

RuleUpdates

Contents

- [Notes / FAQ](#)
- [Using sa-update](#)
 - [What is sa-update?](#)
 - [Channels](#)
 - [sa-update commandline](#)
 - [Channel / Location Related](#)
 - [GPG Related](#)
 - [Other](#)
 - [Examples](#)
 - [Installed Updates](#)
 - [Publishing channels for sa-update](#)
 - [SARE Channels](#)
 - [The Backend](#)

Note: This page is under construction and shouldn't be considered "complete". If there are any questions, etc, please feel free to contact the users at spamassassin.apache.org [list](#) for more information.

Notes / FAQ

- Currently, for 3.1.1+ and 3.2.0, to use any channel for updates requires that updates.spamassassin.org also be used. This is because once the update directory exists, the [SpamAssassin](#) modules expect to find all rules in that directory.

Can plugins be distributed via updates/channels?

From a technical standpoint, updates can contain plugins. However, the default channel, updates.spamassassin.org, will not publish plugins using this method at this time since people are likely not ready to accept automatic code updates.

After sa-update completes, do I have to move the files somewhere for them to be used?

No. By default, sa-update and the [SpamAssassin](#) modules use the same location for updates. This means that after a successful update run, the new rules are available for use. As usual, if using spamd, a restart is required for the new rules to be read in and enabled.

Should I use "--updatedir" to put updates in the default rules directory (ie: /usr/share/spamassassin), or the site rules directory (ie: /etc/mail/spamassassin)?

No! Those two directories have specific uses, and attempting to install updates to that directory will likely cripple your [SpamAssassin](#) installation. This is because when an update is installed, all previous files in the directory are removed first.

If I use "--updatedir" to install an update in a subdirectory of my site rules directory (ie: /etc/mail/spamassassin), a number of my local settings (typically set via local.cf) no longer function. Why?

This can happen if the update channel creates a channel file with a name lexically after the local settings, such as "updates_spamassassin_org.cf". For more information, read the "Using sa-update" section below, and also refer to the man page for "spamassassin".

What do I need to do with-respect-to sa-update after a SpamAssassin upgrade?

Whenever you upgrade the version of [SpamAssassin](#) that's installed, the directory where the updates are expected to be changes (it's based on the version). So whenever you upgrade, you will want to run sa-update for all of the channels that you want to have installed.

After upgrading SpamAssassin several times, there are a number of directories .../3.001001, .../3.001002, .../3.001003, etc. What should I do with them?

You definitely want to keep the latest version that matches the version of [SpamAssassin](#) that you have installed. However, the older versions can be removed whenever you like.

How often should I run sa-update?

As often as you like. It typically depends on what time-frame is comfortable for you, and how quickly channels are going to be publishing updates. Generally speaking, once a day is a good starting point.

After sa-update was used, the report contact setting becomes @@.

This was fixed in 3.1.4. This issue was tracked in [bug 4862](#).

Can I use sa-update as non-root too?

Of course. It works just fine as non-root for personal installations. Updates end up in \$HOME/var, which is where your [SingleUserUnixInstall](#) of [SpamAssassin](#) will look for them.

What if I need update requests to go through a proxy server?

sa-update uses the `LWP::UserAgent` module, which allows certain environment variables to be set so that requests use defined proxy servers. The main one of interest is "http_proxy", which should be set to an URL defining the proxy. ie: `export http_proxy='http://proxy.example.com:8080/'`

For more information about how to use sa-update and how it works, please read below.

Using sa-update

What is sa-update?

The goal of sa-update is to download new configuration files (rules, scores, etc.) so that [SpamAssassin](#) will use them to better catch spam and/or to avoid catching ham messages. The main reason to use sa-update is that the old method of disseminating rules, releasing a new version [SpamAssassin](#), is a lengthy process that can take many months. Spam is rapidly changing, and new rules are often written in response. With sa-update, those rules can quickly (potentially within minutes) be distributed and the new spam caught.

Simply put, sa-update allows rules to be distributed as they are developed, while full [SpamAssassin](#) releases can focus on bug fixes and new features.

Channels

Simply put, channels are locations where sa-update can download rule and configuration files. By default, sa-update will use the `updates.spamassassin.org` channel to receive official updates from the [SpamAssassin](#) project, but anyone can create a channel and publish updates. By default, sa-update (and spamassassin) expect to find updates in the `/var/lib/spamassassin/<spamassassin version>` directory, which will have each channel in its own directory underneath. Each channel will also have a channel cf file and (optionally) a channel pre file to load the update's configurations in the update's parent directory. For example:

```
/var/lib/spamassassin
`-- 3.001004
   |-- updates_spamassassin_org
   `-- updates_spamassassin_org.cf
```

Shows the channel `updates.spamassassin.org` available for [SpamAssassin](#) 3.1.4, underneath the `/var/lib/spamassassin` directory.

For more information about what makes up a channel and how it all works together, please see the *Publishing channels for sa-update* section below.

