

NAME

rtsold, **rtsol** - router solicitation daemon

SYNOPSIS

rtsold [-dDfFimu1] [-M *script-name*] [-O *script-name*] [-A *script-name*] [-p *pidfile*] [-R *script-name*]
interface ...

rtsold [-dDfFimu1] [-M *script-name*] [-O *script-name*] [-A *script-name*] [-p *pidfile*] [-R *script-name*] -a

rtsol [-dDiu] [-M *script-name*] [-O *script-name*] [-A *script-name*] [-R *script-name*] *interface ...*

rtsol [-dDiu] [-M *script-name*] [-O *script-name*] [-A *script-name*] [-R *script-name*] -a

DESCRIPTION

rtsold is the daemon program to send ICMPv6 Router Solicitation messages on the specified interfaces. If a node (re)attaches to a link, **rtsold** sends some Router Solicitations on the link destined to the link-local scope all-routers multicast address to discover new routers and to get non link-local addresses.

rtsold should be used on IPv6 hosts (non-router nodes) only.

If you invoke the program as **rtsol**, it will transmit probes from the specified *interface*, without becoming a daemon. In other words, **rtsol** behaves as "**rtsold -f1 interfaces**".

Specifically, **rtsold** sends at most 3 Router Solicitations on an interface after one of the following events:

- Just after invocation of **rtsold** daemon.
- The interface is up after a temporary interface failure. **rtsold** detects such failures by periodically probing to see if the status of the interface is active or not. Note that some network cards and drivers do not allow the extraction of link state. In such cases, **rtsold** cannot detect the change of the interface status.
- Every 60 seconds if the **-m** option is specified and the **rtsold** daemon cannot get the interface status. This feature does not conform to the IPv6 neighbor discovery specification, but is provided for mobile stations. The default interval for router advertisements, which is on the order of 10 minutes, is slightly long for mobile stations. This feature is provided for such stations so that they can find new routers as soon as possible when they attach to another link.

Once **rtsold** has sent a Router Solicitation, and has received a valid Router Advertisement, it refrains from sending additional solicitations on that interface, until the next time one of the above events occurs.

When sending a Router Solicitation on an interface, **rtsold** includes a Source Link-layer address option if the interface has a link-layer address.

rtsold manages a per-interface parameter to detect if a separate protocol is needed for configuration

parameters other than host's addresses. At the invocation time, the flag is FALSE, and becomes TRUE when the daemon receives a router advertisement with the OtherConfig flag being set. A script file can be specified to deal with the case (see below). When **rtsold** start resending router solicitation messages by one of the conditions events, the daemon resets the parameter because the event may indicate a change on the attached link.

Upon receipt of signal SIGUSR1, **rtsold** will dump the current internal state into `/var/run/rtsold.dump`.

The options are as follows:

- a** Autoprobe outgoing interfaces. **rtsold** will try to find any non-loopback, IPv6-capable interfaces and send router solicitation messages on all of them.
- d** Enable debugging.
- D** Enable more debugging including the printing of internal timer information.
- f** Prevent **rtsold** from becoming a daemon (foreground mode). Warning messages are generated to standard error instead of `syslog(3)`.
- F** Explicitly configure the kernel to accept Router Advertisements and disable IPv6 forwarding. These settings are required for proper **rtsold** operation. Without this option, the current settings will be obeyed; if they are incompatible with proper operation, warning messages will be generated, but Router Solicitations will still be sent. The settings may be changed manually with `sysctl(8)` and `ifconfig(8)`.
- i** Transmit Router Solicitation packets immediately, without waiting the normal random (between 0 and 1 second) delay. This option should not be used on networks where it might result in congestion due to many hosts simultaneously (re)connecting and sending such packets.
- m** Enable mobility support. If this option is specified, **rtsold** sends probing packets to default routers that have advertised Router Advertisements when the node (re)attaches to an interface. Moreover, if the option is specified, **rtsold** periodically sends Router Solicitation on an interface that does not support SIOCGIFMEDIA ioctl.
- 1** Perform only one probe. Transmit Router Solicitation packets until at least one valid Router Advertisement packet has arrived on each *interface*, then exit.
- M** *script-name*
Specifies a supplement script file to handle the Managed Configuration flag of the router

advertisement. When the flag changes from FALSE to TRUE, **rtsold** will invoke *script-name* with a first argument of the receiving interface name and a second argument of the sending router address, expecting the script will then start a protocol for the managed configuration. *script-name* must be the absolute path from root to the script file, be a regular file, and be created by the same owner who runs **rtsold**.

-O *script-name*

Specifies a supplement script file to handle the Other Configuration flag of the router advertisement. When the flag changes from FALSE to TRUE, **rtsold** will invoke *script-name* with a first argument of the receiving interface name and a second argument of the sending router address, expecting the script will then start a protocol for the other configuration. The script will not be run if the Managed Configuration flag in the router advertisement is also TRUE. *script-name* must be the absolute path from root to the script file, be a regular file, and be created by the same owner who runs **rtsold**.

-A *script-name*

Specifies a supplement script file to always be called for the router advertisement. **rtsold** will invoke *script-name* with a first argument of the receiving interface name and a second argument of the sending router address. *script-name* must be the absolute path from root to the script file, be a regular file, and be created by the same owner who runs **rtsold**.

-p *pidfile*

Writes the process ID of **rtsold** to *pidfile* instead of the default PID file */var/run/rtsold.pid*.

-R *script-name*

Specifies a script to run when router advertisement options RDNSS (Recursive DNS Server) or DNSSL (DNS Search List) are encountered. The information of DNS servers and DNS search domains will be sent to standard input of this script. The *resolvconf(8)* script is used by default.

-u Specifies whether to add the source address of Router Advertisement messages to the interface name in the parameters of the RDNSS and DNSSL scripts.

If **-u** is specified, the interface name in the script parameters will be 'ifname:slaac:[RA-source-address]'.

Otherwise it will be 'ifname:slaac'.

FILES

/var/run/rtsold.pid The PID of the currently running **rtsold**.

/var/run/rtsold.dump Internal state dump file.

EXIT STATUS

The **rtsold** utility exits 0 on success, and >0 if an error occurs.

SEE ALSO

resolvconf(8), rtadvd(8), sysctl(8)

HISTORY

The **rtsold** command is based on the **rtsol** command, which first appeared in WIDE/KAME IPv6 protocol stack kit. **rtsol** is now integrated into rtsold(8).

BUGS

When a network card is removed and reinserted, the corresponding interface index may change. However, **rtsold** assumes such changes will not occur, and always uses the index that it got at invocation. As a result, **rtsold** may not work if you reinsert a network card. In such a case, **rtsold** should be killed and restarted.

The IPv6 autoconfiguration specification assumes a single-interface host. You may see kernel error messages if you try to autoconfigure a host with multiple interfaces. Also, it seems contradictory for **rtsold** to accept multiple *interface* arguments.