How to Release

This document describes how to release Sentry. It is a work in progress and should be refined by the Release Manager (RM) as they come across aspects of the release process not yet documented here.

NOTE: This document outlines how to do a source only release. Releasing and publishing binary artifacts requires steps that are not outlined here.

NOTE: For the purpose of illustration, this document assumes that the version being released is 1.8.0, and the following development version will become 2.0.0

Prerequisites

Policy documents

Frequently asked questions for making Apache releases are available on Releases FAQ page. Release Process guide also has good information on best practices for releasing artifacts from an Apache project.

The Release Manager (RM) must go through the policy document to understand all the tasks and responsibilities of running a release.

Formal DISCUSS thread

We need to send out an email to dev@ proposing a release and reference this thread in the heads up email(see below)

Naming conventions:

Release name: (Major.Minor.Patch)

Branch name: (Major.Minor)

Tag-name: (Major.Minor.Patch)

Give a heads up

The RM should first create an umbrella issue and then setup a timeline for release branch point. The time for the day the umbrella issue is created to the release branch point must be at least two weeks in order to give the community a chance to prioritize and commit any last minute features and issues they would like to see in the upcoming release.

The RM should then send the pointer to the umbrella issue along with the tentative timeline for branch point to the developer lists (TODO: Also send to users list once we create one). Any work identified as release related that needs to be completed should be added as a subtask of the umbrella issue to allow users to see the overall release progress in one place.

```
To: dev@sentry.apache.org
Subject: Work on release X.Y.Z has started

Created a tracking jira (jira#) for the Sentry release X.Y.Z and plan to branch tentatively on $date.

What would you like to see included?
Thanks,
$RM
```

Sanitize JIRA

1. Before a release is done, make sure that any issues that are *fixed* have their fix version setup correctly. Run the following JIRA query to see which resolved issues do not have their fix version set up correctly:

   ```
   project = sentry and resolution = fixed and fixVersion is empty
   ```

   The result of the above query should be empty. If some issues do show up in this query that have been fixed since the last release, please bulk-edit them to set the fix version to '1.8.0'.
2. Move the unresolved jiras to the next release

   ```
   project = sentry and fixVersion = X.Y.Z and status not in( resolved, done, Accepted, Closed)
   ```

3. You can also run the following query to make sure that the issues fixed for the to-be-released version look accurate:

   ```
   project = sentry and fixVersion = 'X.Y.Z'
   ```

4. Finally, check out the output of the **JIRA release note tool** to see which JIRAs are included in the release, in order to do a sanity check.

Monitor active issues

It is important that between the time the intent to release email is sent and the release branch is created, no experimental or potentially destabilizing work is checked into the trunk. While it is acceptable to introduce major changes, they must be thoroughly reviewed and have good test coverage to ensure that the release branch does not start of being unstable.

If necessary the RM can discuss if certain issues should be fixed on the trunk in this time, and if so what is the gating criteria for accepting them.

Creating Release Artifacts

Communicate with the community

1. Send an email to dev list to
   Notify that you are about to branch. Ask to hold off any commits until this is finished.
2. Send another email after branching is done.

Preparing branches

1. Create a release branch
   a. Clone fresh repository copy

      ```
      git clone https://git-wip-us.apache.org/repos/asf/sentry.git
      cd sentry
      ```

   b. Checkout master branch

      ```
      git checkout master
      ```

c. Check that current HEAD points to commit on which you want to base new release branch. Checkout particular commit if not.

    ```
    git log # Check current branch history. HEAD should point to commit that you want to be base for
    your release branch
    git checkout abcdef123 # Check out particular commit that should be base for release branch if -^
    ```

d. Create new release branch with name "branch-$version"

    ```
    git checkout -b branch-X.Y.Z
    ```

e. Update CHANGELOG in the trunk to indicate the changes going into the new version.

