



# SIGNING YOUR COMMITS TO GITHUB USING GPG

By Matt Langlois



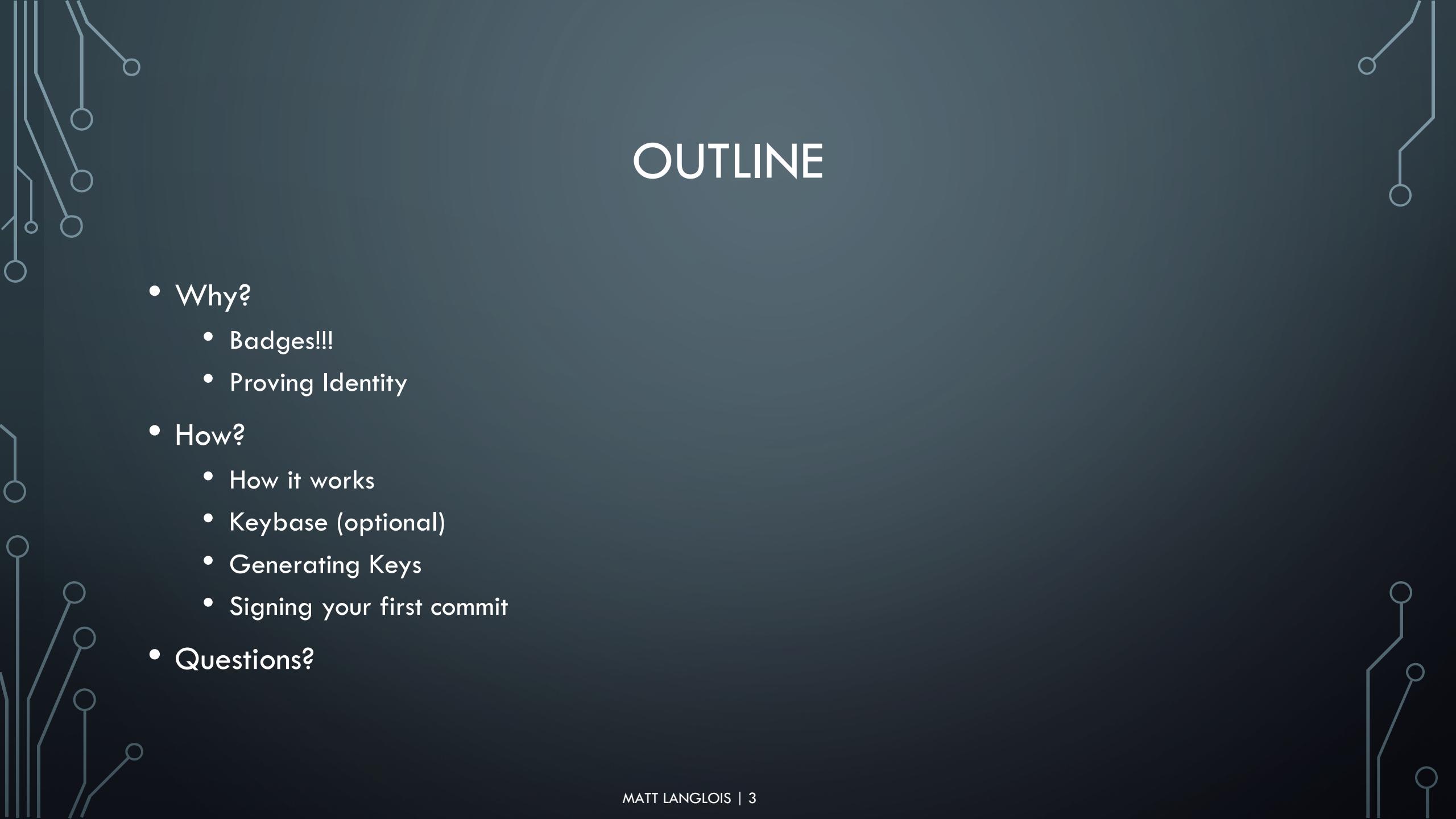
WHOAMI

&

@



uOttawa

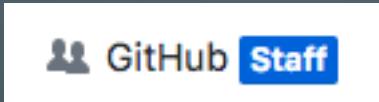


# OUTLINE

- Why?
  - Badges!!!
  - Proving Identity
- How?
  - How it works
  - Keybase (optional)
  - Generating Keys
  - Signing your first commit
- Questions?

