Author: Martin Rasmussen 4th Semester Securtiy 2016

PGP encryption in E-Mails

In this implemenation, GnuPG(GPG) was used as the encryption standard.

OpenPGP allows us to encrypt and sign our data (in this report, email messages), so that these can be opened with the correct key.

Using a complete and free implementation of this OpenPGP standard, “GnuPG”, we can also use all of the front end applications and plugins that implement this specific cypher.

The Assignment.

In the assignment for this report, we try to implement the encryption of GPG. To test out the encryption, we will do the following:

1. Install a mail client, with plugin support. We chose Mozilla Thunderbird, because it is easy setup, and Mozilla is famous for it’s wide and easy access to plugins (as in firefox).
2. Install GnuPG libraries.
3. Install Enigmail encryption plugin.
   Enigmail is a plugin to implement a specific encryption. Thunderbird comes with OpenPGP and MIME support by default, but to use opensource or other free projects like GnuPG we need a client that can open and use encryption cypher on selection.
4. Create Key pairs. (private and public, as well as revocation key).
5. Send encrypted and signed Email.
6. Receive and decrypt email from previous point.
7. Try to upload Public key to an open PGP Key server.
8. Import key from other computer, decrypt message with this.

Implementing GnuPG encryption in Thunderbird Mail Client.

After installing Thunderbird, I installed the Enigmail plugin.

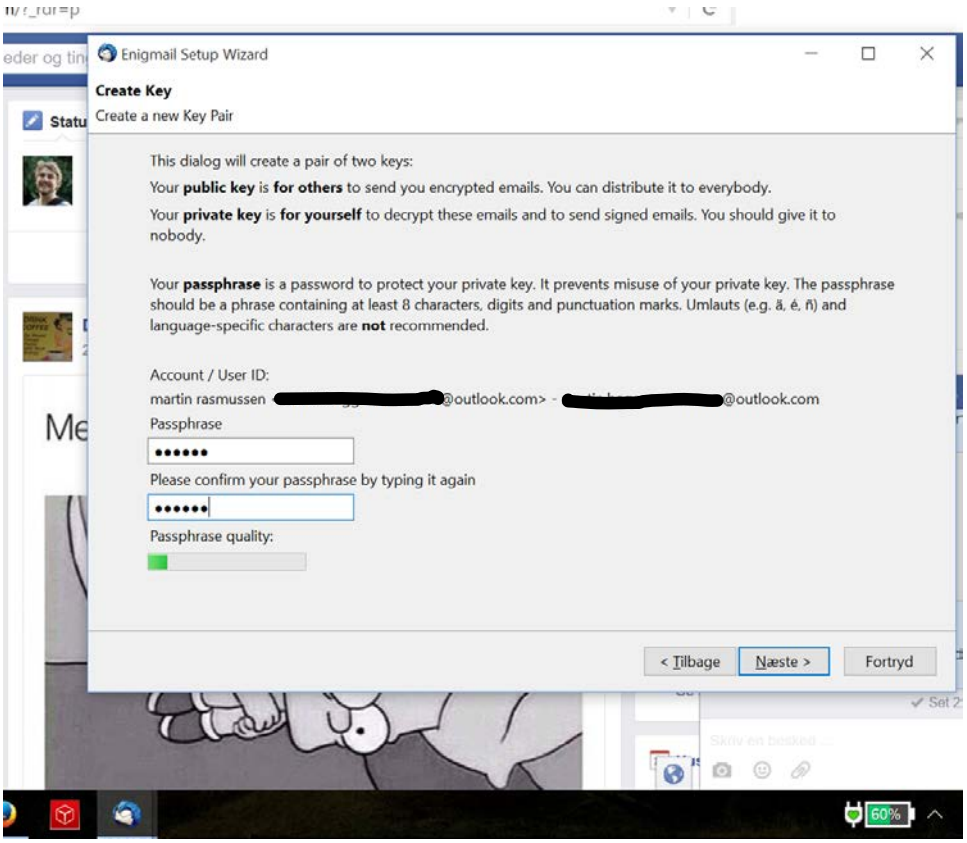
Upon finished installation, Enigmail prompts about the Key Generation Wizard, as it did not find any generation ones yet.

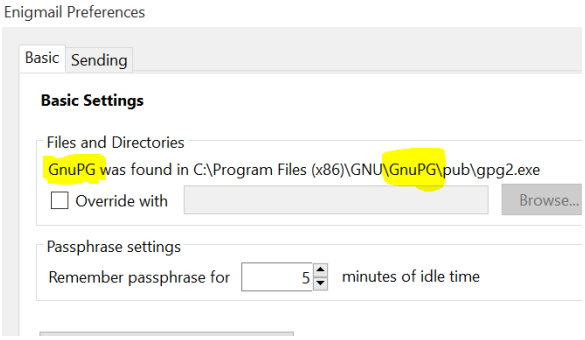
It automatically detects the installed encryption standard on the computer, as well as the email account logged in.