    The change list can be swiped from the **JIRA release note tool** (use the "text" format for the change log). See JIRA Cleanup above to ensure that the release notes generated by this tool are what you are expecting.

f. RM should update the year of the file "NOTICE.txt".

g. Remove -SNAPSHOT from the release branch and commit

    ```
    mvn versions:set -DnewVersion=X.Y.Z -DgenerateBackupPoms=false
    git add .
    git commit -m "SENTRY-XXXX: Removing -SNAPSHOT from X.Y release branch"
    ```

h. Check your changes and push new branch to Apache repository. "repository name" is the name of the repository, for example "origin".

    ```
    mvn -U clean package -Pdownload-hadoop -Pdist -s <settings.xml>
    git push <repository name> origin branch-X.Y.Z
    ```

i. Check that branch was correctly propagated to Apache repository.

2. Prepare the master for next release (TODO: update change log?)

    ```
    git checkout master
    mvn versions:set -DnewVersion=<X.Y.Z+1>-SNAPSHOT -DgenerateBackupPoms=false
    git add .
    git commit -m "SENTRY-XXXX: Preparing for sentry <Release name> development"
    git push origin master:master
    ```

3. Send an email announcing new branch

    ```
    To: dev@sentry.apache.org
    Subject: New sentry release branch-X.Y


    I've just created new release branch for upcoming <Release name> release.
    Please continue committing to master branch as usual.
    I'll cherry-pick commits to branch-X.Y on per needed basis.


    Thanks,
    $RM
    ```

Create release tar balls

1. Check out release branch

    ```
    git checkout branch-X.Y
    ```

2. Create tag on this commit to identify precise point where the RC was generated and push this tag to main repository

```
git tag -a release-X.Y.Z -m "Sentry X.Y.Z release"
#Make sure compiles/tests run fine and rat check is fine
mvn clean install -DskipTests -Pdist
mvn test
mvn verify -DskipTests (to do the rat check)
git push origin release-X.Y.Z
```

3. If an rc1, rc2, etc is needed, delete that tag before creating a new one:

```
git tag -d release-X.Y.Z
git push origin :refs/tags/release-X.Y.Z
```

4. Create temporary directory where you'll be preparing all required artifacts

```
mkdir -p /tmp/sentry-release-preparations
```

5. Create source artifact and move it to your temporary directory (TODO: git verify?)

```
git archive --format=tar --prefix=apache-sentry-X.Y.Z-src/ HEAD | gzip > /tmp/sentry-release-preparations
/apache-sentry-X.Y.Z-src.tar.gz
```

Sanity Check

1. Make sure the tar and the rc match

```
cd /tmp/sentry-release-preparations
tar -xvf apache-sentry-X.Y.Z-src.tar.gz

#Do a fresh clone of the tag
git clone https://git-wip-us.apache.org/repos/asf/sentry.git
cd sentry/
git checkout tags/release-X.Y.Z
cd ..
diff -r sentry apache-sentry-X.Y.Z-src
```

2. Make sure code compiles and tests pass on the untared src.

```
cd apache-sentry-X.Y.Z-src
mvn clean install -DskipTests -Pdist
mvn test
mvn verify -DskipTests (to do the rat check)
```

Create signatures and check sums (TODO: Use maven gpg plugin?)

All artifacts must be signed and checksummed. In order to sign a release you will need a PGP key. You should get your key signed by a few other people. You will also need to recv their keys from a public key server. See the Apache release signing page for more details. If you add your PGP key fingerprint to your Apache profile, your key should automatically be added to https://people.apache.org/keys/group/sentry.asc

1. Change your working directory to the temporal one

```
cd /tmp/sentry-release-preparations
```

2. Sing each file with your key

```
 for file in *.tar.gz; do gpg --armor --output $file.asc --detach-sig $file; done
```

3. You can immediately verify your signature

```
for file in *.tar.gz; do gpg --verify $file.asc $file; done
```

4. Create sha512 check sum as sha1 is no longer safe

```
for file in *.tar.gz; do sha512sum $file > $file.sha512; done
```

Upload artifacts

Upload artifacts and all created check sums with signatures to https://dist.apache.org/repos/dist/dev/sentry

1. Install svn client "yum install subversion" more details in https://tecadmin.net/install-subversion-1-8-on-centos-rhel/

2. Checkout the content from "https://dist.apache.org/repos/dist/dev/sentry/". Then create new branch directory, add binary files and source files. Then commit them to repository server.

```
svn checkout https://dist.apache.org/repos/dist/dev/sentry/ <local path>
svn mkdir X.Y.Z
svn add <file_name>
svn commit -m 'commit message'
```

Update KEYS file

Only PMC can change the KEYS file.