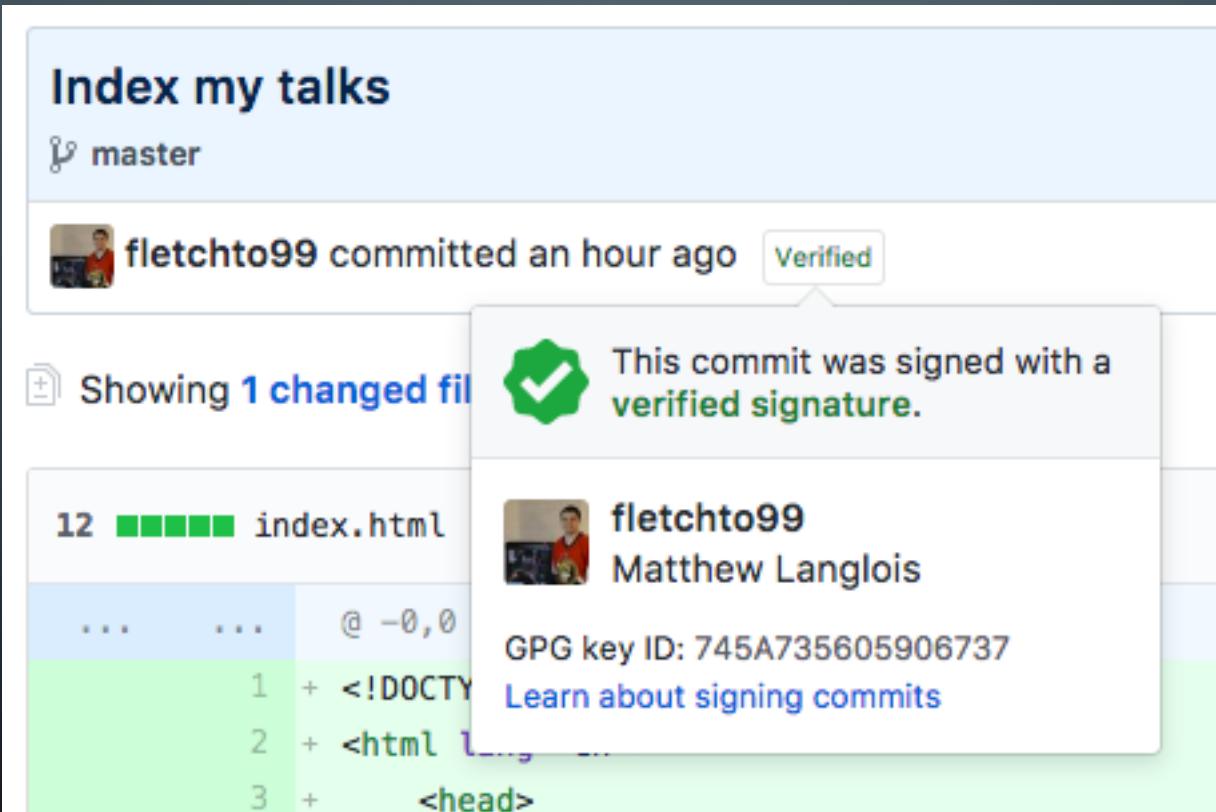
# BADGES!!!

Who doesn't love this?!?!



# BADGES!!!

Then you can also have these!!!



# PROVE YOUR IDENTITY

The screenshot shows a GitHub repository page for `fletchto99 / CommitSigningDemo`. The repository has 0 stars and 0 forks. The main navigation bar includes links for Code, Issues (0), Pull requests (0), Wiki, Releases (0), Settings, and More.

A prominent message in the center says **This is why you should sign your commit**. Below it, a commit from user `defunkt` is shown, committed 20 seconds ago. The commit details are: 1 parent `ff6fc52`, commit `06d4c0d61e9a5a08e1a8267381416b4f9041cec8`, patch diff.

The commit message content is displayed in a diff viewer:

```
diff --git a/Readme.md b/Readme.md
index ff6fc52..06d4c0d 100644
--- a/Readme.md
+++ b/Readme.md
@@ -1 +1 @@
- This is why signing commits is important!
+ This is why signing commits is important! - Written & committed by fletchto99
```

At the bottom right of the diff viewer, there are buttons for No Whitespace, Unified, and Split.

# HOW IT WORKS

- You set an email address in your gitconfig
- You set an email address on github & verify it
- You generate a GPG key locally with the verified email on github
- You put the public key on github
- You sign your commit using your private key
- Github validates the signature using your public key & the commits email address

# KEYBASE (OPTIONAL – RECOMMENDED)

- Register at <https://Keybase.io>
- A place to store your proofs of identity online
- Keybase has a CLI through homebrew `brew cask install keybase`
- You'll also need GPG – `brew install gpg`

# GENERATING KEYS

- First run `Keybase login`
- Next run `Keybase pgp gen --multi`
- Make sure the email entered matched a verified email address on your github account
- (Optional) Upload encrypted private key to keybase

# USING YOUR NEW KEY

- Use `gpg --list-secret-keys` to list the IDs of your keys

```
/Users/mattlanglois/.gnupg/pubring.kbx
-----
sec    rsa4096 2017-06-20 [SC] [expires: 2033-06-16]
      5595ADD45E88CF8E26C89090745A735605906737
uid          [ unknown] Matthew Langlois <tletchto99@gmail.com>
ssb    rsa4096 2017-06-20 [E] [expires: 2033-06-16]
```

- The output is the ID of your key which will be used to sign things

# USING YOUR NEW KEY - CONT

- Run `git config --global user.signingkey <secret key id>`
- Run `git config --global commit.gpgsign true`

# FINAL STEPS

- Navigate to <https://github.com/settings/keys>
- Press 'New GPG key'
- In your terminal run `keybase pgp export -q <key id>`
- Copy everything that is output



# GO COMMIT & PUSH SOMETHING!!

- You are now a verified committer 😊



# QUESTIONS/LIVE DEMO?

# KEEP IN TOUCH!

- Github: <https://github.com/fletchto99>
- Twitter: [@fletchto99](https://twitter.com/fletchto99)
- Further reading:
- Fun random website: <https://pi.fletchto99.com>