SOFTWARE CRAFT
(/)
About programming and software craftsmanship.

Signing Git Tags

March 14th, 2013

One of Git’s interesting features is definitely the ability to sign tags. Using your GPG key-pair you can create a signed tag. Anyone with your public key can verify that you are the true creator of that tag.

Why you should sign your tags

As Git is a distributed version control system (DVCS) you need a to verify the source of a release. Just like you verify checksums of releases you download to make sure it’s the release you expect it to be (you do verify your downloads, don’t you?!), you want to prove that the tag is created by the expected trusted source.

While signing tags is more important in open-source projects, it’s generally also a good idea for your private projects. See the signing as one of the core team members placing stamp in your repository, as they release a version of their software.

Also signing tags is dead easy as I’ll show you next.

Create a signed tag

To create a signed tag all you need is a Git repository and a GPG private key you want to use to sign the tag. Don’t worry if you don’t have GPG set up yet, I’ll cover this a little later.

To make things really easy set your GPG signing key as a global configuration setting. Obviously you don’t need to do this - especially if you have more than one account (e.g. work and private) you want multiple signing keys so set them up per repository (discard the global option).

```
$ git config --global user.signingkey [gpg-key-id]
```

Now all you need to do is create a tag and sign it.

```
$ git tag --sign [signed-tag-name]
```

SUBSCRIBE

By Email (https://feedburner.google.com/fb/a/mailverify?uri=softwarecraft&loc=en_US)

Atom feed (http://feeds.feedburner.com/softwarecraft)

RECENT POSTS

- Global Day of Coderetreat 2014 (/global-day-of-coderetreat-2014.html)
- Moved to Github Pages (/moved-to-github-pages.html)
- Java's Volatile Modifier (/2014/07/javas-volatile-modifier.html)
- Book Review: Understanding the 4 Rules of Simple Design (/2014/04/book-review-understanding-4-rules-of.html)
- Method References in Ruby (/2014/04/method-references-in-ruby.html)
- Is your Scrum team Agile? (/2014/04/is-your-scrum-team-agile.html)
- Lineman.js (/2014/03/linemanjs.html)
- Fast Remote Service Tests (/2014/02/fast-remote-service-tests.html)
- Code Katas (/2014/02/code-katas.html)
- Acceptance Test Driven Development (/2014/01/acceptance-test-driven-development.html)

Verify a tag

Verifying tags is as easy as creating one.

```
$ git tag --verify [signed-tag-name]
```

How ever, you need the signer's public key to be able to verify the commit. Most maintainers put their public key in a special object in the repository - an object that can only be reached from a specific tag (or of course its ow n SHA).

You can make your public key available in a tag as follow s.

```
$ gpg -a --export [gpg-key-id] | git hash-object -w --stdin
[object SHA]

$ git tag -a [object SHA] maintainer-pgp-pub
```

To import a maintainer's public key into your keyring you can show the tag.

```
$ git show maintainer-pgp-pub | gpg --import
```

Generally it is also a good idea to leave instructions on importing your public key and verifying tags in the tag message.

Create a GPG key

If you don't have a GPG key pair yet it's about time to create one. If you're in doubt most likely you don't have any but it's alw ays a good idea to check your list of GPG keys.

```
$ gpg --list-keys
```

If you need to, generate a key. Choose the RSA algorithm if you can as DSA only supports up to 1024 bits.

```
$ gpg --gen-key
```

As you see using GPG and signing Git tags is really simple, so there's actually no reason not to sign your tags.

So go and sign and verify!

Comments are disabled. Please email me (mailto:bart@thesoftwarecraft.com) instead.

ABOUT THE AUTHOR

(/about/)
Bart is a technologist who specialises in agile software development. He is passionate about creating working software that is easy to change and maintain. (/about/)

DISCLAIMER

Some of the links contained within this site have my referral id (e.g. Amazon), which provides me with a small commission for each sale. Thank you for your support.