Todays Agenda (2/1)

- Key signing party review (finish this document)
- 3 GRADED Assignments for today
  - Mkijowski signed public key (pilot dropbox)
  - YOUR thrice-signed public key (pilot dropbox)
  - Pilot Quiz (you need your public key fingerprint)

It's going to be a lame party isn't it...

You all should have run the following:

```
gpg --full-generate-key
```

- Use your real name and campus email!
- Use all the bits (4096)!
- Feel free to set an expiration of 16w (weeks) if you plan on throwing this away ***AT THE END OF THIS SEMESTER***
- Use a password that you will not forget!
- We will use this key again so make sure you save it!

Goals for today

- Establish trust with instructor (and 3 other students in this class)
- Import my public key
- Sign and Return my public key
- Exchange and sign 3 other keys

Import my public key

- Attempt 1: `gpg --recv-keys E47763416159625F60ACE88A7E5CF54E1BBA3984`
- Attempt 2: `gpg --keyserver keyserver.ubuntu.com --recv-keys E47763416159625F60ACE88A7E5CF54E1BBA3984`

OR

Copy and paste the contents (from `kijowski-gpg/`) into a file in linux and import it with:

- `gpg --import kijowski.gpg`
- OR: `gpg --import kijowski.gpg.pub.txt`

Check fingerprint, sign my key

```
E477 6341 6159 625F 60AC  E88A 7E5C F54E 1BBA 3984 <- Look
                                7E5C F54E 1BBA 3984 <- This
```

- `gpg --edit-key matthew.kijowski@wright.edu`
- `sign`
- `save`
- `gpg --armor --export matthew.kijowski@wright.edu`

Return this key to me via pilot dropbox AS A PLAIN TEXT FILE!

Lets refresh hashing and Asymmetric encryption

A hash gaurantees integrity (message has not changed).

Public key can decrypt messages from private. (Authenticity)

Private key can decrypt messages from public. (Confidentiality)

Lets party!

- Convince your table mates that you are the person with the given email and share your fingerprint!
- Exchange public keys (send/recv or just export and email/discord)
  - `gpg --armor --export YOUR.email@wright.edu`
- Import `gpg --import their-filename` OR `gpg --recv-keys`
- Sign `gpg --edit-key THEIR.email@wright.edu` > `sign` > `save`
- Return signed key to them reverse of above exchange
  - `gpg --armor --export THEIR.email@wright.edu`
- Import your new signatures `gpg --import filename`

Among your table (and Discord)

- Download/copy/paste each person's public key from discord
- Make a file in your home directory for each key
- Files should start with `-----BEGIN PGP PUBLIC KEY BLOCK-----`
- Import with the following

```
gpg import <filename>
```

Day 2 (what are we even doing?)

- Exchanging keys
- Establishing a Web of Trust!

But what else can we be doing...

- `gpg --list-sigs <Email or Fingerprint>`

Signed sealed delivered

Lets sign a message!

- Create a `sample.txt` file with a public message (Hello World or some such thing).
- `gpg --sign sample.txt`
- share the output sample.gpg with someone you exchanged keys with
- `cat` the output file, can you read the contents?
- `gpg --verify sample.gpg`

Other signing options

- `gpg --clear-sign`
- `gpg --detach-sign`

Now for some fun

Lets send a secret message!

- Create a text file `secret-message.txt`
- Choose someone you have exchanged keys with
- Encrypt the file: `gpg --output secret-message.gpg --encrypt --recipient their.email@wright.edu`
- Send them `secret-message.txt` via email or discord
- The recipient can decrypt with:
  - `gpg --output secret.txt --decrypt secret-message.gpg`

Back up your gpg keys!!!

```
tar -cpzf gnupg.tar.gz ~/.gnupg/
```

if you are using a Wright State laptop

```
cp gnupg.tar.gz /mnt/c/Users/student/Desktop/
```

Save this file!!!

You can also backup your private key and any public keys with:

- `gpg --armor --export-secret-key your.email@wright.edu`
- `gpg --armor --export your.email@wright.edu`
- `gpg --armor --export friends@wright.edu`

(Web of Trust)[https://en.wikipedia.org/wiki/Web_of_trust]

Decentralized trust network similar to what we have done in class.

- Hard to attack
- easy to lose private key
- hard to get started as a new user (how do you gain trust?)

Wouldn't it be nice

- Lets pretend we are lazy...
- How would you improve the Web of Trust?

I really hope we invented a Certificate Authority. . .

Raise your hand if you know the following companies

- Thawte
- Comodo
- WoSign
- Symantec
- GoDaddy

(Timeline of Certificat Authority Fail-
ures)[https://sslmate.com/resources/certificate_authority_failures]

PKI

The methods, policies, roles, hardware, software, and procedures that facilitate creating, managing, distributing, using, storing, and revoking public keys is called a Public Key Infrastructure (PKI).