NAME

brctl – ethernet bridge administration

SYNOPSIS

brctl [command]

DESCRIPTION

brctl is used to set up, maintain, and inspect the ethernet bridge configuration in the Linux kernel.

An ethernet bridge is a device commonly used to connect different networks of ethernets together, so that these ethernets will appear as one ethernet to the participants.

Each of the ethernets being connected corresponds to one physical interface in the bridge. These individual ethernets are bundled into one bigger ('logical') ethernet, this bigger ethernet corresponds to the bridge network interface.

INSTANCES

The command **brctl addbr <name>** creates a new instance of the ethernet bridge. The network interface corresponding to the bridge will be called <name>.

The command **brctl delbr <name>** deletes the instance <name> of the ethernet bridge. The network interface corresponding to the bridge must be down before it can be deleted!

The command **brctl show** shows all current instances of the ethernet bridge.

PORTS

Each bridge has a number of ports attached to it. Network traffic coming in on any of these ports will be forwarded to the other ports transparently, so that the bridge is invisible to the rest of the network (i.e. it will not show up in *traceroute(8)*).

The command **brctl addif <brname> <ifname>** will make the interface <ifname> a port of the bridge <brname>. This means that all frames received on <ifname> will be processed as if destined for the bridge. Also, when sending frames on <brname>, <ifname> will be considered as a potential output interface.

The command **brctl delif <brname> <ifname>** will detach the interface <ifname> from the bridge <brname>.

The command **brctl show <brname>** will show some information on the bridge and its attached ports.

AGEING

The bridge keeps track of ethernet addresses seen on each port. When it needs to forward a frame, and it happens to know on which port the destination ethernet address (specified in the frame) is located, it can 'cheat' by forwarding the frame to that port only, thus saving a lot of redundant copies and transmits.

However, the ethernet address location data is not static data. Machines can move to other ports, network cards can be replaced (which changes the machine's ethernet address), etc.

brctl showmacs <brname> shows a list of learned MAC addresses for this bridge.

brctl setageing <brname> <time> sets the ethernet (MAC) address ageing time, in seconds. After <time> seconds of not having seen a frame coming from a certain address, the bridge will time out (delete) that

address from the Forwarding DataBase (fdb).

brctl setgcint <brname> <time> sets the garbage collection interval for the bridge <brname> to <time> seconds. This means that the bridge will check the forwarding database for timed out entries every <time> seconds.

SPANNING TREE PROTOCOL

Multiple ethernet bridges can work together to create even larger networks of ethernets using the IEEE 802.1d spanning tree protocol. This protocol is used for finding the shortest path between two ethernets, and for eliminating loops from the topology. As this protocol is a standard, Linux bridges will interwork properly with other third party bridge products. Bridges communicate with each other by sending and receiving BPDUs (Bridge Protocol Data Units). These BPDUs can be recognized by an ethernet destination address of 01:80:c2:00:00:00.

The spanning tree protocol can also be turned off (for those situations where it just doesn't make sense, for example when this Linux box is the only bridge on the LAN, or when you know that there are no loops in the topology.)

brctl(8) can be used for configuring certain spanning tree protocol parameters. For an explanation of these parameters, see the IEEE 802.1d specification (or send me an email). The default values should be just fine. If you don't know what these parameters mean, you probably won't feel the desire to tweak them.

brctl stp <bridge> <state> controls this bridge instance's participation in the spanning tree protocol. If <state> is "on" or "yes" the STP will be turned on, otherwise it will be turned off. When turned off, the bridge will not send or receive BPDUs, and will thus not participate in the spanning tree protocol. If your bridge isn't the only bridge on the LAN, or if there are loops in the LAN's topology, DO NOT turn this option off. If you turn this option off, please know what you are doing.

brctl setbridgeprio <bridge> <priority> sets the bridge's priority to <priority>. The priority value is an unsigned 16-bit quantity (a number between 0 and 65535), and has no dimension. Lower priority values are 'better'. The bridge with the lowest priority will be elected 'root bridge'.

brctl setfd <bridge> <time> sets the bridge's 'bridge forward delay' to <time> seconds.

brctl sethello <bridge> <time> sets the bridge's 'bridge hello time' to <time> seconds.

brctl setmaxage <bridge> <time> sets the bridge's 'maximum message age' to <time> seconds.

brctl setpathcost <bridge> <port> <cost> sets the port cost of the port <port> to <cost>. This is a dimensionless metric (from 1 to 65535, default is 100).

brctl setportprio <bridge> <port> <priority> sets the port <port>'s priority to <priority>. The priority value is a number from 0 (min) to 63 (max), default is 32, and has no dimension. This metric is used in the designated port and root port selection algorithms.

NOTES

brctl(8) is obsolete. Some features such as STP guard, hairpin mode, fast leave and root block are intentionally not implemented in this command. Instead use **bridge** command from **iproute2** package for a more full set of features.

SEE ALSO
      iptables(8)

AUTHOR
      Lennert Buytenhek <buytenh@gnu.org> Stephen Hemminger <stephen@networkplumber.org>