@ -347,7 +347,7 @@ static int packet_sendmsg_spkt(struct kiocb *iocb, struct socket *sock,
    */

    saddr->spkt_device[13] = 0;
- dev = dev_get_by_name(saddr->spkt_device);
+ dev = dev_get_by_name(&init_net, saddr->spkt_device);
    err = -ENODEV;
    if (dev == NULL)
        goto out_unlock;
@@ -743,7 +743,7 @@ static int packet_sendmsg(struct kiocb *iocb, struct socket *sock,
    }

- dev = dev_get_by_index(ifindex);
+ dev = dev_get_by_index(&init_net, ifindex);
    err = -ENXIO;
    if (dev == NULL)
        goto out_unlock;
@@ -938,7 +938,7 @@ static int packet_bind_spkt(struct socket *sock, struct sockaddr *uaddr,
int add
    return -EINVAL;
    strcpy(name,uaddr->sa_data,sizeof(name));

- dev = dev_get_by_name(name);
+ dev = dev_get_by_name(&init_net, name);
    if (dev) {
        err = packet_do_bind(sk, dev, pkt_sk(sk)->num);
        dev_put(dev);
@@ -965,7 +965,7 @@ static int packet_bind(struct socket *sock, struct sockaddr *uaddr, int
addr_len

    if (sll->sll_ifindex) {
        err = -ENODEV;
- dev = dev_get_by_index(sll->sll_ifindex);
+ dev = dev_get_by_index(&init_net, sll->sll_ifindex);
        if (dev == NULL)
            goto out;
    }

```

```

@@ -1162,7 +1162,7 @@ static int packet_getname_spkt(struct socket *sock, struct sockaddr
*uaddr,
    return -EOPNOTSUPP;

    uaddr->sa_family = AF_PACKET;
- dev = dev_get_by_index(pkt_sk(sk)->ifindex);
+ dev = dev_get_by_index(&init_net, pkt_sk(sk)->ifindex);
    if (dev) {
        strcpy(uaddr->sa_data, dev->name, 15);
        dev_put(dev);
@@ -1187,7 +1187,7 @@ static int packet_getname(struct socket *sock, struct sockaddr *uaddr,
    sll->sll_family = AF_PACKET;
    sll->sll_ifindex = po->ifindex;
    sll->sll_protocol = po->num;
- dev = dev_get_by_index(po->ifindex);
+ dev = dev_get_by_index(&init_net, po->ifindex);
    if (dev) {
        sll->sll_hatype = dev->type;
        sll->sll_halen = dev->addr_len;
@@ -1239,7 +1239,7 @@ static int packet_mc_add(struct sock *sk, struct packet_mreq_max
*mreq)
    rtnl_lock();

    err = -ENODEV;
- dev = __dev_get_by_index(mreq->mr_ifindex);
+ dev = __dev_get_by_index(&init_net, mreq->mr_ifindex);
    if (!dev)
        goto done;

@@ -1293,7 +1293,7 @@ static int packet_mc_drop(struct sock *sk, struct packet_mreq_max
*mreq)
    if (--ml->count == 0) {
        struct net_device *dev;
        *mlp = ml->next;
- dev = dev_get_by_index(ml->ifindex);
+ dev = dev_get_by_index(&init_net, ml->ifindex);
        if (dev) {
            packet_dev_mc(dev, ml, -1);
            dev_put(dev);
@@ -1321,7 +1321,7 @@ static void packet_flush_mclist(struct sock *sk)
    struct net_device *dev;

    po->mclist = ml->next;
- if ((dev = dev_get_by_index(ml->ifindex)) != NULL) {
+ if ((dev = dev_get_by_index(&init_net, ml->ifindex)) != NULL) {
    packet_dev_mc(dev, ml, -1);
    dev_put(dev);
}

```

```

diff --git a/net/rose/rose_route.c b/net/rose/rose_route.c
index bbcbad1..a6e29fc 100644
--- a/net/rose/rose_route.c
+++ b/net/rose/rose_route.c
@@ -578,7 +578,7 @@ static struct net_device *rose_ax25_dev_get(char *devname)
{
    struct net_device *dev;

