

**NAME**

saned – SANE network daemon

**SYNOPSIS**

**saned** [ **-a** [ *username* ] | **-d** [ *n* ] | **-s** [ *n* ] | **-h** ]

**DESCRIPTION**

**saned** is the SANE (Scanner Access Now Easy) daemon that allows remote clients to access image acquisition devices available on the local host.

**OPTIONS**

The **-a** flag requests that **saned** run in standalone daemon mode. In this mode, **saned** will detach from the console and run in the background, listening for incoming client connections; **inetd** is not required for **saned** operations in this mode. If the optional **username** is given after **-a**, **saned** will drop root privileges and run as this user (and group).

The **-d** and **-s** flags request that **saned** run in debug mode (as opposed to **inetd**(8) daemon mode). In this mode, **saned** explicitly waits for a connection request. When compiled with debugging enabled, these flags may be followed by a number to request debug info. The larger the number, the more verbose the debug output. E.g., **-d128** will request printing of all debug info. Debug level 0 means no debug output at all. The default value is 2. If flag **-d** is used, the debug messages will be printed to stderr while **-s** requests using syslog.

If **saned** is run from **inetd**, **xinetd** or **systemd**, no option can be given.

The **-h** flag displays a short help message.

**CONFIGURATION**

First and foremost: **saned** is not intended to be exposed to the internet or other non-trusted networks. Make sure that access is limited by **tcpwrappers** and/or a firewall setup. Don't depend only on **saned**'s own authentication. Don't run **saned** as root if it's not necessary. And do **not** install **saned** as **setuid** root.

The *saned.conf* configuration file contains both options for the daemon and the access list.

**data\_portrange** = *min\_port* - *max\_port*

Specify the port range to use for the data connection. Pick a port range between 1024 and 65535; don't pick a too large port range, as it may have performance issues. Use this option if your **saned** server is sitting behind a firewall. If that firewall is a Linux machine, we strongly recommend using the Netfilter *nf\_conntrack\_sane* module instead.

The access list is a list of host names, IP addresses or IP subnets (CIDR notation) that are permitted to use local SANE devices. IPv6 addresses must be enclosed in brackets, and should always be specified in their compressed form. Connections from localhost are always permitted. Empty lines and lines starting with a hash mark (#) are ignored. A line containing the single character "+" is interpreted to match any hostname. This allows any remote machine to use your scanner and may present a security risk, so this shouldn't be used unless you know what you're doing.

A sample configuration file is shown below:

```
# Daemon options
data_portrange = 10000 - 10100
# Access list
scan-client.somedomain.firm
# this is a comment
192.168.0.1
192.168.2.12/29
[::1]
[2001:db8:185e::42:12]/64
```

The case of the host names does not matter, so AHost.COM is considered identical to ahost.com.

## SERVER DAEMON CONFIGURATION

For **saned** to work properly in its default mode of operation, it is also necessary to add the appropriate configuration for *(x)inetd* or *systemd*. (see below). Note that your *inetd* must support IPv6 if you want to connect to **saned** over IPv6 ; *xinetd*, *openbsd-inetd* and *systemd* are known to support IPv6, check the documentation for your *inetd* daemon.

In the sections below the configuration for *inetd*, *xinetd* and *systemd* are described in more detail.

For the configurations below it is necessary to add a line of the following form to */etc/services*:

```
sane-port 6566/tcp # SANE network scanner daemon
```

The official IANA short name for port 6566 is "sane-port". The older name "sane" is now deprecated.

## INETD CONFIGURATION

It is required to add a single line to the *inetd* configuration file (*/etc/inetd.conf*)

The configuration line normally looks like this:

```
sane-port stream tcp nowait saned.saned @SBINDIR@/saned saned
```

However, if your system uses **tcpd**(8) for additional security screening, you may want to disable **saned** access control by putting "+" in *saned.conf* and use a line of the following form in */etc/inetd.conf* instead:

```
sane-port stream tcp nowait saned.saned /usr/sbin/tcpd @SBINDIR@/saned
```

Note that both examples assume that there is a **saned** group and a **saned** user. If you follow this example, please make sure that the access permissions on the special device are set such that **saned** can access the scanner (the program generally needs read and write access to scanner devices).