If your PGP key is not yet in the project's KEYS file, you need to first add that in. To do this, checkout the KEYS file and update it using the following commands:

```
$ svn co https://dist.apache.org/repos/dist/release/sentry sentry-release
$ cd sentry-release
$ (gpg --list-sigs <KEY-ID> && gpg --armor --export <KEY-ID> ) >> KEYS
$ svn commit -m "Adding PGP public key to KEYS file" KEYS
```

Once this file has been updated, you need to publish it in the appropriate dist directory for the project on `http://www.apache.org/dist`. To do this, you must copy the file as follows:

```
$ scp KEYS people.apache.org:/www/www.apache.org/dist/sentry/KEYS
```

This will take some time to propagate in which you can continue with the other steps of the release process.

Publishing Apache Sentry artifacts to maven central

1. Checkout the relevant git tag, e.g. git checkout release-1.8.0

2. Make sure you have an entry in your ~/.m2/settings.xml with id "apache.staging.https" with your Apache username + password, e.g.:

```
<settings>
  <servers>
    <server>
      <id>apache.releases.https</id>
      <username><apache-username></username>
        <password><apache-password></password>
    </server>
    <server>
      <id>apache.staging.https</id>
      <username><apache-username></username>
        <password><apache-password></password>
    </server>
  </servers>
</settings>
```

3. Deploy the release with: **mvn clean deploy -Psign-artifact**s

4. Go to https://repository.apache.org/ and **log in** using your Apache username + password.

5. Click on "Staging Repositories" on the left hand side.

6. Select the entry that starts with orgapachesentry and click on "close". If "close" fails, repeat Step 3-5.

7. Verify via the URL that should appear after refresh that the artifacts look as expected.

8. Run vote. The detail on running vote is in section "**Running the vote**"

9. After approval, click on "release".

Verifying a release candidate

Following are the typical things we need to verify before voting on a release candidate. And the release manager should verify them too before calling out a vote.

- Make sure RCs are hosted @ https://dist.apache.org/repos/dist/dev/sentry
- Should be in format apache-$project-$version.tar.gz
- Verify Signatures and hashes. You may have to import the public key of the release manager to verify the signatures. (gpg --recv-key <last8 of public key>)
- git tag matches the released bits (diff -rf)
- Can compile successfully from source
- Verify NOTICE.txt file has right content (current year etc)
- All files have correct headers (Rat check should be clean - mvn verify)
  - Make sure there are no conflicting licenses (TODO: how)
- No jar files or the like in the release

Running the vote

Voting has to be done on dev@sentry.apache.org You can close the vote after voting period expires and you accumulate sufficient votes.

Call for voting on dev list (PMC)

1. Template of the email to call for voting

a. To find the link of fixed issues, go to "https://issues.apache.org/jira/projects/SENTRY/versions", and click the version to be released.

b. The "orgapachesentry-<id>" is from Step 6 in Section "**Publishing Apache Sentry artifacts to maven central**"

```
To: dev@sentry.apache.org
Subject: [VOTE] Release Sentry version X.Y.Z
Content:

This is the release of Apache Sentry, version X.Y.Z.

It fixes the following issues: https://issues.apache.org/jira/projects/SENTRY/versions/12342213

Source files : https://dist.apache.org/repos/dist/dev/sentry/X.Y.Z


Maven artifacts are available: https://repository.apache.org/content/repositories/orgapachesentry-<id>

Tag to be voted on (rc#): https://git-wip-us.apache.org/repos/asf/sentry/?p=sentry.git;a=commit;h=<commit-hash-of-the-tag>
Sentry's KEYS containing the PGP key we used to sign the release:
https://people.apache.org/keys/group/sentry.asc (or http://www.apache.org/dist/sentry/KEYS for non committers)

Note that this is a source only release and we are voting on the source: tag=release-X.Y.Z, SHA=<Checksum of the last commit in the branch> (You can get the hash of the tag by doing "git rev-list release-X.Y.Z | head -n 1" )

Vote will be open for 72 hours.

[ ] +1 approve
[ ] +0 no opinion
[ ] -1 disapprove (and reason why)


Thanks,
$RM
```

The vote has to be called on the dev list. Upon receiving 3 +1s from the PMC, reply to the voting thread and prefixing **[RESULT]** to the subject line with the results.

