Finanstilsynet MiFID II/MiFIR reporting - PKI installation and operations guide for Windows.

November 2017

Table of Contents

Version control

Version	Date	Author	Comments
1.0	16/11/2017	Trond A. S. Andersen	First draft
1.1	23/11/2017	Trond A. S. Andersen	Added chpt. 2.4.4.4 and chpt. 2.4.5.

1 Introduction

1.1 Purpose and audience of this document

This document describes the installation and configuration of the software required in order to encrypt and digitally sign report files submitted to Finanstilsynet as part of obliged transaction reporting and/or CPRS reporting according to MiFID II/MiFIR.
The intended audience of this document is

1) IT staff/technical personnel amongst the executing entities obliged to report to Finanstilsynet according to MiFID II/MiFIR, and/or
2) Technical personnel within Approved Reporting Mechanisms (ARMs) conducting reporting on behalf of executing entities as described in 1), as well as
3) Finanstilsynet's internal IT staff.

1.2 Prerequisites

1.2.1 General assumptions

In general, the guidelines/instructions provided by this document are assuming:

1. Submitting entity having a test environment and/or a production environment in place;
2. Submitting entity already having a X.509 formatted certificate in place, issued/signed by a trust service provider (TSP), which, according to most recent version of official service provider trust list for EU (ref. 1*) at any given time, is to be considered an officially qualified trust service provider
3. Finanstilsynet being notified of, having acknowledged and whitelisted, public IP address(es) identifying the computer(s) utilized by each submitting entity for file transmissions to the TRS II or the CPRS systems.
4. Submitting entities having obtained credentials for accessing the production environment and/or the test environment(s) for the TRS II and/or the CPRS reporting system(s) at Finanstilsynet.

1.2.2 Windows user account usage

For the correct use of the tools prescribed, Administrator access to the local system onto which the tools are deployed is required for all scenarios described throughout this document, including:

- Software installation
- Utilizing command prompts
- Configuring system path or other environment variables
- Executing any cryptographic operations

GnuPG is user sensitive in the sense that the tools included in the GnuPG package apply configuration settings on a per-user-basis. Therefore, using improper system accounts or insufficient credentials during setup could result in the tools being not operable or encryption and/or signing keys being unavailable.

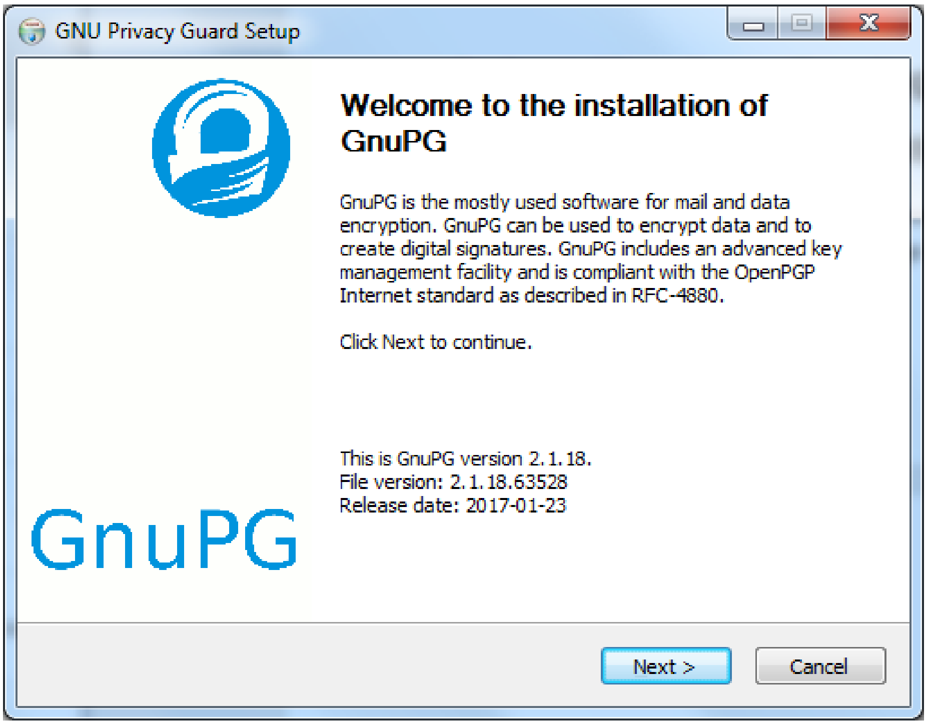
2 Installing & configuring GnuPG

2.1 Download GnuPG v2.1.18 for Windows

GnuPG was originally written for the Linux OS platform. However, there are various Windows compilations available, and with the purpose of transaction reporting and CPRS reporting to Finanstilsynet under MiFID II/MiFIR particularly in mind, the version v2.1.18 of the GnuPG software package has been thoroughly tested.
This software is available to download via the following links:
https://gnupg.org/ftp/gcrypt/binary/gnupg-w32-2.1.18_20170123.exe
ftp://ftp.gnupg.org/gcrypt/binary/gnupg-w32-2.1.18_20170123.exe

