Subject: [PATCH net-2.6.25 2/3][IPV6] Unify and cleanup calls to addrconf_sysctl_register
Posted by Pavel Emelianov on Fri, 30 Nov 2007 18:54:51 GMT
View Forum Message <> Reply to Message

Currently this call is (ab)used similar to devinet one - it
registers sysctls for devices and for the "default" confs, while
the "all" sysctls are registered separately. But unlike its
devinet brother, the passed inet6_device is needed.

The fix is to make a __addrconf_sysctl_register(), which registers
sysctls for all "devices" we need, including "default" and "all" :)

The original addrconf_sysctl_register() calls the introduced
function, passing the inet6_device, device name and ifindex (to
be used as procname and ctl_name) into it.

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

```
diff --git a/net/ipv6/addrconf.c b/net/ipv6/addrconf.c
index 2d2886a..8b93593 100644
--- a/net/ipv6/addrconf.c
+++ b/net/ipv6/addrconf.c
@@ -4118,12 +4118,11 @@ static struct addrconf_sysctl_table
  },
 };

-static void addrconf_sysctl_register(struct inet6_dev *idev, struct ipv6_devconf *p)
+static void __addrconf_sysctl_register(char *dev_name, int ctl_name,
+  struct inet6_dev *idev, struct ipv6_devconf *p)
 {
  int i;
- struct net_device *dev = idev ? idev->dev : NULL;
  struct addrconf_sysctl_table *t;
- char *dev_name = NULL;

  t = kmemdup(&addrconf_sysctl, sizeof(*t), GFP_KERNEL);
  if (t == NULL)
@@ -4133,13 +4132,6 @@ static void addrconf_sysctl_register(struct inet6_dev *idev, struct
ipv6_devconf
   t->addrconf_vars[i].data += (char*)p - (char*)&ipv6_devconf;
   t->addrconf_vars[i].extra1 = idev; /* embedded; no ref */
  }
- if (dev) {
-  dev_name = dev->name;
-  t->addrconf_dev[0].ctl_name = dev->ifindex;
```

```
- } else {
-  dev_name = "default";
-  t->addrconf_dev[0].ctl_name = NET_PROTO_CONF_DEFAULT;
- }

  /*
   * Make a copy of dev_name, because '.procname' is regarded as const
@@ -4150,6 +4142,7 @@ static void addrconf_sysctl_register(struct inet6_dev *idev, struct
ipv6_devconf
  if (!dev_name)
   goto free;

+ t->addrconf_dev[0].ctl_name = ctl_name;
  t->addrconf_dev[0].procname = dev_name;

  t->addrconf_dev[0].child = t->addrconf_vars;
@@ -4172,6 +4165,13 @@ out:
  return;
 }

+static void addrconf_sysctl_register(struct inet6_dev *idev,
+  struct ipv6_devconf *p)
+{
+ __addrconf_sysctl_register(idev->dev->name, idev->dev->ifindex,
+   idev, p);
+}
+
 static void addrconf_sysctl_unregister(struct ipv6_devconf *p)
 {
  if (p->sysctl) {
@@ -4270,9 +4270,10 @@ int __init addrconf_init(void)
  ipv6_addr_label_rtnl_register();

 #ifdef CONFIG_SYSCTL
- addrconf_sysctl.sysctl_header =
-  register_sysctl_table(addrconf_sysctl.addrconf_root_dir);
- addrconf_sysctl_register(NULL, &ipv6_devconf_dflt);
+ __addrconf_sysctl_register("all", NET_PROTO_CONF_ALL,
+   NULL, &ipv6_devconf);
+ __addrconf_sysctl_register("default", NET_PROTO_CONF_DEFAULT,
+   NULL, &ipv6_devconf_dflt);
 #endif

  return 0;
--
1.5.3.4
```

Subject: Re: [PATCH net-2.6.25 2/3][IPV6] Unify and cleanup calls to addrconf_sysctl_register
Posted by Herbert Xu on Sat, 01 Dec 2007 13:24:22 GMT
View Forum Message <> Reply to Message

On Fri, Nov 30, 2007 at 09:54:51PM +0300, Pavel Emelyanov wrote:
>
> +static void addrconf_sysctl_register(struct inet6_dev *idev,
> + struct ipv6_devconf *p)

Due to your simplification you no longer need the second argument
as it can now be derived from the first as is the case for IPv4.