- if ((dev = dev_get_by_name(devname)) == NULL)
+ if ((dev = dev_get_by_name(&init_net, devname)) == NULL)
    return NULL;

    if ((dev->flags & IFF_UP) && dev->type == ARPHRD_AX25)
@@ -596,7 +596,7 @@ struct net_device *rose_dev_first(void)
    struct net_device *dev, *first = NULL;

    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if ((dev->flags & IFF_UP) && dev->type == ARPHRD_ROSE)
        if (first == NULL || strcmp(dev->name, first->name, 3) < 0)
            first = dev;
@@ -614,7 +614,7 @@ struct net_device *rose_dev_get(rose_address *addr)
    struct net_device *dev;

    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if ((dev->flags & IFF_UP) && dev->type == ARPHRD_ROSE && rosecmp(addr, (rose_address
*)dev->dev_addr) == 0) {
        dev_hold(dev);
        goto out;
@@ -631,7 +631,7 @@ static int rose_dev_exists(rose_address *addr)
    struct net_device *dev;

    read_lock(&dev_base_lock);
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if ((dev->flags & IFF_UP) && dev->type == ARPHRD_ROSE && rosecmp(addr, (rose_address
*)dev->dev_addr) == 0)
        goto out;
    }
}
diff --git a/net/sched/act_mirred.c b/net/sched/act_mirred.c
index 5795789..fd7bca4 100644
--- a/net/sched/act_mirred.c
+++ b/net/sched/act_mirred.c
@@ -20,6 +20,7 @@
#include <linux/rtnetlink.h>

```



```

#include <linux/module.h>
#include <linux/init.h>
+#include <net/net_namespace.h>
#include <net/netlink.h>
#include <net/pkt_sched.h>
#include <linux/tc_act/tc_mirred.h>
@@ -73,7 +74,7 @@ static int tcf_mirred_init(struct rtattr *rta, struct rtattr *est,
    parm = RTA_DATA(tb[TCA_MIRRED_PARMS-1]);

    if (parm->ifindex) {
- dev = __dev_get_by_index(parm->ifindex);
+ dev = __dev_get_by_index(&init_net, parm->ifindex);
    if (dev == NULL)
        return -ENODEV;
    switch (dev->type) {
diff --git a/net/sched/cls_api.c b/net/sched/cls_api.c
index 5f0fbca..0365797 100644
--- a/net/sched/cls_api.c
+++ b/net/sched/cls_api.c
@@ -154,7 +154,7 @@ replay:
    /* Find head of filter chain. */

    /* Find link */
- if ((dev = __dev_get_by_index(t->tcm_ifindex)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, t->tcm_ifindex)) == NULL)
        return -ENODEV;

    /* Find qdisc */
@@ -387,7 +387,7 @@ static int tc_dump_tfilter(struct sk_buff *skb, struct netlink_callback *cb)

    if (cb->nlh->nlmsg_len < NLMSG_LENGTH(sizeof(*tcm)))
        return skb->len;
- if ((dev = dev_get_by_index(tcm->tcm_ifindex)) == NULL)
+ if ((dev = dev_get_by_index(&init_net, tcm->tcm_ifindex)) == NULL)
        return skb->len;

    if (!tcm->tcm_parent)
diff --git a/net/sched/em_meta.c b/net/sched/em_meta.c
index 650f09c..e998961 100644
--- a/net/sched/em_meta.c
+++ b/net/sched/em_meta.c
@@ -291,7 +291,7 @@ META_COLLECTOR(var_sk_bound_if)
    } else {
        struct net_device *dev;

- dev = dev_get_by_index(skb->sk->sk_bound_dev_if);
+ dev = dev_get_by_index(&init_net, skb->sk->sk_bound_dev_if);
        *err = var_dev(dev, dst);

```

```

    if (dev)
        dev_put(dev);
diff --git a/net/sched/sch_api.c b/net/sched/sch_api.c
index efc383c..39d3278 100644
--- a/net/sched/sch_api.c
+++ b/net/sched/sch_api.c
@@ -607,7 +607,7 @@ static int tc_get_qdisc(struct sk_buff *skb, struct nlmsg_hdr *n, void *arg)
    struct Qdisc *p = NULL;
    int err;