## XINETD CONFIGURATION

If *xinetd* is installed on your system instead of *inetd* the following example for */etc/xinetd.conf* may be helpful:

```
# default: off
# description: The sane server accepts requests
# for network access to a local scanner via the
# network.
service sane-port
{
    port          = 6566
    socket_type   = stream
    wait          = no
    user          = saned
    group         = saned
    server        = @SBINDIR@/saned
}
```

## SYSTEMD CONFIGURATION

**Saned** can be compiled with explicit *systemd* support. This will allow logging debugging information to be forwarded to the *systemd* journal. The *systemd* support requires compilation with the *systemd-devel* package installed on the system. this is the preferred option.

**Saned** can be used with *systemd* without the *systemd* integration compiled in, but then logging of debug information is not supported.

The *systemd* configuration is different for the 2 options, so both are described below.

### Systemd configuration for saned with systemd support compiled in

for the *systemd* configuration we need to add 2 configuration files in */etc/systemd/system*.

The first file we need to add here is called *saned.socket*. It shall have the following contents:

```
[Unit]
```

```
Description=saned incoming socket
```

```
[Socket]
ListenStream=6566
Accept=yes
MaxConnections=1
```

```
[Install]
WantedBy=sockets.target
```

The second file to be added is *saned@.service* with the following contents:

```
[Unit]
Description=Scanner Service
Requires=saned.socket

[Service]
ExecStart=/usr/sbin/saned
User=saned
Group=saned
StandardInput=null
StandardOutput=syslog
StandardError=syslog
Environment=SANE_CONFIG_DIR=@CONFIGDIR@
# If you need to debug your configuration uncomment the next line and
# change it as appropriate to set the desired debug options
# Environment=SANE_DEBUG_DLL=255 SANE_DEBUG_BJNP=5

[Install]
Also=saned.socket
```

You need to set an environment variable for **SANE\_CONFIG\_DIR** pointing to the directory where saned can find its configuration files. you will have to remove the # on the last line and set the variables for the desired debugging information if required. Multiple variables can be set by separating the assignments by spaces as shown in the example above.

Unlike *(x)inetd*, systemd allows debugging output from backends set using **SANE\_DEBUG\_XXX** to be captured. See the man-page for your backend to see what options are supported. With the service unit as described above, the debugging output is forwarded to the system log.

### Systemd configuration when saned is compiled without systemd support

This configuration will also work when Saned is compiled WITH systemd integration support, but it does not allow debugging information to be logged.

for systemd configuration for saned, we need to add 2 configuration files in */etc/systemd/system*.

The first file we need to add here is called *saned.socket*. It is identical to the version for systemd with the support compiled in. It shall have the following contents:

```
[Unit]
Description=saned incoming socket

[Socket]
ListenStream=6566
Accept=yes
MaxConnections=1

[Install]
```

```
WantedBy=sockets.target
```

The second file to be added is *saned@.service*. This one differs from the version with systemd integration compiled in:

```
[Unit]
Description=Scanner Service
Requires=saned.socket

[Service]
ExecStart=/usr/sbin/saned
User=saned
Group=saned
StandardInput=socket

Environment=SANE_CONFIG_DIR=/etc/sane.d

[Install]
Also=saned.socket
```

## FILES

*/etc/hosts.equiv*

The hosts listed in this file are permitted to access all local SANE devices. Caveat: this file imposes serious security risks and its use is not recommended.

*@CONFIGDIR@/saned.conf*

Contains a list of hosts permitted to access local SANE devices (see also description of **SANE\_CONFIG\_DIR** below).

*@CONFIGDIR@/saned.users*

If this file contains lines of the form

```
user:password:backend
```

access to the listed backends is restricted. A backend may be listed multiple times for different user/password combinations. The server uses MD5 hashing if supported by the client.

## ENVIRONMENT

### **SANE\_CONFIG\_DIR**

This environment variable specifies the list of directories that may contain the configuration file. Under UNIX, the directories are separated by a colon (':'), under OS/2, they are separated by a semi-colon(';'). If this variable is not set, the configuration file is searched in two default directories: first, the current working directory (".") and then in *@CONFIGDIR@*. If the value of the environment variable ends with the directory separator character, then the default directories are searched after the explicitly specified directories. For example, setting **SANE\_CONFIG\_DIR** to */tmp/config:* would result in directories *tmp/config*, *.*, and *@CONFIGDIR@* being searched (in this order).

## SEE ALSO

**sane(7)**, **scanimage(1)**, **xscanimage(1)**, **xcam(1)**, **sane-dll(5)**, **sane-net(5)**, **sane-"backendname"(5)**  
<http://www.penguin-breeder.org/?page=sane-net>

## AUTHOR

David Mosberger