So let's get rid of that while we're at it.

Thanks,
--
Visit Openswan at http://www.openswan.org/
Email: Herbert Xu ~{PmV>HI~} <herbert@gondor.apana.org.au>
Home Page: http://gondor.apana.org.au/~herbert/
PGP Key: http://gondor.apana.org.au/~herbert/pubkey.txt

Subject: [PATCH net-2.6.25 (resend) 2/3][IPV6] Unify and cleanup calls to addrconf_sysctl_register
Posted by Pavel Emelianov on Sat, 01 Dec 2007 13:45:09 GMT
View Forum Message <> Reply to Message

Currently this call is (ab)used similar to devinet one - it
registers sysctls for devices and for the "default" confs, while
the "all" sysctls are registered separately. But unlike its
devinet brother, the passed inet6_device is needed.

The fix is to make a __addrconf_sysctl_register(), which registers
sysctls for all "devices" we need, including "default" and "all" :)

The original addrconf_sysctl_register() calls the introduced
function, passing the inet6_device, device name and ifindex (to
be used as procname and ctl_name) into it.

Thanks to Herbert again for pointing out, that we can shrink the
argument list to 1 :)

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

---

diff --git a/net/ipv6/addrconf.c b/net/ipv6/addrconf.c

```
index 2d2886a..ea1673d 100644
--- a/net/ipv6/addrconf.c
+++ b/net/ipv6/addrconf.c
@@ -101,7 +101,7 @@
 #define TIME_DELTA(a,b) ((unsigned long)((long)(a) - (long)(b)))

 #ifdef CONFIG_SYSCTL
-static void addrconf_sysctl_register(struct inet6_dev *idev, struct ipv6_devconf *p);
+static void addrconf_sysctl_register(struct inet6_dev *idev);
 static void addrconf_sysctl_unregister(struct ipv6_devconf *p);
 #endif

@@ -400,7 +400,7 @@ static struct inet6_dev * ipv6_add_dev(struct net_device *dev)
          NET_IPV6_NEIGH, "ipv6",
          &ndisc_ifinfo_sysctl_change,
          NULL);
- addrconf_sysctl_register(ndev, &ndev->cnf);
+ addrconf_sysctl_register(ndev);
 #endif
  /* protected by rtnl_lock */
  rcu_assign_pointer(dev->ip6_ptr, ndev);
@@ -2386,7 +2386,7 @@ static int addrconf_notify(struct notifier_block *this, unsigned long
event,
           NET_IPV6, NET_IPV6_NEIGH, "ipv6",
           &ndisc_ifinfo_sysctl_change,
           NULL);
-   addrconf_sysctl_register(idev, &idev->cnf);
+   addrconf_sysctl_register(idev);
 #endif
    err = snmp6_register_dev(idev);
    if (err)
@@ -4118,12 +4118,11 @@ static struct addrconf_sysctl_table
  },
 };

-static void addrconf_sysctl_register(struct inet6_dev *idev, struct ipv6_devconf *p)
+static void __addrconf_sysctl_register(char *dev_name, int ctl_name,
+  struct inet6_dev *idev, struct ipv6_devconf *p)
 {
  int i;
- struct net_device *dev = idev ? idev->dev : NULL;
  struct addrconf_sysctl_table *t;
- char *dev_name = NULL;

  t = kmemdup(&addrconf_sysctl, sizeof(*t), GFP_KERNEL);
  if (t == NULL)
@@ -4133,13 +4132,6 @@ static void addrconf_sysctl_register(struct inet6_dev *idev, struct
ipv6_devconf
```

```
   t->addrconf_vars[i].data += (char*)p - (char*)&ipv6_devconf;
   t->addrconf_vars[i].extra1 = idev; /* embedded; no ref */
  }
- if (dev) {
-  dev_name = dev->name;
-  t->addrconf_dev[0].ctl_name = dev->ifindex;
- } else {
-  dev_name = "default";
-  t->addrconf_dev[0].ctl_name = NET_PROTO_CONF_DEFAULT;
- }

  /*
   * Make a copy of dev_name, because '.procname' is regarded as const
@@ -4150,6 +4142,7 @@ static void addrconf_sysctl_register(struct inet6_dev *idev, struct
ipv6_devconf
  if (!dev_name)
   goto free;

+ t->addrconf_dev[0].ctl_name = ctl_name;
  t->addrconf_dev[0].procname = dev_name;

  t->addrconf_dev[0].child = t->addrconf_vars;
@@ -4172,6 +4165,12 @@ out:
  return;
 }

+static void addrconf_sysctl_register(struct inet6_dev *idev)
+{
+ __addrconf_sysctl_register(idev->dev->name, idev->dev->ifindex,
+   idev, &idev->cnf);
+}
+
 static void addrconf_sysctl_unregister(struct ipv6_devconf *p)
 {
  if (p->sysctl) {
@@ -4270,9 +4269,10 @@ int __init addrconf_init(void)
  ipv6_addr_label_rtnl_register();

 #ifdef CONFIG_SYSCTL
- addrconf_sysctl.sysctl_header =
-  register_sysctl_table(addrconf_sysctl.addrconf_root_dir);
- addrconf_sysctl_register(NULL, &ipv6_devconf_dflt);
+ __addrconf_sysctl_register("all", NET_PROTO_CONF_ALL,
+   NULL, &ipv6_devconf);
+ __addrconf_sysctl_register("default", NET_PROTO_CONF_DEFAULT,
+   NULL, &ipv6_devconf_dflt);
 #endif
```

```
  return 0;
--
1.5.3.4
```