- if ((dev = __dev_get_by_index(tcm->tcm_ifindex)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, tcm->tcm_ifindex)) == NULL)
    return -ENODEV;

    if (clid) {
@@ -674,7 +674,7 @@ replay:
    clid = tcm->tcm_parent;
    q = p = NULL;

- if ((dev = __dev_get_by_index(tcm->tcm_ifindex)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, tcm->tcm_ifindex)) == NULL)
    return -ENODEV;

    if (clid) {
@@ -881,7 +881,7 @@ static int tc_dump_qdisc(struct sk_buff *skb, struct netlink_callback *cb)
    s_q_idx = q_idx = cb->args[1];
    read_lock(&dev_base_lock);
    idx = 0;
- for_each_netdev(dev) {
+ for_each_netdev(&init_net, dev) {
    if (idx < s_idx)
        goto cont;
    if (idx > s_idx)
@@ -932,7 +932,7 @@ static int tc_ctl_tclass(struct sk_buff *skb, struct nlmsg_hdr *n, void *arg)
    u32 qid = TC_H_MAJ(clid);
    int err;

- if ((dev = __dev_get_by_index(tcm->tcm_ifindex)) == NULL)
+ if ((dev = __dev_get_by_index(&init_net, tcm->tcm_ifindex)) == NULL)
    return -ENODEV;

    /*
@@ -1115,7 +1115,7 @@ static int tc_dump_tclass(struct sk_buff *skb, struct netlink_callback
    *cb)

    if (cb->nlh->nlmsg_len < NLMSG_LENGTH(sizeof(*tcm)))
        return 0;
- if ((dev = dev_get_by_index(tcm->tcm_ifindex)) == NULL)

```

```

+ if ((dev = dev_get_by_index(&init_net, tcm->tcm_ifindex)) == NULL)
    return 0;

    s_t = cb->args[0];
diff --git a/net/sctp/ipv6.c b/net/sctp/ipv6.c
index 5953260..00a093a 100644
--- a/net/sctp/ipv6.c
+++ b/net/sctp/ipv6.c
@@ -843,7 +843,7 @@ static int sctp_inet6_bind_verify(struct sctp_sock *opt, union sctp_addr
*addr)
    if (type & IPV6_ADDR_LINKLOCAL) {
        if (!addr->v6.sin6_scope_id)
            return 0;
-   dev = dev_get_by_index(addr->v6.sin6_scope_id);
+   dev = dev_get_by_index(&init_net, addr->v6.sin6_scope_id);
    if (!dev)
        return 0;
    if (!ipv6_chk_addr(&addr->v6.sin6_addr, dev, 0)) {
@@ -874,7 +874,7 @@ static int sctp_inet6_send_verify(struct sctp_sock *opt, union sctp_addr
*addr)
    if (type & IPV6_ADDR_LINKLOCAL) {
        if (!addr->v6.sin6_scope_id)
            return 0;
-   dev = dev_get_by_index(addr->v6.sin6_scope_id);
+   dev = dev_get_by_index(&init_net, addr->v6.sin6_scope_id);
    if (!dev)
        return 0;
    dev_put(dev);
diff --git a/net/sctp/protocol.c b/net/sctp/protocol.c
index 0052ff4..193835d 100644
--- a/net/sctp/protocol.c
+++ b/net/sctp/protocol.c
@@ -176,7 +176,7 @@ static void sctp_get_local_addr_list(void)
    struct sctp_af *af;

    read_lock(&dev_base_lock);
-   for_each_netdev(dev) {
+   for_each_netdev(&init_net, dev) {
        __list_for_each(pos, &sctp_address_families) {
            af = list_entry(pos, struct sctp_af, list);
            af->copy_addrlist(&sctp_local_addr_list, dev);
diff --git a/net/socket.c b/net/socket.c
index bf5aba7..f123343 100644
--- a/net/socket.c
+++ b/net/socket.c
@@ -794,9 +794,9 @@ static ssize_t sock_aio_write(struct kiocb *iocb, const struct iovec *iov,
    */

```