2.2 Install GnuPG

In order to start installation of the GnuPG tools, execute the installation file downloaded from the gnupg.org website. By executing this file, the window representing the initial step of the GnuPG installation wizard (shown below) should appear on the desktop:



GNU Privacy Guard Setup

Welcome to the installation of GnuPG

GnuPG is the mostly used software for mail and data encryption. GnuPG can be used to encrypt data and to create digital signatures. GnuPG includes an advanced key management facility and is compliant with the OpenPGP Internet standard as described in RFC-4880.

Click Next to continue.

This is GnuPG version 2.1.18.
File version: 2.1.18.63528
Release date: 2017-01-23

Figure 1- GnuPG Installation Wizard

As for the next steps of the procedure, it's advisable to just accept the default setup options suggested by the GnuPG installation wizard, simply pushing the Next button in the windows form associated with each subsequent step of the installation process.

2.3 Set the environment path to the GnuPG executable directory

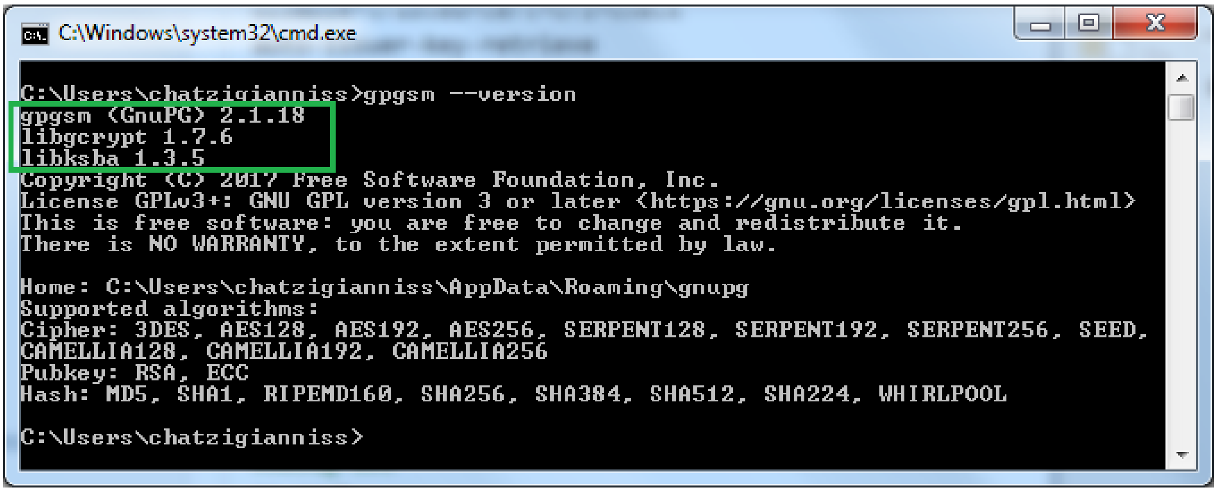
2.3.1 Add the directory to the path

In order to make utilization of the GnuPG tools more convenient and easier accessible to users, the path to the GnuPG executeables should be added to the system's environment path variable, using the following procedure:

1. Right click "This PC" -> Properties -> Advanced System Settings
2. Tab: Advanced -> Environment Variables
3. System Variables -> Path
4. Add the binary folder of GnuPG, eg C:\Program Files (x86)\GnuGP\bin

2.3.2 Verify that the path has been set correctly

If the environment path has been configured correctly, the gpgsm utility should be executable form any directory/folder on the host system.
In order to verify, open a new Command prompt window and run the following command:
gpgsm --version
If the environment path has been configured correctly, the gpgsm tool is executed and a screen similar to the one shown in the image below is displayed on the desktop:



```
C:\Users\chatzigianniss>gpgsm --version
gpgsm (GnuPG) 2.1.18
libgcrypt 1.7.6
libksba 1.3.5
Copyright (C) 2017 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: C:\Users\chatzigianniss\AppData\Roaming\gnupg
Supported algorithms:
Cipher: 3DES, AES128, AES192, AES256, SERPENT128, SERPENT192, SERPENT256, SEED,
CAMELLIA128, CAMELLIA192, CAMELLIA256
Pubkey: RSA, ECC
Hash: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224, WHIRLPOOL

C:\Users\chatzigianniss>
```

Figure 2- gpgsm --version (output)

The command output will also serve as verification that you are using the correct version of GnuPG. However, if any other or parallel version of GnuPG is installed, make sure to uninstall, so that only one instance/version of the GnuPG software is installed on the host system.

