GPG Encryption Documentation

There are several files where it is required to encrypt the file separately before sending downstream. Below are instructions on how to setup gpg software and how to encrypt or decrypt files.

Initial Setup

- Installing GPG
  - Software can be procured here: https://www.gpg4win.org/
  - Install on desired server, accept the defaults
  - Software is profile based and is installed at this location: C:\Users\Username\AppData\Roaming\gnupg
  - They keystore can be accessed through the command line or through Kleopatra, the graphical tool that is installed with gpg for windows.
  - When you launch gpg you will have to create a key pair before you can import any public keys, create the key pair, and save the passphrase in a password manager.
  - Here is the gpg documentation for config options: https://www.gnupg.org/documentation/manuals/gnupg-devel/GPG-Configuration-Options.html

Encrypting a file

- In this scenario, a file will be encrypted and sent downstream.
  - Obtain the public key from the downstream party; the file should be a text file with a header and footer like below. In between the header and footer will be many random characters.
    - -----BEGIN PGP PUBLIC KEY BLOCK-----

      Version: PGP 6.5.1

      -----END PGP PUBLIC KEY BLOCK-----

- Import the public key into the gpg keystore.
  - Open Kleopatra, on the toolbar choose Import a certificate from a file.
  - Browse to the public key file and choose it.
  - Accept the prompts and trust the key, the new public key will now display in the Kleopatra window.
- Once the public key has been imported into the keystore, copy the keystore files out to a location so the script can use them.
  - Browse to the keystore location: C:\Users\Username\AppData\Roaming\gnupg
  - Copy the entire gnupg folder and paste it into a relevant location, these keystore files will be referenced by the script.
- Encrypt the file via powershell

  GPG Encrypt

  ```
  gpg --homedir "Path to gpg folder " --encrypt -r "Public Key Name" -o "Path to Encrypted output file\EncryptedFileName.pgp" "Path to input file and name\filename.txt"
  ```

- Command Reference

- --homedir: this tell gpg where to look for the public key used for encryption
- --encrypt: tells gpg you want to encrypt a file
- -r: this is the public key name from Kleopatra
- -o: this is output file which is encrypted with the public key
- This command should take target file and encrypt it with the public key and create a new output file. The encrypted output file should be sent to the FI.

Decrypting a File

- In this scenario, an encrypted file will be received and decrypted.
  - Using Kleopatra, create a key pair.
    - In the File menu, select New Key Pair.
    - Choose Open PGP as the key type.
    - Set a Name and Email Address.
    - Set a passphrase for the Key pair, save this passphrase in a password manager.
  - Copy the keystore from the user profile folder to a relevant location for decrypting the file.
    - Keystore location: C:\Users\Username\AppData\Roaming\gnupg
    - Copy the entire gnupg folder.
  - In Kleopatra choose the key pair you want to export and choose Export from the toolbar.
  - Save the file and send it to party who will encrypt the file.
  - The encrypting party will use the public key to encrypt the file, when the file is received, it will be decrypted with the private key.
  - Decrypt the file via powershell.

    GPG Decrypt

    ```
    gpg --homedir "Path to gnupg folder" --pinentry-mode=loopback --passphrase Password --output "Path to Decrypted Output File" --ignore-mdc-error --decrypt "Path to Encrypted Input File"
    ```

  - Command Reference
    - --homedir: sets the home directory for the key store.
    - --pinentry-mode=loopback: tells gpg to look at the passphrase in the command line.
    - --passphrase: the passphrase for the key pair being used to decrypt the file.
    - --decrypt: uses the private key to decrypt the input file.
    - --output: the file path and name of the decrypted file.

Troubleshooting

- If the gpg agent will not start when called, a message like this will display in the log. To fix, start the agent manually by running the gpg connect agent command below in a PowerShell window.
  - gpg: no running gpg-agent - starting 'C:\Program Files (x86)\GnuPG\bin\gpg-agent.exe'
    gpg: waiting for the agent to come up ... (5s)
    gpg: waiting for the agent to come up ... (4s)
    gpg: waiting for the agent to come up ... (3s)
    gpg: waiting for the agent to come up ... (2s)
    gpg: waiting for the agent to come up ... (1s)
    gpg: can't connect to the agent: IPC connect call failed
    gpg: encrypted with 2048-bit RSA key, ID C8EE9CC3AAA07EC7, created 2020-01-28
    gpg: decryption failed: No secret key

    Gpg Connect Agent

```
gpg-connect-agent /bye
```

Notes

- It is possible to reuse keystore files, the keystore folder can be copied and reused in multiple locations.
- It is best practice to remove homedir folders when the gpg agent process is finished with them. It is recommended to copy the homedir folder to a temp folder and then recursively remove it once the decryption process is complete. If this is not done the gpg agent process will stay open until the timeout period elapses.