```

static DEFINE_MUTEX(br_ioctl_mutex);
-static int (*br_ioctl_hook) (unsigned int cmd, void __user *arg) = NULL;
+static int (*br_ioctl_hook) (struct net *, unsigned int cmd, void __user *arg) = NULL;

-void brioclt_set(int (*hook) (unsigned int, void __user *))
+void brioclt_set(int (*hook) (struct net *, unsigned int, void __user *))
{
    mutex_lock(&br_ioctl_mutex);
    br_ioctl_hook = hook;
@@ -806,9 +806,9 @@ void brioclt_set(int (*hook) (unsigned int, void __user *))
EXPORT_SYMBOL(brioclt_set);

static DEFINE_MUTEX(vlan_ioctl_mutex);
-static int (*vlan_ioctl_hook) (void __user *arg);
+static int (*vlan_ioctl_hook) (struct net *, void __user *arg);

-void vlan_ioctl_set(int (*hook) (void __user *))
+void vlan_ioctl_set(int (*hook) (struct net *, void __user *))
{
    mutex_lock(&vlan_ioctl_mutex);
    vlan_ioctl_hook = hook;
@@ -837,16 +837,20 @@ EXPORT_SYMBOL(dlci_ioctl_set);
static long sock_ioctl(struct file *file, unsigned cmd, unsigned long arg)
{
    struct socket *sock;
+ struct sock *sk;
    void __user *argp = (void __user *)arg;
    int pid, err;
+ struct net *net;

    sock = file->private_data;
+ sk = sock->sk;
+ net = sk->sk_net;
    if (cmd >= SIOCDEVPRIVATE && cmd <= (SIOCDEVPRIVATE + 15)) {
- err = dev_ioctl(cmd, argp);
+ err = dev_ioctl(net, cmd, argp);
    } else
#ifdef CONFIG_WIRELESS_EXT
    if (cmd >= SIOCIWFIRST && cmd <= SIOCIWLAST) {
- err = dev_ioctl(cmd, argp);
+ err = dev_ioctl(net, cmd, argp);
    } else
#endif /* CONFIG_WIRELESS_EXT */
    switch (cmd) {
@@ -872,7 +872,7 @@ static long sock_ioctl(struct file *file, unsigned cmd, unsigned long arg)

    mutex_lock(&br_ioctl_mutex);
    if (br_ioctl_hook)

```

```

- err = br_ioctl_hook(cmd, argp);
+ err = br_ioctl_hook(net, cmd, argp);
  mutex_unlock(&br_ioctl_mutex);
  break;
  case SIOCGIFVLAN:
@@ -883,7 +887,7 @@ static long sock_ioctl(struct file *file, unsigned cmd, unsigned long arg)

  mutex_lock(&vlan_ioctl_mutex);
  if (vlan_ioctl_hook)
- err = vlan_ioctl_hook(argp);
+ err = vlan_ioctl_hook(net, argp);
  mutex_unlock(&vlan_ioctl_mutex);
  break;
  case SIOCADDLCI:
@@ -906,7 +910,7 @@ static long sock_ioctl(struct file *file, unsigned cmd, unsigned long arg)
  * to the NIC driver.
  */
  if (err == -ENOIOCTLCMD)
- err = dev_ioctl(cmd, argp);
+ err = dev_ioctl(net, cmd, argp);
  break;
}
return err;
diff --git a/net/tipc/eth_media.c b/net/tipc/eth_media.c
index 406f0d2..d6fc057 100644
--- a/net/tipc/eth_media.c
+++ b/net/tipc/eth_media.c
@@ -135,7 +135,7 @@ static int enable_bearer(struct tipc_bearer *tb_ptr)

/* Find device with specified name */

- for_each_netdev(pdev){
+ for_each_netdev(&init_net, pdev){
  if (!strcmp(pdev->name, driver_name, IFNAMSIZ)) {
    dev = pdev;
    break;
diff --git a/net/wireless/wext.c b/net/wireless/wext.c
index b8069af..e8b3409 100644
--- a/net/wireless/wext.c
+++ b/net/wireless/wext.c
@@ -673,22 +673,22 @@ static const struct seq_operations wireless_seq_ops = {

static int wireless_seq_open(struct inode *inode, struct file *file)
{
- return seq_open(file, &wireless_seq_ops);
+ struct seq_file *seq;
+ int res;
+ res = seq_open(file, &wireless_seq_ops);

```

