

# Release Process

## Tika release process

### Before the release

Run `mvn ossindex:audit -Dossindex.fail=true` to see if there are any vulnerable dependencies.

Run `mvn versions:display-plugin-updates` and `mvn versions:display-dependency-updates` to identify any updates that need to be made.

Make sure to run full regression tests after making updates with `mvn -Prelease-profile clean verify`.

### Making a release

If you've done this before, have all programs installed, all necessary karma granted, and everything configured, steps 1 through 9 will take around an hour (tests take a while to run). If you've never done this before, you will probably run into an issue not covered in this set of instructions. So, it will take longer (maybe a couple hours, maybe a couple days).

Either way, please update these instructions with any clarifications you think might help the next release manager.

1. `git clone https://github.com/apache/tika.git` release OR `git clone https://github.com/apache/tika.git -b branch_1x`
2. Add to the `<scm>` section in tika-parent's pom

```
<connection>scm:git:https://github.com/apache/</connection>
<developerConnection>scm:git:https://github.com/apache/</developerConnection>
<url>https://github.com/apache/tika</url>
```

Background: for some reason in the 2.x branch these lines are deleted during each release process. They are not deleted in the 1.x branch.

3. Update CHANGES.txt with release date (Release X.Y.Z - MM/dd/yyyy (date format matters!) and (if needed) add additional changelog entries.
4. On the Tika JIRA, create versions X.Y.Z, X.(Y+1), and X.(Y+2), if not already done.
  - a. Move any lingering unresolved issues from X.Y.Z to X.(Y+1).
    - i. Query open issues: e.g. "project = Tika AND resolution = unresolved AND fixVersion = 1.16"
    - ii. Upper right, select bulk change
    - iii. Select all, modify to fixVersion = X.(Y+1)
    - iv. At the bottom of the screen, unselect the email updates
5. `git add CHANGES.txt` && `git commit -m "Update CHANGES.txt for X.Y.Z release."` && `git push origin`
6. Run `mvn apache-rat:check` and fix any missing license headers / add excluded for properly non-licensed fails
7. `setenv MAVEN_OPTS "-Xms128m -Xmx256m"` (or `export MAVEN_OPTS="-Xms128m -Xmx256m"` if you are using bash)
8. `mvn release:prepare`
  - a. Make sure all the release numbers are set to X.Y.Z (from X.Y.Z-SNAPSHOT)
  - b. When prompted, the scm tag name should be X.Y.Z (version number). If this is a release candidate, the tag should be X.Y.Z-rcN.
  - c. Watch for a BUILD SUCCESSFUL message
  - d. Make sure you have gnupg (or variant) installed (see <https://www.apache.org/dev/openpgp.html>)
9. `mvn release:perform`
  - a. Make sure you can log on to <https://repository.apache.org/>
  - b. Make sure you have the following entry in your `~/.m2/settings.xml` file: <https://maven.apache.org/settings.html#Servers>. The server ID should be `apache.releases.https`.
10. Head over to <https://repository.apache.org/> (Apache's Nexus Server)
  - a. Login with your ASF username and password
  - b. Make sure there is only one repo from your build and that it contains everything. A transmission failure can lead to multiple repositories, none of which are valid... even when `mvn release:perform` alleges SUCCESS
  - c. "Close" the staging repository with message Apache Tika X.Y.Z release candidate #N.
  - d. Copy the URL from the closed staged repository for Tika (you'll need this later)
11. Update CHANGES.txt with a "Release X.Y.Z + 1 - Current Development" section.
12. Check the contents of `target/checkout/target/X.Y.Z/*` into <https://dist.apache.org/repos/dist/dev/tika/> (tika-X.Y.Z-src.zip{.asc|.sha512}, tika-app-X.Y.Z.jar{.asc|.sha512} and tika-server-X.Y.Z.jar{.asc|.sha512}, and CHANGES.txt.
  - a. `svn co https://dist.apache.org/repos/dist/dev/tika/ dist.dev.tika`
  - b. `cd dist.dev.tika`
  - c. `mkdir X.Y.Z`
  - d. `rm -r X.Y.Z-1` (remove the former dist dev version folder)
  - e. `cp ../release/target/checkout/target/X.Y.Z/* X.Y.Z` (copy the artifacts into the new dist.dev.tika/X.Y.Z directory)
  - f. `cd X.Y.Z`
  - g. `mv CHANGES.txt CHANGES-X.Y.Z.txt`
  - h. make sure the KEYS file is at `dist.dev.tika/KEYS` -- if it isn't there already copy the KEYS file to go along with the jars and other release artifacts; add your key to KEYS if you haven't already
  - i. `svn add X.Y.Z`
  - j. `svn commit -m "Add Tika X.Y.Z RC#N artifacts."`
13. See the file in `target/checkout/target/vote.txt` - it contains the contents of the release email you should send (with subject [VOTE] Apache Tika X.Y.Z Release Candidate N to [dev@tika.apache.org](mailto:dev@tika.apache.org) and to [user@tika.apache.org](mailto:user@tika.apache.org)). Ensure all URLs are correct and update the email to include links to the artifacts from #8 (including .sha512, .asc and CHANGES.txt)
14. If VOTE fails:

- a. Drop the new Tika staging repository on <https://repository.apache.org>
- b. Make an entry in CHANGES.txt which documents the subsequent changes between release candidates. This is extremely useful if the candidates are cut from master and master has moved on since the release candidate was originally cut.
- c. Go back to #1.
15. If the VOTE passes:
  - a. Send [RESULT] [VOTE] tallying the VOTE to [dev@tika.apache.org](mailto:dev@tika.apache.org) and [user@tika.apache.org](mailto:user@tika.apache.org)
  - b. Head over to Nexus from #6 and "Release" the repository with message Apache Tika X.Y.Z release (and make sure the box is checked to auto drop).
  - c. `svn rm https://dist.apache.org/repos/dist/release/tika/X.Y.Z-1*` (delete the previous release)
  - d. `svn mv https://dist.apache.org/repos/dist/dev/tika/X.Y.Z/* https://dist.apache.org/repos/dist/release/tika/X.Y.Z/*`
  - e. `git clone https://github.com/apache/tika.git X.Y.Z-rcN`
  - f. Add final tag for the release and delete the X.Y.Z-rcN tag:
    - i. Look up commit for X.Y.Z-rcN commit: `git rev-list -n 1 1.25-rc2`; let's say '0090ebac8e4ff4083a9c0c5d3dc55f545ad6f951'
    - ii. `git tag -a X.Y.Z 0090eba -m "Tagging X.Y.Z release"`
    - iii. `git tag -d X.Y.Z-rcN`
    - iv. `git push --tags`
    - v. `git push origin :refs/tags/X.Y.Z-rcN`
16. Update the version of any unreleased modules (like tika-dotnet).
17. Update Tika site.
  - a. Update parent version number in the site pom.xml file.
  - b. Update the Documentation section of src/site/site.xml. Make sure to add the collapse tag for the now old version.
  - c. Update the "Full List of Supported Formats" section of the formats page of version X.Y.Z using Tika-411; e.g. `java -jar release-X.Y.Z/tika-app/target/tika-app-1.15.jar --list-parser-details-apt >> src/site/apt/1.15/formats.apt`
  - d. Update the version towards the top of X.Y.Z/formats.apt: `src/site/apt/1.15/formats.apt`
  - e. Update the version number in the dependency examples and the commandline options in the new X.Y.Z src/site/apt/X.Y.Z/gettingstarted.apt page.
  - f. Update src/site/resources/doap.rdf with the new release.
  - g. Create a new example and formats page (truncated to where step 15.3 should go next time) for the next version (but don't list it in the sidebar).
  - h. Update src/site/apt/index.apt.vm to include the announcement and change link to previous release CHANGES.txt file (from [https://dist.apache.org/repos/dist/release/tika/X.Y.Z/CHANGES-X.\(Y-1\).txt](https://dist.apache.org/repos/dist/release/tika/X.Y.Z/CHANGES-X.(Y-1).txt) to [https://archive.apache.org/dist/tika/X.\(Y-1\)/CHANGES-X.\(Y-1\).txt](https://archive.apache.org/dist/tika/X.(Y-1)/CHANGES-X.(Y-1).txt)).
  - i. Update src/site/apt/X.Y.Z/index.apt.
    - i. Generate the contributor list with <https://github.com/chrismattmann/apachestuff/blob/master/extract-tika-contribs> (I use wget and then apply the pipe logic)
    - ii. Update the link to the JIRA query (e.g. <https://s.apache.org/XowY>) via Apache's shortening service <https://s.apache.org>
    - iii. Generate the issues list by running <https://github.com/chrismattmann/apachestuff/blob/master/extract-tika-issues.py> CHANGES-X.Y.Z.txt output.txt X.Y.Z (I have local mods to allow this to work with python3. I should open a PR...)
  - j. Generate the Javadoc.
    - i. `unzip tika-X.Y.Z-src.zip`
    - ii. `mvn javadoc:aggregate`
    - iii. `mkdir ../tika-site/publish/X.Y.Z`
    - iv. `mv target/site/apidocs ../tika-site/publish/X.Y.Z/api`
  - k. Update tika-server documentation (We haven't had a license for a good while. This is no longer active):
    - i. In the unzipped src release, cd tika-server
    - ii. `mvn -Pserver install` (in the tika-server module to build the miredot documentation)
    - iii. `mkdir ../tika-site/src/site/resources/X.Y.Z`
    - iv. `rm -r tika-site/src/site/resources/X.Y.Z/miredot` (if it exists?!)
    - v. `cp -r tika-server/target/miredot ../tika-site/src/site/resources/X.Y.Z`
  - l. Update the downloads section (Unless you're supporting two major versions at a time, this should be automatic).
  - m. Check everything looks good with `mvn site:run`.
  - n. `mvn clean install`
  - o. `svn add <any files changed for site>` (probably: `src/site/apt/X.Y.Z` and `src/site/resources/X.Y.Z/miredot`)  
`&& svn commit -m "Update website for X.Y.Z release."`
  - p. On the Tika JIRA, "release" version X.Y.Z and update any (new) straggler X.Y.Z issues to X.(Y+1).
18. Send announcements to [user@tika.apache.org](mailto:user@tika.apache.org), [dev@tika.apache.org](mailto:dev@tika.apache.org), and [announce@apache.org](mailto:announce@apache.org) lists (from your Apache email) - see <https://s.apache.org/ytaod> for a sample. If editing and resending an earlier email, make sure that you use text format and/or make sure that editing the links updates the underlying links and not just the anchor text.
19. Log on to <https://reporter.apache.org/addrelease.html?tika> and add the release data (version and date) to the database

**NOTE:** if anything goes wrong during `mvn:release-prepare` or `mvn release:perform`:

1. `mvn release:clean`
2. Undo any commits that changed the version number
3. Delete the X.Y.Z-rcN tag: `git push --delete origin X.Y.Z-rcN && git tag --delete X.Y.Z-rcN`

After the release, consider making a PR to make the upgrade in Apache Solr. See [UpgradingTikaInSolr](#).

After the release, announce any CVEs that were fixed and update [security.apt](#) and republish the site.

## After the project release

## Release the Docker Image

[Release Process for tika-docker](#)

## Release the Helm Chart

[Release Process for tika-helm](#)