Example close email:

```
To: dev@sentry.apache.org
Subject: [RESULT][VOTE] Release Sentry version X.Y.Z
Voting is now closed and has passed with the following tally,

Binding +1s: Patrick Hunt, Arvind Prabhakar, Andrei Savu
Non binding +1s: Xuefu Zhang, Jarcec Cecho, Ashish Paliwal.

Thanks to everyone who voted! I'll continue with the rest of the release process.

$RM
```

Rolling out the Release

Close JIRA version

You need to close the release in JIRA so that everyone knows that your version should not be used as "fixVersion" for new bugs. You must be a Jira administrator or a project administrator to do the Jira Software tasks on this page. Sentry PM should have this access.

Go to JIRA "Administer project" page and follow "Versions" in left menu. Table with list of all releases should appear, click on additional menu on the right of your release and choose "Release" option. Submit release date and you're done.

Upload the artifacts

In order to release you have to checkout release repository located on https://dist.apache.org/repos/dist/release/sentry/ and add release artifacts there. Only Sentry PMC can do this.

```
svn co https://dist.apache.org/repos/dist/release/sentry/ sentry-release
cd sentry-release
mkdir X.Y.Z/
cp $source_to_your_artifacts X.Y.Z/
svn add X.Y.Z

svn commit -m "Add $source release"
```

It may take up to 24 hours for all mirrors to sync up (http://www.apache.org/dyn/closer.cgi/sentry/)

Update the website

1. Prepare to edit the website.

   ```
   svn co https://svn.apache.org/repos/asf/sentry/
   ```

   a. Add the release to the sentry/site/trunk/content/general/downloads.mdtext
   b. Add the release to the sentry/site/trunk/content/general/history.mdtext
   c. Commit the changes
   d. Go to https://cms.apache.org/sentry/ get the sentry working copy, force a new copy if you don't see your changes yet, and then ask it to publish the new copy. That should update the downloads and history page as per the changes you made to downloads.mdtext, history.mdtext earlier. If you don't see it yet, you may have to wait 24 hours - it should be there by then.
2. Update the blog with new features.

Announce the release

Send an email to announce@apache.org (the from: address must be @apache.org). For example,

```
To: announce@apache.org, dev@sentry.apache.org
Subject: [ANNOUNCE] Apache Sentry X.Y.Z released

The Apache Sentry team is happy to announce the release of version X.Y.Z.
Apache Sentry is a system to enforce fine grained role based authorization
to data and metadata stored on a Hadoop cluster.

The release bits are available at:
http://sentry.apache.org/general/downloads.html

Sentry X.Y.Z Release Notes are available here:

https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12314720&version=12343306


We would like to thank all contributors who made the release possible.

Regards,
Sentry team
```

Preparing Branch for Future Maintenance Release

After the release has been completed, prepare the branch for the next development cycle.

1. Check out the release branch with:

   ```
   git clone https://git-wip-us.apache.org/repos/asf/sentry.git

   cd sentry

   git checkout branch-X.Y
   ```

2. Increment the `version` property value in all pom.xml files and add the `SNAPSHOT` suffix. For example, if the released version was `0.7.0`, the new value should be `0.7.1-SNAPSHOT`. Please note that the `SNAPSHOT` suffix is required in order to indicate that this is an unreleased development branch. Use Maven's Versions plugin to do this as follows:

   ```
   mvn versions:set -DnewVersion=0.7.1-SNAPSHOT -DgenerateBackupPoms=false
   ```

3. Verify that the build is working with changes.
4. Commit these changes with a comment "Preparing for X.Y.Z+1 development".