

---

Subject: [PATCH] Module for ip utility to support veth device (v.2.1)

Posted by [Pavel Emelianov](#) on Wed, 11 Jul 2007 09:33:55 GMT

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

---

The usage is

```
# ip link add [name] type veth [peer <name>] [mac <mac>] [peer_mac <mac>]
```

This version doesn't include the fix for ip/iplink.c as Patrick said that he had included it into his patches already.

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

---

```
Makefile | 6 +++-
link_veth.c | 89 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
veth.h | 14 ++++++++
3 files changed, 108 insertions(+), 1 deletion(-)
```

```
diff --git a/ip/Makefile b/ip/Makefile
index 9a5bfe3..b46bce3 100644
--- a/ip/Makefile
+++ b/ip/Makefile
@@ -8,8 +8,9 @@ RTMONOBJ=rtmon.o
ALLOBJ=$(IPOBJ) $(RTMONOBJ)
SCRIPTS=ifcfg rtpr routel routef
TARGETS=ip rtmon
+LIBS=link_veth.so
```

```
-all: $(TARGETS) $(SCRIPTS)
+all: $(TARGETS) $(SCRIPTS) $(LIBS)
```

```
ip: $(IPOBJ) $(LIBNETLINK) $(LIBUTIL)
```

```
@@ -24,3 +25,6 @@ clean:
```

```
LDLIBS += -ldl
LDFLAGS += -Wl,-export-dynamic
+
+%.so: %.c
+ $(CC) $(CFLAGS) -shared $< -o $@
diff --git a/ip/link_veth.c b/ip/link_veth.c
new file mode 100644
index 0000000..7760684
--- /dev/null
+++ b/ip/link_veth.c
@@ -0,0 +1,89 @@
+/*
```

```

+ * link_veth.c veth driver module
+ *
+ * This program is free software; you can redistribute it and/or
+ * modify it under the terms of the GNU General Public License
+ * as published by the Free Software Foundation; either version
+ * 2 of the License, or (at your option) any later version.
+ *
+ * Authors: Pavel Emelianov <xemul@openvz.org>
+ *
+ */
+
+#include <string.h>
+
+#include "utils.h"
+#include "ip_common.h"
+#include "veth.h"
+
+#define ETH_ALEN 6
+
+static void usage(void)
+{
+ printf("Usage: ip link add ... type veth "
+ "[peer <peer-name>] [mac <mac>] [peer_mac <mac>]\n");
+}
+
+static int veth_parse_opt(struct link_util *lu, int argc, char **argv,
+ struct nlmsg_hdr *hdr)
+{
+ __u8 mac[ETH_ALEN];
+
+ for (; argc != 0; argv++, argc--) {
+ if (strcmp(*argv, "peer") == 0) {
+ argv++;
+ argc--;
+ if (argc == 0) {
+ usage();
+ return -1;
+ }
+
+ addattr_l(hdr, 1024, VETH_INFO_PEER,
+ *argv, strlen(*argv));
+
+ continue;
+ }
+
+ if (strcmp(*argv, "mac") == 0) {
+ argv++;
+ argc--;

```

```

+ if (argc == 0) {
+     usage();
+     return -1;
+ }
+
+ if (hexstring_a2n(*argv, mac, sizeof(mac)) == NULL)
+     return -1;
+
+ addattr_l(hdr, 1024, VETH_INFO_MAC,
+     mac, ETH_ALEN);
+ continue;
+ }
+
+ if (strcmp(*argv, "peer_mac") == 0) {
+     argv++;
+     argc--;
+     if (argc == 0) {
+         usage();
+         return -1;
+     }
+
+     if (hexstring_a2n(*argv, mac, sizeof(mac)) == NULL)
+         return -1;
+
+     addattr_l(hdr, 1024, VETH_INFO_PEER_MAC,
+         mac, ETH_ALEN);
+     continue;
+ }
+
+ usage();
+ return -1;
+ }
+
+ return 0;
+}
+
+struct link_util veth_link_util = {
+    .id = "veth",
+    .parse_opt = veth_parse_opt,
+};
diff --git a/ip/veth.h b/ip/veth.h
new file mode 100644
index 0000000..b84a530
--- /dev/null
+++ b/ip/veth.h
@@ -0,0 +1,14 @@
+#ifndef __NET_VETH_H__
+#define __NET_VETH_H__

```

```
+
+enum {
+ VETH_INFO_UNSPEC,
+ VETH_INFO_MAC,
+ VETH_INFO_PEER,
+ VETH_INFO_PEER_MAC,
+
+ __VETH_INFO_MAX
+#define VETH_INFO_MAX (__VETH_INFO_MAX - 1)
+};
+
+#endif
```

---