

# Integrate Sophos Antivirus with Amavis in Hermes SEG

This guide will walk you through installing, configuring and integrating Sophos Antivirus for Linux with Amavis to be used in conjunction with ClamAV.

## Install Sophos Antivirus for Linux

First, download Sophos Antivirus for Linux from the link below. As of this writing, the file is named sav-linux-free-9.tgz.:

<https://www.sophos.com/en-us/products/free-tools/sophos-antivirus-for-linux.aspx>

Extract the file:

```
tar -xvzf sav-linux-free-9.tgz
```

This will create a sophos-av directory. Switch to that directory:

```
cd sophos-av
```

Run install.sh and follow the default options to install Sophos:

```
./install.sh
```

*NOTE: When prompted for the type of auto-update you want, select Sophos*

*NOTE: When prompted for the version you want, select Free*

*NOTE: By default Sophos will update itself automatically every 60-minutes as long as your server is connected to the Internet*

## Install SAVDI (Sophos Antivirus Dynamic Interface)

SAV Dynamic Interface will be used as the interface between Sophos Antivirus and Amavis using the SOPHIE protocol that Amavis already supports instead of the SPPP protocol that Amavis version 2.6.5 which comes with Ubuntu 12.04 LTS does not support.

Before you install SAV Dynamic Interface (SAVDI) on a server running Sophos Anti-Virus for Unix/Linux Version 9 you need to perform some additional steps before and after the install. First,

you must create symbolic links for libsavi.so.3 and libssp.so.0. You need to create those links so that SAVDI can locate these libraries during installation.

### 32-bit Servers ONLY

If you are using a 32-bit version of Ubuntu you only need to create a link for libssp.so.0 since the link for libsavi.so.3 is already created when you install Sophos Antivirus 9. Issue the following command:

```
ln -s /opt/sophos-av/lib/libssp.so.0 /usr/local/lib/libssp.so.0
```

*Note: If you have installed Sophos Anti-Virus to a non-default location then change the source path to this location.*

### 64-bit Servers ONLY

If you are using a 64-bit version of Ubuntu, you need to create links for both libssp.so.0 and libsavi.so.3 as follows:

```
ln -s /opt/sophos-av/lib64/libsavi.so.3 /usr/local/lib/libsavi.so.3
ln -s /opt/sophos-av/lib64/libssp.so.0 /usr/local/lib/libssp.so.0
```

*Note: If you have installed Sophos Anti-Virus to a non-default location then change the source path to this location.*

Now it's time to install SAVDI. Download SAVDI from <https://www.sophos.com/en-us/support/downloads/standalone-installers/sav-dynamic-interface.aspx>. Please note that you must have a Sophos username and password in order to download it.

Extract the .tar file (As of this writing, SAVDI was version 2.3)

```
tar -xvf savdi-23-linux-32bit.tar
```

This creates a savdi-install directory. Go to that directory:

```
cd savdi-install
```

Run savdi\_install.sh:

```
./savdi_install.sh
```

After installation, you will get the following warning because the virus data is detected in a non-default directory, it's ok to ignore:

```
Warning: Virus data found at /opt/sophos-av/lib/sav
```

Make a copy of /usr/local/savdi/savdid.conf file for backup just in case:

```
cp /usr/local/savdi/savdid.conf /usr/local/savdi/savdid.backup
```

Edit /usr/local/savdi/savdid.conf:

```
vi /usr/local/savdi/savdid.conf
```

Locate the below entries:

```
#virusdatadir: /var/sav/vdbs  
#idedir: /var/sav/vdbs
```

Change these to:

```
virusdatadir: /opt/sophos-av/lib/sav  
idedir: /opt/sophos-av/lib/sav
```

*Note: The '#' comment character needs to be removed from each entry*

Locate the following entry and delete everything underneath that line:

```
# Define a IP channel for localhost
```

Next, insert the following underneath the above line:

```
channel {  
  commprotocol {  
    type: UNIX  
    socket: /var/run/savdid/savdid.sock  
    user: amavis  
    group: amavis  
    requesttimeout: 120  
    sendtimeout: 2  
    recvtimeout: 5  
  }  
  scanprotocol {  
    type: SOPHIE  
    allowscandir: SUBDIR  
    maxscandata: 500000  
    maxmemorysize: 250000  
    tmpfilestub: /tmp/savid_tmp
```

```
}
scanner {
type: SAVI
inprocess: YES
maxscantime: 3
maxrequesttime: 10
deny: /dev
deny: /home
savigrp: GrpArchiveUnpack 0
savigrp: GrpInternet 1
savists: Xml 1
}
}
```

Save the file

In order to start savdid on system startup, you must create a script in `/etc/init.d/` directory:

```
vi /etc/init.d/savdid
```

