
Subject: [PATCH 1/6][NETNS]: Introduce a netns_core structure.
Posted by [Pavel Emelianov](#) on Mon, 31 Mar 2008 12:09:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

There's already some stuff on the struct net, that should better be folded into netns_core structure. I'm making the per-protocol inuse counter be per-net also, which is also a candidate for this, so introduce this structure and populate it a bit.

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

```
include/net/net_namespace.h | 6 ++----
include/net/netns/core.h    | 13 ++++++++
net/core/sysctl_net_core.c  | 12 ++++++-----
net/socket.c                | 2 +-
4 files changed, 22 insertions(+), 11 deletions(-)
create mode 100644 include/net/netns/core.h
```

```
diff --git a/include/net/net_namespace.h b/include/net/net_namespace.h
index f8f3d1a..c01d45f 100644
```

```
--- a/include/net/net_namespace.h
+++ b/include/net/net_namespace.h
@@ -8,6 +8,7 @@
#include <linux/workqueue.h>
#include <linux/list.h>
```

```
+#include <net/netns/core.h>
#include <net/netns/unix.h>
#include <net/netns/packet.h>
#include <net/netns/ipv4.h>
@@ -46,10 +47,7 @@ struct net {
```

```
    struct sock *rtnl; /* rtnetlink socket */

- /* core sysctls */
- struct ctl_table_header *sysctl_core_hdr;
- int sysctl_somaxconn;
-
```

```
+ struct netns_core core;
    struct netns_packet packet;
    struct netns_unix unix;
    struct netns_ipv4 ipv4;
```

```
diff --git a/include/net/netns/core.h b/include/net/netns/core.h
new file mode 100644
index 0000000..0e8c0f8
--- /dev/null
+++ b/include/net/netns/core.h
```

```

@@ -0,0 +1,13 @@
+ #ifndef __NETNS_CORE_H__
+ #define __NETNS_CORE_H__
+
+ struct ctl_table_header;
+
+ struct netns_core {
+ /* core sysctls */
+ struct ctl_table_header *sysctl_hdr;
+
+ int sysctl_somaxconn;
+ };
+
+ #endif
diff --git a/net/core/sysctl_net_core.c b/net/core/sysctl_net_core.c
index 130338f..5fc8010 100644
--- a/net/core/sysctl_net_core.c
+++ b/net/core/sysctl_net_core.c
@@ -127,7 +127,7 @@ static struct ctl_table net_core_table[] = {
 {
     .ctl_name = NET_CORE_SOMAXCONN,
     .procname = "somaxconn",
-    .data = &init_net.sysctl_somaxconn,
+    .data = &init_net.core.sysctl_somaxconn,
     .maxlen = sizeof(int),
     .mode = 0644,
     .proc_handler = &proc_dointvec
@@ -161,7 +161,7 @@ static __net_init int sysctl_core_net_init(struct net *net)
 {
     struct ctl_table *tbl, *tmp;

- net->sysctl_somaxconn = SOMAXCONN;
+ net->core.sysctl_somaxconn = SOMAXCONN;

     tbl = net_core_table;
     if (net != &init_net) {
@@ -178,9 +178,9 @@ static __net_init int sysctl_core_net_init(struct net *net)
     }
 }

- net->sysctl_core_hdr = register_net_sysctl_table(net,
+ net->core.sysctl_hdr = register_net_sysctl_table(net,
     net_core_path, tbl);
- if (net->sysctl_core_hdr == NULL)
+ if (net->core.sysctl_hdr == NULL)
     goto err_reg;

     return 0;

```

```

@@ -196,8 +196,8 @@ static __net_exit void sysctl_core_net_exit(struct net *net)
{
    struct ctl_table *tbl;

-   tbl = net->sysctl_core_hdr->ctl_table_arg;
-   unregister_net_sysctl_table(net->sysctl_core_hdr);
+   tbl = net->core.sysctl_hdr->ctl_table_arg;
+   unregister_net_sysctl_table(net->core.sysctl_hdr);
    BUG_ON(tbl == net_core_table);
    kfree(tbl);
}
diff --git a/net/socket.c b/net/socket.c
index 79e5382..9b5c917 100644
--- a/net/socket.c
+++ b/net/socket.c
@@ -1375,7 +1375,7 @@ asmlinkage long sys_listen(int fd, int backlog)

    sock = sockfd_lookup_light(fd, &err, &fput_needed);
    if (sock) {
-   somaxconn = sock_net(sock->sk)->sysctl_somaxconn;
+   somaxconn = sock_net(sock->sk)->core.sysctl_somaxconn;
        if ((unsigned)backlog > somaxconn)
            backlog = somaxconn;
    }

```

--
1.5.3.4
