

NAME

sc_attach — simple scamper driver.

SYNOPSIS

```
sc_attach [-?dDv] [-c command] [-i infile] [-o outfile] [-O options]  
          [-p [ip:]port] [-P priority] [-R unix-remote] [-U unix-local]
```

DESCRIPTION

The **sc_attach** utility provides the ability to connect to a running `scamper(1)` instance, have a set of commands defined in a file be executed, and the output be written into a single file, in warts format. The options are as follows:

- ? prints a list of command line options and a synopsis of each.
- d prints each command sent to `scamper(1)` on stderr.
- D causes **sc_attach** to operate as a daemon.
- v prints the current revision of **sc_attach** and exits.
- c *command*
specifies the `scamper(1)` command to prepend to each address in the input file.
- i *infile*
specifies the name of the input file which consists of a sequence of `scamper(1)` commands or addresses (with the -c option), one per line. If '-' is specified, commands are read from stdin.
- o *outfile*
specifies the name of the output file to be written. The output file will use the warts format. If '-' is specified, output will be sent to stdout.
- O *options*
allows the behavior of **sc_attach** to be further tailored. The current choices for this option are:
 - **random**: shuffle the input commands randomly.
 - **impatient**: send commands to scamper without waiting for scamper to ask for them.
- p [*ip:*]*port*
specifies the IP address and port where a `scamper(1)` is accepting control socket connections. If an IP address is not specified, **sc_attach** connects to the specified port on the local host.
- P *priority*
specifies the mixing priority `scamper(1)` should assign to the source.
- R *unix-remote*
specifies the unix domain socket on the local host where a remote `scamper(1)` instance is accepting commands.
- U *unix-local*
specifies the unix domain socket on the local host where a local `scamper(1)` instance is accepting commands.

EXAMPLES

Given a set of commands in a file named `infile.txt`:

```
tbit -M 1280 -u 'http://www.example.com/' 2620:0:2d0:200::10  
trace -P udp-paris -M 192.0.2.1  
ping -P icmp-echo 192.0.32.10
```

and a `scamper(1)` instance listening on port 31337, then these commands can be executed using:

```
sc_attach -i infile.txt -o outfile.warts -p 31337
```

Given a set of addresses in a file named `infile2.txt`:

```
2620:0:2d0:200::10  
192.0.2.1  
192.0.32.10
```

these addresses can be pinged with `sc_attach` operating as a daemon with:

```
sc_attach -D -c 'ping' -i infile2.txt -o outfile2.warts -p 31337
```

SEE ALSO

`scamper(1)`, `sc_wartsdump(1)`, `sc_warts2json(1)`

AUTHORS

`sc_attach` was written by Matthew Luckie <mjl@luckie.org.nz>.