Enter the following in that file:

```
#!/bin/sh
#
# savdid /etc/init.d/ initscript for savdid
#
#
# How this thing works:
# ${START} must be only what is needed for start-stop-daemon, DO NOT
# ADD ANY PARAMETERS HERE! we might use it for --test, for example.
# ${STOP} works just like ${START}, --signal is used with it.
#
# ${PARAMS} are the parameters to give the daemon when really starting
# it.
### BEGIN INIT INFO
# Provides: savdid
# Required-Start: $syslog $network $local_fs $remote_fs
# Required-Stop: $syslog $network $local_fs $remote_fs
# Should-Start:
# Should-Stop:
# Default-Start: 2 3 4 5
```

```
# Default-Stop: 0 1 6
# Short-Description: Starts savdid AntiVirus
# Description: Launches the savdid AntiVirus daemon
### END INIT INFO
PATH=/sbin:/bin:/usr/sbin:/usr/bin
DAEMON=/usr/local/bin/savdid
NAME=savdid
DAEMONNAME=savdid
DESC=savdid
PIDFILE=/var/run/savdid/${NAME}.pid
. /lib/lsb/init-functions
test -f ${DAEMON} || exit 0
set -e
START="--start --quiet --pidfile $PIDFILE --exec ${DAEMON}"
STOP="--stop --quiet --pidfile $PIDFILE"
PARAMS="-d"
case "$1" in
start)
echo -n "Starting $DESC: "
mkdir -p /var/run/savdid
if start-stop-daemon ${START} -- ${PARAMS} >/dev/null ; then
echo "savdid."
else
if start-stop-daemon --test ${START} >/dev/null 2>&1; then
echo "(failed)."
exit 1
else
echo "(already running)."
exit 0
fi
fi
;;
stop)
echo -n "Stopping $DESC: "
if start-stop-daemon ${STOP} --retry 10 >/dev/null ; then
echo "savdid."
else
if start-stop-daemon --test ${START} >/dev/null 2>&1; then
echo "(not running)."
exit 0
```

```
else
echo "(failed).\"
exit 1
fi
fi
;;
restart|force-reload)
$0 stop
exec $0 start
;;
status)
status_of_proc -p $PIDFILE $DAEMON $NAME && exit 0 || exit $?
;;
*)
N=/etc/init.d/savdid
echo \"Usage: $N {start|stop|restart|force-reload|status}\" >&2
exit 1
;;
esac
exit 0
```

Save the file and make it executable:

```
chmod +x /etc/init.d/savdid
```

Next, we need to make sure the service we just created will start during system startup. First, install chkconfig:

```
apt-get install chkconfig
```

Next, run chkconfig savdid:

```
chkconfig savdid
```

You should get the following output:

```
savdid off
```

So, we need to activate the savdid service. Run the following command:

```
chkconfig savdid on
```

In my system, running the command above gave me the following error:

```
/sbin/insserv: No such file or directory
```

This can be easily resolved by creating the following link:

```
ln -s /usr/lib/insserv/insserv /sbin/insserv
```

and then run the "chkconfig savdid on" command again. After the command completes running, run the following command again:

```
chkconfig savdid
```

Should output the following:

```
savdid on
```

Now, start the savdid service:

```
service savdid start
```

Next, edit /etc/amavis/conf.d/15-av\_scanners:

```
vi /etc/amavis/conf.d/15-av_scanners
```

Locate the @av\_scanners line, uncomment the 'Sophie' entry and make it look like below (Note how we point it to the savdid socket file with /var/run/savdid/savdid.sock):

```
['Sophie',  
 \&ask_daemon, [ "{}\n", '/var/run/savdid/savdid.sock'],  
 qr/(?x)^ 0+ ( : | [\000\r\n]* $)/m, qr/(?x)^ 1 ( : | [\000\r\n]* $)/m,  
 qr/(?x)^ [-+]? \d+ : (.*) [\000\r\n]* $/m ],
```

Save the file & Restart Amavis:

```
service amavis restart
```

Look for the following lines in /var/log/mail.log:

```
smtp amavis[5181]: Using primary internal av scanner code for Sophie  
smtp amavis[5181]: Using primary internal av scanner code for ClamAV-clamd  
smtp amavis[5181]: Found secondary av scanner ClamAV-clamscan at /usr/bin/clamscan
```

Test Sophos integration is working by monitoring the `/var/tmp/savdi/log/xxxxxx.log` file where `xxxxxx` is today's date (Note any errors with `savdid` will be logged in this file as well):

```
tail -f /var/tmp/savdi/log/160325.log
```

Send the EICAR virus test file to one of your recipients and ensure an entry similar to the one below is logged in the `/var/tmp/savdi/log/xxxxxx.log` file:

```
160325:070020 [56F510E6/1] 00030405 Threat found  
Identity: 'EICAR-AV-Test' "/var/lib/amavis/tmp/amavis-20160325T062724-05186/parts/p001"
```

Finally, reboot your system and ensure the `savdid` service has started by running the following command:

```
ps -A|grep savdid
```

If the service started, you should see a message similar to below:

```
2201 ? 00:00:00 savdid  
2203 ? 00:00:05 savdid
```

That's it! Enjoy your server with additional protection from Sophos AV.

This guide was possible thanks to the invaluable contributions of Peter Kieser  
<https://peterkieser.com/>.

---

Revision #1

Created 30 December 2020 12:11:50 by Dino Edwards

Updated 8 February 2021 23:03:25 by Dino Edwards