Encrypting data with GPG

Tip: stay focused

- The background for this topic is HUGE,
  - While the objectives for this topic are tiny.

- I have a video for all foundations on Youtube.
  - *Introduction to cryptography, PKI and certificates.*

Tip: stay focused

- For LPIC1, all you need to know is:
  - Generating and securing a new key pair.
  - Sharing and importing public keys.
  - Encrypting and decrypting data.
  - Signing and verifying data.

Generating a key pair

- You need ample "entropy", a source of randomness.

- *gpg --gen-key*
  - This makes new keys in *~/.gnupg/keyring*.
  - Use unique email addresses!

See: Understanding random number generation

Inventory of key pairs

- You can see which keys are available to you:
  - *gpg --list-keys*

- Type "*[unknown]*" indicates an imported public key.
- "*ls -al ~/.gnupg*" does not show individual keys.

Sharing public keys

- Exporting your & importing someone's public key:

```
$ gpg --export tess@fedora.local > ./fedora.pub

$ gpg --import /tmp/ubuntu.pub
```

Encrypting data

- First make a message, like *secret.txt*.
- Then:

```
$ gpg --out ./secret.gpg \
--recipient tess@fedora.local \
--encrypt ./secret.txt                  # 👈🏼 must be last!
```

Decrypting data

- Transfer the *secret.gpg* to your recipient.
- They can decrypt it:

```
$ gpg --decrypt /tmp/secret.gpg
```

Signing / verifying

- Signing does not require encryption.

```
$ gpg --out contract.gpg \
--local-user tess@ubuntu.nl \
--sign contract.txt                # 👈🏼 must be last!

$ gpg --verify contract.gpg
```

LAB: Using GPG

Assignment

- On both your VMs, create a GPG key pair.
  - One for *yourname@fedora.local*
  - One for *yourname@ubuntu.local*

- Import the public keys across the systems.
  - Fedora -> Ubuntu, and Ubuntu -> Fedora.

Assignment

- On Fedora:
  - Create a message file "*secret.txt*".
  - Encrypt this file to send it to your Ubuntu user.
  - Sign the encrypted message.
  - Copy the encrypted and signed file to Ubuntu.

Assignment

- On Ubuntu:
  - Verify the signature on the file.
  - Decrypt the received file.

- Verify: are the contents correct?