

PublishingRuleUpdates

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.

For info on how a [SpamAssassin](#) committer can update the updates.spamassassin.org channel see [ManualRuleUpdates](#).

Publishing channels for sa-update

Channels are fairly simple to setup and use. Simply put, updates are files contained in a standard tar.gz archive, distributed via HTTP. To allow for frequent update checks from clients, a lightweight method (DNS queries) is used to specify what update version is the latest for any given version of [SpamAssassin](#). sa-update also uses a mirror file which lists all of the URLs where the update can be downloaded from, optionally including weights for different mirrors.

For more info on using "sa-update" to download rule updates, see [RuleUpdates](#).

How does it work?

When looking for an update, sa-update reverses the version and makes a DNS TXT query for `<version>.<channel>`. ie: Running 3.1.1's sa-update, the default updates.spamassassin.org channel causes a DNS lookup for `1.1.3.updates.spamassassin.org`. The query is for a TXT record containing the update number, which should be a monotonically increasing value. Assuming an update is necessary, sa-update will then read the `MIRRORED.BY` file (downloading it first if necessary from the URL found in a DNS TXT record `mirrors.<channel>`).

Summary

DNS Records:

- mirrors.<channel> TXT "http://URL/TO/MIRRORS/FILE"

```
mirrors.updates.spamassassin.org TXT "http://spamassassin.apache.org/updates/MIRRORED.BY"
```

- <version>.<channel> TXT "UPDATE_NUMBER"

```
*.1.3.updates.spamassassin.org TXT "386156"
```

Mirrors file:

- List of URLs which contain the updates

```
http://buildbot.spamassassin.org.nyud.net:8090/updatesstage/
```

Update file:

- Only files (no directories) contained in a tar.gz archive file named `UPDATE_NUMBER.tar.gz`
- A sha1sum file named `UPDATE_NUMBER.tar.gz.sha1` with the output of

```
sha1sum UPDATE_NUMBER.tar.gz
```

- Recommended is a detached GPG signature for the update named `UPDATE_NUMBER.tar.gz.asc` via something like

```
gpg -bas UPDATE_NUMBER.tar.gz
```

. Optionally this can be skipped, but users need to use "--nogpg" with sa-update

Examples

Here is a short example of how an update for [SpamAssassin](#) 3.1.x would be published. By convention, we use the svn version of the directory as the update number.

```

$ svnversion
386156
$ ls -al
total 528
drwxr-xr-x  3 felicity fame  4096 Mar 14 13:10 .
drwxr-xr-x  4 felicity fame    27 Feb 19 14:55 ..
-rw-r--r--  1 felicity fame  5479 Feb 19 14:55 10_misc.cf
-rw-r--r--  1 felicity fame  8112 Mar 13 02:49 20_advance_fee.cf
-rw-r--r--  1 felicity fame  1602 Feb 19 14:55 20_anti_ratware.cf
-rw-r--r--  1 felicity fame  6690 Mar 13 02:49 20_body_tests.cf
-rw-r--r--  1 felicity fame  1534 Mar 13 02:49 20_compensate.cf
[...]
-rw-r--r--  1 felicity fame 33266 Mar 11 18:13 50_scores.cf
-rw-r--r--  1 felicity fame  8564 Mar 14 13:10 80_additional.cf
$ tar -cf 386156.tar *.cf
$ ls -la 386156.tar
-rw-r--r--  1 felicity fame 471040 Mar 15 15:37 386156.tar
$ gzip -9v 386156.tar
386156.tar:      74.1% -- replaced with 386156.tar.gz
$ ls -la 386156.tar.gz
-rw-r--r--  1 felicity fame 122206 Mar 15 15:37 386156.tar.gz
$ mv 386156.tar.gz /www/spamassassin/updates
$ vi /var/named/updates.spamassassin.org
[increase serial number, add in new update line '*.1.3 TXT "386156"']
$ rndc reload

```

- What if I want to publish updates for some versions but not others?

The answer is pretty simple and comes down to DNS records. As long as the reverse version DNS request does not return a TXT record, sa-update will consider there to be no updates available. ie: if we want to publish update 386156 for [SpamAssassin](#) 3.1.x, but not 3.1.0, we could use the following records:

```

0.1.3 A    127.0.0.1
*.1.3 TXT  "386156"

```

When v3.1.0's sa-update looks for an update, it gets no TXT response (having an existing 0.1.3 record overrides the wildcard record), and therefore it sees no updates available. However, when v3.1.1 or above looks for an update, it gets "386156" returned.

More Examples based on custom channel sa.zmi.at

Here is a short summary of what I do to provide the ZMI_GERMAN ruleset via it's channel sa.zmi.at. I found it hard to grab all the bits together, so this should help others to save some time installing their own channel.

```

# $version is the version taken from the ruleset, example "311"
cp $ruleset_tested $releasepath/70_zmi_german.cf
cd $releasepath
tar czf $version.tar.gz 70_zmi_german.cf
# create gpg signature. Needs to remove an existing .asc file first:
rm -f $version.tar.gz.asc 2>/dev/null
gpg --homedir $sa_channel/.gnupg -bas $version.tar.gz
# create the .sha1 sum:
sha1sum $version.tar.gz >$version.tar.gz.sha1
# publish everything on the web server:
rsync -qa $version.tar.gz* $websrv::sa-german-channel/

```

Now we auto-generate the DNS zone for sa.zmi.at, basically it contains this:

```
@      SOA      ns.zmi.at.      ns-admin.zmi.at. ( "
      $(date +%y%j)$version ; serial
[snip]
      A      212.69.164.60 ; IP address of webserver for sa.zmi.at
mirrors 86400 TXT      "http://sa.zmi.at/sa-update-german/MIRRORED.BY"
*.1.3   TXT      $version
*.2.3   TXT      $version
*.3.3   TXT      $version
```

HOWTO setup gpg

Setting up a new gpg key for a channel is simple if you know what to do. Here are the quick steps to save some time:

First create a new key in a new directory:

```
mkdir .gnupg-sa.zmi.at
gpg --homedir .gnupg-sa.zmi.at --gen-key
```

Now list that key so you know it's key-id, then export it to an ascii file. This is what your users have to import to sa-update later:

```
gpg --homedir .gnupg-sa.zmi.at --list-key
# our key id is 40F74481
gpg --homedir .gnupg-sa.zmi.at --armor --export 40F74481 >sa.zmi.at.asc
# upload our key to a keyserver
gpg --homedir .gnupg-sa.zmi.at --keyserver hkp://wwwkeys.pgp.net --send-keys 40F74481
```

In order to use the new channel, users need to do the following. I assume the .asc file has been downloaded from a website:

```
sa-update --import sa.zmi.at.asc
```

from now on, simply use

```
sa-update --gpgkey 40F74481 --channel sa.zmi.at
```

you can use it with "-D" for the first time, which gives a lot of debug output so you can see if everything went well.

The Backend

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

[TheoVanDinter](#) [MichaelMonnerie](#)