Subject: [PATCH net-2.6.25 (resend) 3/3][IPV6] Use ctl paths to register addrconf sysctls
Posted by Pavel Emelianov on Sat, 01 Dec 2007 13:46:41 GMT
View Forum Message <> Reply to Message

This looks very much like the patch for ipv4's devinet.

This is also intended to help us with the net namespaces
and saves the ipv6.ko size by ~320 bytes.

The difference from the first version is just the patch
offsets, that changed due to changes in the patch #2.

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

```
diff --git a/net/ipv6/addrconf.c b/net/ipv6/addrconf.c
index ea1673d..dbff389 100644
--- a/net/ipv6/addrconf.c
+++ b/net/ipv6/addrconf.c
@@ -3848,10 +3848,7 @@ static struct addrconf_sysctl_table
 {
  struct ctl_table_header *sysctl_header;
  ctl_table addrconf_vars[__NET_IPV6_MAX];
- ctl_table addrconf_dev[2];
- ctl_table addrconf_conf_dir[2];
- ctl_table addrconf_proto_dir[2];
- ctl_table addrconf_root_dir[2];
+ char *dev_name;
 } addrconf_sysctl __read_mostly = {
  .sysctl_header = NULL,
  .addrconf_vars = {
@@ -4072,50 +4069,6 @@ static struct addrconf_sysctl_table
    .ctl_name = 0, /* sentinel */
   }
  },
- .addrconf_dev = {
-  {
-   .ctl_name = NET_PROTO_CONF_ALL,
-   .procname = "all",
-   .mode  = 0555,
-   .child  = addrconf_sysctl.addrconf_vars,
```

```
-  },
-  {
-   .ctl_name = 0, /* sentinel */
-  }
- },
- .addrconf_conf_dir = {
-  {
-   .ctl_name = NET_IPV6_CONF,
-   .procname = "conf",
-   .mode  = 0555,
-   .child  = addrconf_sysctl.addrconf_dev,
-  },
-  {
-   .ctl_name = 0, /* sentinel */
-  }
- },
- .addrconf_proto_dir = {
-  {
-   .ctl_name = NET_IPV6,
-   .procname = "ipv6",
-   .mode  = 0555,
-   .child  = addrconf_sysctl.addrconf_conf_dir,
-  },
-  {
-   .ctl_name = 0, /* sentinel */
-  }
- },
- .addrconf_root_dir = {
-  {
-   .ctl_name = CTL_NET,
-   .procname = "net",
-   .mode  = 0555,
-   .child  = addrconf_sysctl.addrconf_proto_dir,
-  },
-  {
-   .ctl_name = 0, /* sentinel */
-  }
- },
 };

 static void __addrconf_sysctl_register(char *dev_name, int ctl_name,
@@ -4124,6 +4077,17 @@ static void __addrconf_sysctl_register(char *dev_name, int ctl_name,
  int i;
  struct addrconf_sysctl_table *t;

+#define ADDRCONF_CTL_PATH_DEV 3
+
+ struct ctl_path addrconf_ctl_path[] = {
```

```
+  { .procname = "net", .ctl_name = CTL_NET, },
+  { .procname = "ipv6", .ctl_name = NET_IPV6, },
+  { .procname = "conf", .ctl_name = NET_IPV6_CONF, },