```
+ if (!res) {
+ seq = file->private_data;
+ seq->private = get_net(PROC_NET(inode));
+ }
+ return res;
+}
+
+static int wireless_seq_release(struct inode *inode, struct file *file)
+{
+ struct seq_file *seq = file->private_data;
+ struct net *net = seq->private;
+ put_net(net);
+ return seq_release(inode, file);
+}

static const struct file_operations wireless_seq_fops = {
@@ -681,17 +696,22 @@ static const struct file_operations wireless_seq_fops = {
    .open    = wireless_seq_open,
    .read    = seq_read,
    .llseek  = seq_lseek,
-   .release = seq_release,
+   .release = wireless_seq_release,
};

-int __init wext_proc_init(void)
+int wext_proc_init(struct net *net)
{
    /* Create /proc/net/wireless entry */
- if (!proc_net_fops_create(&init_net, "wireless", S_IRUGO, &wireless_seq_fops))
+ if (!proc_net_fops_create(net, "wireless", S_IRUGO, &wireless_seq_fops))
    return -ENOMEM;

    return 0;
}
+
+void wext_proc_exit(struct net *net)
+{
+ proc_net_remove(net, "wireless");
+}
+
+/* CONFIG_PROC_FS */

/***** IOCTL SUPPORT *****/
@@ -1011,7 +1031,7 @@ static int ioctl_private_call(struct net_device *dev, struct ifreq *ifr,
    * Main IOCTL dispatcher.
    * Check the type of IOCTL and call the appropriate wrapper...
    */
-static int wireless_process_ioctl(struct ifreq *ifr, unsigned int cmd)
+static int wireless_process_ioctl(struct net *net, struct ifreq *ifr, unsigned int cmd)
```

```

{
    struct net_device *dev;
    iw_handler handler;
@@ -1020,7 +1040,7 @@ static int wireless_process_ioctl(struct ifreq *ifr, unsigned int cmd)
    * The copy_to/from_user() of ifr is also dealt with in there */

    /* Make sure the device exist */
- if ((dev = __dev_get_by_name(ifr->ifr_name)) == NULL)
+ if ((dev = __dev_get_by_name(net, ifr->ifr_name)) == NULL)
    return -ENODEV;

    /* A bunch of special cases, then the generic case...
@@ -1054,7 +1074,7 @@ static int wireless_process_ioctl(struct ifreq *ifr, unsigned int cmd)
}

/* entry point from dev ioctl */
-int wext_handle_ioctl(struct ifreq *ifr, unsigned int cmd,
+int wext_handle_ioctl(struct net *net, struct ifreq *ifr, unsigned int cmd,
    void __user *arg)
{
    int ret;
@@ -1066,9 +1086,9 @@ int wext_handle_ioctl(struct ifreq *ifr, unsigned int cmd,
    && !capable(CAP_NET_ADMIN))
    return -EPERM;

- dev_load(ifr->ifr_name);
+ dev_load(net, ifr->ifr_name);
    rtnl_lock();
- ret = wireless_process_ioctl(ifr, cmd);
+ ret = wireless_process_ioctl(net, ifr, cmd);
    rtnl_unlock();
    if (IW_IS_GET(cmd) && copy_to_user(arg, ifr, sizeof(struct ifreq)))
        return -EFAULT;
diff --git a/net/x25/x25_route.c b/net/x25/x25_route.c
index 060fcfa..86b5b4d 100644
--- a/net/x25/x25_route.c
+++ b/net/x25/x25_route.c
@@ -129,7 +129,7 @@ void x25_route_device_down(struct net_device *dev)
    */
    struct net_device *x25_dev_get(char *devname)
    {
- struct net_device *dev = dev_get_by_name(devname);
+ struct net_device *dev = dev_get_by_name(&init_net, devname);

    if (dev &&
        (!(dev->flags & IFF_UP) || (dev->type != ARPHRD_X25
--
1.5.3.rc6.17.g1911

```

---

Containers mailing list  
Containers@lists.linux-foundation.org  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---