
Subject: [PATCH net-2.6.25 5/10][NETNS][FRAGS]: Duplicate sysctl tables for new namespaces.

Posted by [Pavel Emelianov](#) on Tue, 22 Jan 2008 14:01:02 GMT

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

Each namespace has to have own tables to tune their different parameters, so duplicate the tables and register them.

All the tables in sub-namespaces are temporarily made read-only.

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

```
include/net/netns/ipv4.h | 1 +
include/net/netns/ipv6.h | 1 +
net/ipv4/ip_fragment.c   | 42 ++++++
net/ipv6/reassembly.c    | 41 ++++++
4 files changed, 79 insertions(+), 6 deletions(-)
```

diff --git a/include/net/netns/ipv4.h b/include/net/netns/ipv4.h

index 80680e0..15a0b05 100644

--- a/include/net/netns/ipv4.h

+++ b/include/net/netns/ipv4.h

@@ -16,6 +16,7 @@ struct sock;

struct netns_ipv4 {

#ifdef CONFIG_SYSCTL

struct ctl_table_header *forw_hdr;

+ struct ctl_table_header *frags_hdr;

#endif

struct ipv4_devconf *devconf_all;

struct ipv4_devconf *devconf_dflt;

diff --git a/include/net/netns/ipv6.h b/include/net/netns/ipv6.h

index 057c8e4..87ab56a 100644

--- a/include/net/netns/ipv6.h

+++ b/include/net/netns/ipv6.h

@@ -12,6 +12,7 @@ struct ctl_table_header;

struct netns_sysctl_ipv6 {

#ifdef CONFIG_SYSCTL

struct ctl_table_header *table;

+ struct ctl_table_header *frags_hdr;

#endif

struct inet_fragments_ctl frags;

int bindv6only;

diff --git a/net/ipv4/ip_fragment.c b/net/ipv4/ip_fragment.c

index 4f01334..c51e1a1 100644

--- a/net/ipv4/ip_fragment.c

```

+++ b/net/ipv4/ip_fragment.c
@@ -661,17 +661,53 @@ static struct ctl_table ip4_frags_ctl_table[] = {

static int ip4_frags_ctl_register(struct net *net)
{
+ struct ctl_table *table;
+ struct ctl_table_header *hdr;

- hdr = register_net_sysctl_table(net, net_ipv4_ctl_path,
- ip4_frags_ctl_table);
- return hdr == NULL ? -ENOMEM : 0;
+ table = ip4_frags_ctl_table;
+ if (net != &init_net) {
+ table = kmemdup(table, sizeof(ip4_frags_ctl_table), GFP_KERNEL);
+ if (table == NULL)
+ goto err_alloc;
+
+ table[0].mode &= ~0222;
+ table[1].mode &= ~0222;
+ table[2].mode &= ~0222;
+ table[3].mode &= ~0222;
+ table[4].mode &= ~0222;
+ }
+
+ hdr = register_net_sysctl_table(net, net_ipv4_ctl_path, table);
+ if (hdr == NULL)
+ goto err_reg;
+
+ net->ipv4.frags_hdr = hdr;
+ return 0;
+
+err_reg:
+ if (net != &init_net)
+ kfree(table);
+err_alloc:
+ return -ENOMEM;
+}
+
+static void ip4_frags_ctl_unregister(struct net *net)
+{
+ struct ctl_table *table;
+
+ table = net->ipv4.frags_hdr->ctl_table_arg;
+ unregister_net_sysctl_table(net->ipv4.frags_hdr);
+ kfree(table);
+}
+
+else
+static inline int ip4_frags_ctl_register(struct net *net)

```

```

{
    return 0;
}
+
+static inline void ip4_frags_ctl_unregister(struct net *net)
+{
+}
+}
+endif

static int ipv4_frags_init_net(struct net *net)
diff --git a/net/ipv6/reassembly.c b/net/ipv6/reassembly.c
index 241b2cc..0300dcb 100644
--- a/net/ipv6/reassembly.c
+++ b/net/ipv6/reassembly.c
@@ -670,17 +670,52 @@ static struct ctl_table ip6_frags_ctl_table[] = {

static int ip6_frags_sysctl_register(struct net *net)
{
+ struct ctl_table *table;
+ struct ctl_table_header *hdr;

- hdr = register_net_sysctl_table(net, net_ipv6_ctl_path,
- ip6_frags_ctl_table);
- return hdr == NULL ? -ENOMEM : 0;
+ table = ip6_frags_ctl_table;
+ if (net != &init_net) {
+ table = kmemdup(table, sizeof(ip6_frags_ctl_table), GFP_KERNEL);
+ if (table == NULL)
+ goto err_alloc;
+
+ table[0].mode &= ~0222;
+ table[1].mode &= ~0222;
+ table[2].mode &= ~0222;
+ table[3].mode &= ~0222;
+ }
+
+ hdr = register_net_sysctl_table(net, net_ipv6_ctl_path, table);
+ if (hdr == NULL)
+ goto err_reg;
+
+ net->ipv6.sysctl.frags_hdr = hdr;
+ return 0;
+
+err_reg:
+ if (net != &init_net)
+ kfree(table);
+err_alloc:
+ return -ENOMEM;

```

```

+}
+
+static void ip6_fragments_sysctl_unregister(struct net *net)
+{
+ struct ctl_table *table;
+
+ table = net->ipv6.sysctl.fragments_hdr->ctl_table_arg;
+ unregister_net_sysctl_table(net->ipv6.sysctl.fragments_hdr);
+ kfree(table);
+}
+
+static inline int ip6_fragments_sysctl_register(struct net *net)
+{
+ return 0;
+}
+
+static inline void ip6_fragments_sysctl_unregister(struct net *net)
+{
+}
+
+static int ipv6_fragments_init_net(struct net *net)
+
+
+1.5.3.4

```
