NAME

inet − TCP/IP server

SYNOPSIS

inet

DESCRIPTION

Inet is the TCP/IP server. It is a device driver that interfaces between the file server and the low level ethernet device driver. The interface to this server is described in **ip**(4).

Inet starts as a normal process, reads a the configuration file **/etc/inet.conf** to see what it should do, and uses a few special low level system calls to turn itself into a server. The format of the configuration file is as follows:

Configuration

The inet configuration file is fairly simple, here is an example:

```
eth0 DP8390 0 { default; };
psip1;
```

It tells that network 0 (the one containing devices **eth0**, **ip0**, **tcp0** and **udp0**) uses the ethernet device driver handled by task "DP8390" at port 0. This network is marked as the default network, so most programs use it through the unnumbered devices like **/dev/tcp** or **/dev/udp**. Network 1 is a Pseudo IP network that can be used for a serial IP over a modem for instance.

The configuration file may look like a common configuration file as described by **configfile**(5), but it is currently just a simple subset allowing only what you see here. The following network definitions are possible:

eth*N task port* {*options*};

This sets up an ethernet with device name **/dev/eth***N*, built on the given ethernet device driver at the given port at that driver. (If a network driver manages two network cards then they are at port 0 and 1.)

eth*N* **vlan** *id* **eth***M* {*options*}; (Minix-vmd only)

The ethernet **eth***N* uses VLAN number *id* and is built on ethernet **eth***M*. A packet given to this network has a VLAN tag prefixed to it and is then handed over to another ethernet for transmission. Likewise a packet on that ethernet carrying the appropriate VLAN tag has this tag removed and is sent on to this network. The VLAN ethernet behaves like an ordinary ethernet as far as applications are concerned.

psip*N* {*options*};

Creates pseudo IP network **/dev/psip***N*, usable for IP over serial lines, tunnels and whatnot.

OPTIONS

Some options can be given between braces. Minix only understands one of these options, "default". Minix-vmd does them all, of course.

default;

Mark this network as the default network. Exactly one of the networks must be so marked. When **inet** is started it will check and create all the necessary network devices before becoming a server. To know what major device number to use it checks **/dev/ip**, so that device must already exist. It can be created by **MAKEDEV** if need be.

no ip;
no tcp;
no udp;

These options turn the IP, TCP, or UDP layer off. Inet will not enable the devices for these layers, and will deactivate code for these layers. Disabling IP will also disable TCP or UDP, because they need IP to function. An ethernet without an IP layer can be used as for stealth listening. An IP network without TCP or UDP can be used to pester students into creating the missing functionality. Keeps them off the streets, and maybe they'll learn something.

SEE ALSO

ip(4), **boot**(8).

NOTES

The number of networks that can be defined are 2 (Minix-86), 4 (Minix-386) or 16 (Minix-vmd). This limits both the total number and the highest device number you can use.

Getting a network administrator to give you a trunk or multi-VLAN port to run multiple networks on can be a challenge. It questions their idea that VLANs are separate networks, while in reality it is just one big ethernet.

ACKNOWLEDGMENTS

Cindy Crawford, for providing invaluable help debugging this server.

AUTHOR

Code: Philip Homburg <philip@cs.vu.nl>
Manual: Kees J. Bot <kjb@cs.vu.nl>