NOTE: Once the `/var/lib/spamassassin/<spamassassin version>` directory exists, spamassassin expects to find all rules underneath that directory, so make sure that the first time you run sa-update it completes successfully (see below for information about running in debug mode).

sa-update commandline

sa-update has several parameters that can be passed via the commandline.

Channel / Location Related

`--updatedir <path>::` sa-update by default places updates in the `/var/lib/spamassassin/<spamassassin version>` directory. If updates should go into a different directory, specify it via this option.

`--channel <channel>::` sa-update by default only uses the `updates.spamassassin.org` channel for updates. If other channels should be used, specify it with this option. For multiple channels, specify this option multiple times. ie:

```
sa-update --channel updates.spamassassin.org --channel saupdates.example.com
```

`--channelfile <file>::` If multiple channels are going to be used at once, it may be easier to write the channels to a file, and then use this option to point to that file. ie:

```
$ rm -f channels ; touch channels
$ echo updates.spamassassin.org >> channels
$ echo saupdates.example.com >> channels
$ sa-update --channelfile channels
```

--checkonly

Check for update availability, do not install.

--allowplugins

Allow updates to load plugin code (DANGEROUS). You should never enable this for 3rd party update channels, since plugins can execute unrestricted code on your system!

--refreshmirrors

Force the MIRRORED.BY file to be updated.

--install <filename>

Install updates directly from this file. Signature verification will read from filename.asc and filename.sha1 .

GPG Related

--(no)gpg:: By default, sa-update will require the use of GPG signatures to verify that downloaded updates are in fact legitimate. However, there may be channels which do not publish GPG signatures, or the system may not have GPG installed. In these situations, specify the

```
--nogpg
```

option to disable the use of GPG. Note: By using the `--gpgkey` or `--gpgkeyfile` options as shown below, `--gpg` is automatically enabled.

--gpg homedir <path>:: sa-update tries to keep its keys separate from the user's keys by using a different directory for the keyrings (passed to gpg via its `--homedir` option). By default, the location is `/etc/mail/spamassassin/sa-update-keys`. If a different location is desired, use this option to specify it.

--gpgkey <key id>:: Specify which GPG key ids should be trusted to sign update packages. If there are multiple keys, use this option multiple times to list them. Generally it's safer to specify the whole key fingerprint, but it is more common to see simply the last 8 hex digits used. ie:

```
sa-update --gpgkey 26C900A46DD40CD5AD24F6D7DEE01987265FA05B --gpgkey 5244EC45
```

--gpgkeyfile <file>:: Similar to `channelfile`, if there are multiple keys to be trusted, it may be easier to specify them in a file and then use this option to point sa-update at the file. ie:

```
$ rm -f gpgkeys ; touch gpgkeys
$ echo 26C900A46DD40CD5AD24F6D7DEE01987265FA05B >> gpgkeys
$ echo 5244EC45 >> gpgkeys
$ sa-update --gpgkeyfile gpgkeys
```

--import <file>

Import GPG key(s) from <file> into sa-update's keyring. Use multiple times for multiple files.

Other

-D, --debug [area=n,...]

Show debugging information. This can be useful just to see what sa-update is doing, but is also useful to debug problems, etc. This option takes the same optional parameter (DebugChannels) as the other standard [SpamAssassin](#) tools.

-V, --version

Display which version of sa-update is installed. sa-update is versioned by Subversion revision number as opposed to being tied to a specific [SpamAssassin](#) version.

-v, --verbose

Be more verbose, like print updated channel names.

-h, --help

Print usage message.

More information is available via the POD/man page for sa-update.

Examples

```
sa-update && service spamassassin restart
```

This is potentially the most simple example of using sa-update. It will see if an update is available, and download the update and lint-test it as necessary. If there was an update and no problems were detected, sa-update returns 0 and the

```
service spamassassin restart
```

command is run (used on Linux machines to restart the spamassassin service (spamd)).

```
sa-update -D
```

If there is a problem with doing updates, or just some curiosity about what is happening, run sa-update in debug mode and it will show you what it is doing step-by-step.

```
sa-update -D --updatedir /tmp/updates
```

To perform updates without attempting to put the files into the system-wide location (either for users or to test, etc,) use the `--updatedir` option to aim at a different directory for which to put the updates.

Installed Updates

When updates are downloaded, they are put into a directory under the local state dir (default `/var/lib/spamassassin/<spamassassin version>`) similar to:

```
/var/lib/spamassassin
`-- 3.001004
    |-- updates_spamassassin_org
    `-- updates_spamassassin_org.cf
```

The files from the update go into `updates_spamassassin_org`, and the `*.cf` files are then included by `updates_spamassassin_org.cf`, which also keeps track of what update version is installed. Therefore, if it is desired to change the update directory, the `.cf` and the update directory will exist there.

Publishing channels for sa-update

See [PublishingRuleUpdates](#).

SARE Channels

Unfortunately the SARE ruleset has been discontinued in 2009. (Historical details of various channels from SARE can be seen at [SareChannels](#))

The Backend

Details of the rule-update generation backend at updates.spamassassin.org can be read at [SaUpdateBackend](#).