It asks for a passphrase (password for the key).

This was set to “12345678”.

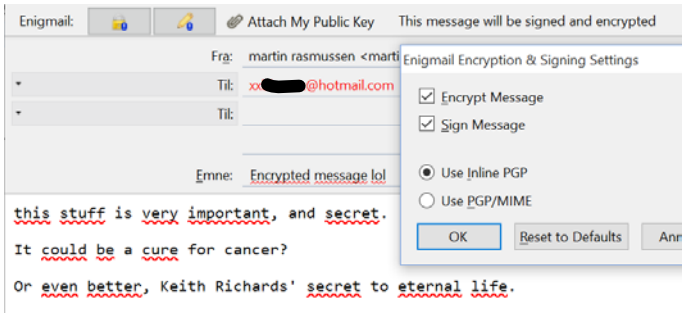
It also prompts for a Revocation key, which is a key that can deactivate the other keys upon losing your private key.





After generating all keys. I opened up the Key Management Settings, to make sure it registered the correct encryption model.

“GnuPG found in” followed by the standard directory, was in settings. This might have been more difficult if I had used a custom filepath instead of default installation.



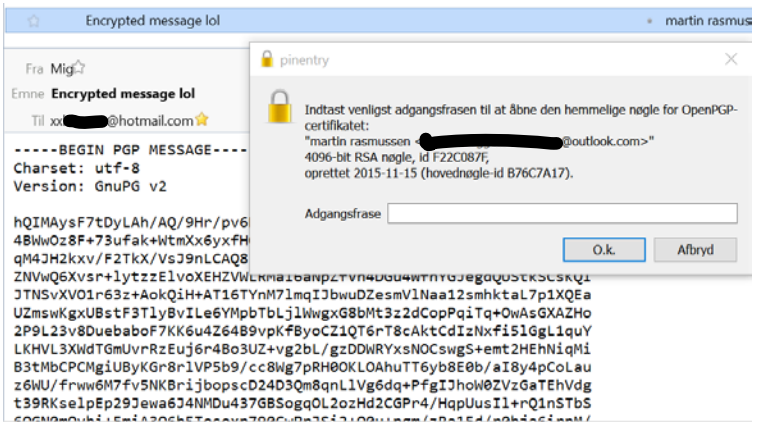
Now that all of the keys were generated, and the correct encryption model was double-checked, it was time to send some mails ☺!

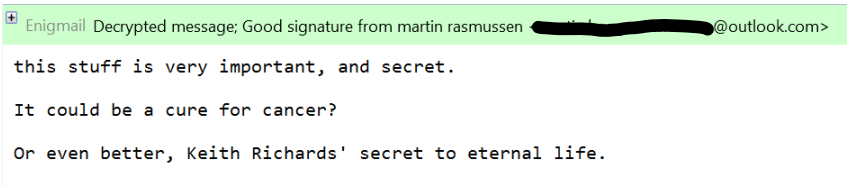
I created an email, and clicked the Enigmail plugin encrypt and sign area. Not to confuse with the Available MIME encryption already in Thunderbird.

The Email was sent to my own email, different alias.

Upon first opening the email, it is a bunch of gibberish, with header "------- BEGIN PGP MESSAGE --------" as well as version and charset.

From this, Enigmail recognizes the encryption, and upon clicking the red text stating that it isn't decrypted yet, asks for the passphrase to the key.

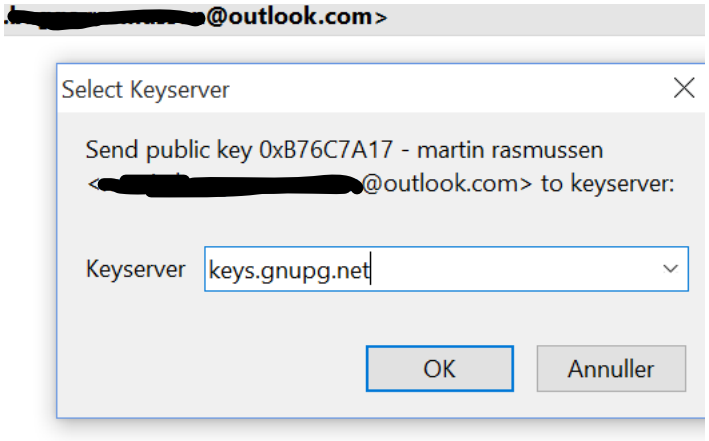




"12345678" and the gibberish is decrypted to the message from ealier.

We successfully sent (encrypted and signed) an email, and then received (decrypted from signature) said email.

Now, we try to upload our public key to a public PGP key server (we, publish it… so.. it is…. "public…!"



This was easy, especially using one of the three servers already configured in Enigmail.

I clicked the Key Management Settings and rightclicked the key. I chose "Upload to PGP Key Server".

Upon opening the upload window, I could choose my own or use a default. Seeing as I was using GnuPG, I chose their server on their own domain, keys.gnupg.net.