

**NAME**

**routed** — network routing daemon

**SYNOPSIS**

**routed** [ **-d** ] [ **-g** ] [ **-q** ] [ **-s** ] [ **-t** ] [*logfile*]

**DESCRIPTION**

**Routed** is invoked at boot time to manage the network routing tables. The routing daemon uses a variant of the Xerox NS Routing Information Protocol in maintaining up to date kernel routing table entries. It used a generalized protocol capable of use with multiple address types, but is currently used only for Internet routing within a cluster of networks.

In normal operation **routed** listens on the `udp(4)` socket for the `route(8)` service (see `services(5)`) for routing information packets. If the host is an internetwork router, it periodically supplies copies of its routing tables to any directly connected hosts and networks.

When **routed** is started, it uses the `SIOCGIFCONF` `ioctl(2)` to find those directly connected interfaces configured into the system and marked “up” (the software loopback interface is ignored). If multiple interfaces are present, it is assumed that the host will forward packets between networks. **Routed** then transmits a *request* packet on each interface (using a broadcast packet if the interface supports it) and enters a loop, listening for *request* and *response* packets from other hosts.

When a *request* packet is received, **routed** formulates a reply based on the information maintained in its internal tables. The *response* packet generated contains a list of known routes, each marked with a “hop count” metric (a count of 16, or greater, is considered “infinite”). The metric associated with each route returned provides a metric *relative to the sender*.

*Response* packets received by **routed** are used to update the routing tables if one of the following conditions is satisfied:

1. No routing table entry exists for the destination network or host, and the metric indicates the destination is “reachable” (i.e. the hop count is not infinite).
2. The source host of the packet is the same as the router in the existing routing table entry. That is, updated information is being received from the very internetwork router through which packets for the destination are being routed.
3. The existing entry in the routing table has not been updated for some time (defined to be 90 seconds) and the route is at least as cost effective as the current route.
4. The new route describes a shorter route to the destination than the one currently stored in the routing tables; the metric of the new route is compared against the one stored in the table to decide this.

When an update is applied, **routed** records the change in its internal tables and updates the kernel routing table. The change is reflected in the next *response* packet sent.

In addition to processing incoming packets, **routed** also periodically checks the routing table entries. If an entry has not been updated for 3 minutes, the entry’s metric is set to infinity and marked for deletion. Deletions are delayed an additional 60 seconds to insure the invalidation is propagated throughout the local internet.

Hosts acting as internetwork routers gratuitously supply their routing tables every 30 seconds to all directly connected hosts and networks. The response is sent to the broadcast address on nets capable of that function, to the destination address on point-to-point links, and to the router’s own address on other networks. The normal routing tables are bypassed when sending gratuitous responses. The reception of responses on each network is used to determine that the network and interface are functioning correctly. If no response is received on an interface, another route may be chosen to route around the interface, or the route may be

dropped if no alternative is available.

Options supported by **routed**:

- d** Enable additional debugging information to be logged, such as bad packets received.
- g** This flag is used on internetwork routers to offer a route to the “default” destination. This is typically used on a gateway to the Internet, or on a gateway that uses another routing protocol whose routes are not reported to other local routers.
- s** Supplying this option forces **routed** to supply routing information whether it is acting as an internetwork router or not. This is the default if multiple network interfaces are present, or if a point-to-point link is in use.
- q** This is the opposite of the **-s** option.
- t** If the **-t** option is specified, all packets sent or received are printed on the standard output. In addition, **routed** will not divorce itself from the controlling terminal so that interrupts from the keyboard will kill the process.

Any other argument supplied is interpreted as the name of file in which **routed**'s actions should be logged. This log contains information about any changes to the routing tables and, if not tracing all packets, a history of recent messages sent and received which are related to the changed route.

In addition to the facilities described above, **routed** supports the notion of “distant” *passive* and *active* gateways. When **routed** is started up, it reads the file `/etc/gateways` to find gateways which may not be located using only information from the `SIOGIFCONF ioctl(2)`. Gateways specified in this manner should be marked passive if they are not expected to exchange routing information, while gateways marked active should be willing to exchange routing information (i.e. they should have a **routed** process running on the machine). Routes through passive gateways are installed in the kernel's routing tables once upon startup. Such routes are not included in any routing information transmitted. Active gateways are treated equally to network interfaces. Routing information is distributed to the gateway and if no routing information is received for a period of the time, the associated route is deleted. Gateways marked *external* are also passive, but are not placed in the kernel routing table nor are they included in routing updates. The function of external entries is to inform **routed** that another routing process will install such a route, and that alternate routes to that destination should not be installed. Such entries are only required when both routers may learn of routes to the same destination.

The `/etc/gateways` is comprised of a series of lines, each in the following format:

```
<net | host> name1 gateway name2 metric value <passive | active | external>
```

The **net** or **host** keyword indicates if the route is to a network or specific host.

*Name1* is the name of the destination network or host. This may be a symbolic name located in `/etc/networks` or `/etc/hosts` (or, if started after `named(8)`, known to the name server), or an Internet address specified in “dot” notation; see `inet(3)`.

*Name2* is the name or address of the gateway to which messages should be forwarded.

*Value* is a metric indicating the hop count to the destination host or network.

One of the keywords **passive**, **active** or **external** indicates if the gateway should be treated as *passive* or *active* (as described above), or whether the gateway is *external* to the scope of the **routed** protocol.

Internetwork routers that are directly attached to the Arpanet or Milnet should use the Exterior Gateway Protocol (EGP) to gather routing information rather than using a static routing table of passive gateways. EGP is required in order to provide routes for local networks to the rest of the Internet system. Sites needing assistance with such configurations should contact the Computer Systems Research Group at Berkeley.

**FILES**

/etc/gateways for distant gateways

**SEE ALSO**

udp(4), icmp(4), XNSrouted(8), htable(8)

*Internet Transport Protocols*, XSIS 028112, Xerox System Integration Standard.

**BUGS**

The kernel's routing tables may not correspond to those of **routed** when redirects change or add routes. **Routed** should note any redirects received by reading the ICMP packets received via a raw socket.

**Routed** should incorporate other routing protocols, such as Xerox NS (XNSrouted(8)) and EGP. Using separate processes for each requires configuration options to avoid redundant or competing routes.

**Routed** should listen to intelligent interfaces, such as an IMP, to gather more information. It does not always detect unidirectional failures in network interfaces (e.g., when the output side fails).

**HISTORY**

The **routed** command appeared in 4.2BSD.