+  { /* to be set */ },
+  { },
+ };
+
+
   t = kmemdup(&addrconf_sysctl, sizeof(*t), GFP_KERNEL);
   if (t == NULL)
    goto out;
@@ -4138,19 +4102,15 @@ static void __addrconf_sysctl_register(char *dev_name, int
ctl_name,
    * by sysctl and we wouldn't want anyone to change it under our feet
    * (see SIOCSIFNAME).
    */
- dev_name = kstrdup(dev_name, GFP_KERNEL);
- if (!dev_name)
+ t->dev_name = kstrdup(dev_name, GFP_KERNEL);
+ if (!t->dev_name)
    goto free;

- t->addrconf_dev[0].ctl_name = ctl_name;
- t->addrconf_dev[0].procname = dev_name;
-
- t->addrconf_dev[0].child = t->addrconf_vars;
- t->addrconf_conf_dir[0].child = t->addrconf_dev;
- t->addrconf_proto_dir[0].child = t->addrconf_conf_dir;
- t->addrconf_root_dir[0].child = t->addrconf_proto_dir;
+ addrconf_ctl_path[ADDRCONF_CTL_PATH_DEV].procname = t->dev_name;
+ addrconf_ctl_path[ADDRCONF_CTL_PATH_DEV].ctl_name = ctl_name;

- t->sysctl_header = register_sysctl_table(t->addrconf_root_dir);
+ t->sysctl_header = register_sysctl_paths(addrconf_ctl_path,
+    t->addrconf_vars);
   if (t->sysctl_header == NULL)
    goto free_procname;

@@ -4158,7 +4118,7 @@ static void __addrconf_sysctl_register(char *dev_name, int ctl_name,
   return;

  free_procname:
- kfree(dev_name);
+ kfree(t->dev_name);
  free:
   kfree(t);
  out:
@@ -4177,7 +4137,7 @@ static void addrconf_sysctl_unregister(struct ipv6_devconf *p)
```

```
   struct addrconf_sysctl_table *t = p->sysctl;
   p->sysctl = NULL;
   unregister_sysctl_table(t->sysctl_header);
-  kfree(t->addrconf_dev[0].procname);
+  kfree(t->dev_name);
   kfree(t);
  }
 }
--
1.5.3.4
```

Subject: Re: [PATCH net-2.6.25 (resend) 3/3][IPV6] Use ctl paths to register addrconf sysctls
Posted by Herbert Xu on Sat, 01 Dec 2007 14:00:05 GMT
View Forum Message <> Reply to Message

On Sat, Dec 01, 2007 at 04:46:41PM +0300, Pavel Emelyanov wrote:
> This looks very much like the patch for ipv4's devinet.
>
> This is also intended to help us with the net namespaces
> and saves the ipv6.ko size by ~320 bytes.
>
> The difference from the first version is just the patch
> offsets, that changed due to changes in the patch #2.
>
> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

Both applied. Thanks!
--
Visit Openswan at http://www.openswan.org/
Email: Herbert Xu ~{PmV>HI~} <herbert@gondor.apana.org.au>
Home Page: http://gondor.apana.org.au/~herbert/
PGP Key: http://gondor.apana.org.au/~herbert/pubkey.txt
