Release Process

The release process is in place to help follow the ASF guidelines for release management.

IMPORTANT: All source distributions must be built with autoconf v2.68 or later, and libtool v2.4 or later. Fedora Core 15 provides such a build environment for example.

Branches

All public releases are cut from a release branch off of master. With the migration to git, we now also sign all release tags.

Tags

For every released artifact, a tag should be created off the appropriate release branch. These tags should be named exactly the same as the version number released, e.g. "2.0.0-alpha" or "2.1.0". The release package should be made from the tag, not from the release branch. With the migration to git, all release tags should be signed with your release GPG key. For example:

```
$ git tag -s 3.1.2
```

You will be prompted for a commit message, and (probably) the gpg passphrase. If you need to specify a specific signing key, you can configure that in git (*signkey*) or specify it with the *-u* option to git tag.

CHANGES, STATUS, and docs/conf.py files

Before cutting a release, please make sure the CHANGES and STATUS files are up to date. In particular, make sure that the CHANGES file on trunk is in sync with the release branches (currently only 2.0.x). The CHANGES should include major feature changes or bug fixes for a release.

Versioning is controlled in the configure.ac file, it should be the only file necessary to be modified to change the release version number.

The documentation version is set by docs/conf.py. Make sure to set the new version on master and make sure the release version is correct.

RAT reports

The Release Audit Tool Tool reports are automatically generated on both trunk and all active release branches. The current reports are

- https://ci.trafficserver.apache.org/files/RAT/rat-master.txt RAT report
- 5.0.0 RAT report

It's up to the release manager to assure that there are no RAT issues unresolved before making a release.

Creating a release

After updating the versioning in configure.ac:

```
autoreconf -i && ./configure
gmake asf-dist
```

Creating a release with checksums and GPG signature

All artifacts must be accompanied with the appropriate GPG signature, and cryptographic checksums. To generate the artifact with the required checksums and signature, do

```
autoreconf -i && ./configure
gmake asf-dist-sign
```

This is the preferred (and recommended) way of creating a release candidate. The release artifact should always be accompanied with the checksums and signature. All release managers must update the KEYS file with their public key, as well as pushing their key to at least one GPG key server.

Voting

A new release candidate should be uploaded into a "personal" area, typically something like http://people.apache.org/~someuser/. Once available, a vote should be started on the dev@trafficserver.apache.org mailing list, and run for 72 hours. After the vote has passed, the artifact can be moved from the personal area to the distribution servers (see next section).

Uploading to distribution servers

All artifacts, and checksums and signatures, should be checked into the following Subversion directory:

```
https://dist.apache.org/repos/dist/release/trafficserver
```

Once committed, these files are automatically published to the appropriate areas on dist.apache.org.