2.4 Configuring the GnuPG infrastructure

2.4.1 GnuPG components & configuration files

GnuPG contains a bundle of different tools and components. How these tools are configured and the extent of which they interact might vary with the preferences and utilization level within the environment of each submitting entity. Since the out-of-the-box, default configuration will constitute sort of a sufficient minimum for the TRSII and the CPRS reporting purposes, this document will not dwell beyond absolute necessity with the details of configuring the various tools included in the GnuPG package. The range of application for the main utilities are, however, briefly described in the following:

- **gpgsm**: Encryption & Singing tool. It is this tools that should be utilized by submitting entities for digital encryption and signing operations, based on qualified X.509 certificates.
- **gpg-agent**: This tool is used as a backend service for gpgsm, providing private (secret) key management functions.
- **dirmngr**: The directory manager, or dirmngr for short, can be configured to perform automatic certificate revocation control operations, such as checking against a CRL lists or query the OCSP responders. Dirmngr can also access the OpenPGP keyserver and automatically download CRLs and certificates.

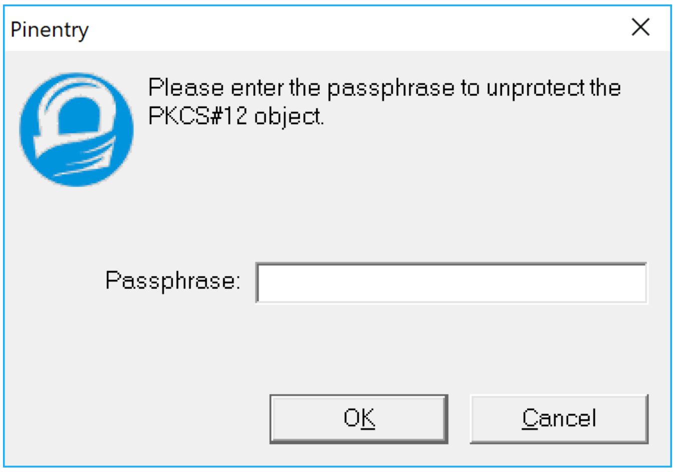
For each of the tools mentioned above, the respective config file(s) is accessed and configured from the GnuPG's homedir, which is by default located at %Appdata%/gnupg, eg. C:\Users\<local user>\AppData\Roaming\gnupg. However, for in-depth instructions on configuring the various tools/services, confer the official GnuPG manual pages, https://www.gnupg.org/documentation/manuals/gnupg/.

2.4.2 Making Submitting Entity's own certificates/keys utilizable for GnuPG

X.509 certificates issued by qualified certificate authorities are typically delivered in the form of PKCS#12-formatted files. The encryption keys incorporated in such certificate files can be imported directly into the submitting entities' GnuPG infrastructure, using the gpgsm utility with the --import option. In order to import a PKCS#12-formatted certificate file issued to a submitting enitity, typically containing both public and private encryption and/or signing keys as well as the entire chain of trust for that particular certificate, the following gpgsm command is submitted, using the command prompt:

gpgsm--import <path\filename>.p12

When importing a certificate containing a private encryption key, the gpgsm utility will typically prompt the user to enter the secret password associated with the private key incorporated in the certificate:

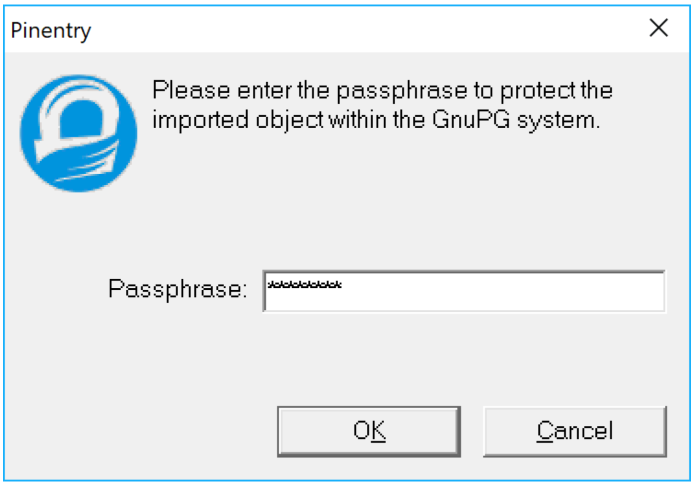


Please enter the passphrase to unprotect the PKCS#12 object.

Passphrase:

Figure 3 - PKCS#12 enter private key password

As the next step, the gpgsm utility will prompt the user to enter (and re-enter) a password for protecting the imported private encryption key internally, within the local GnuPG instance:

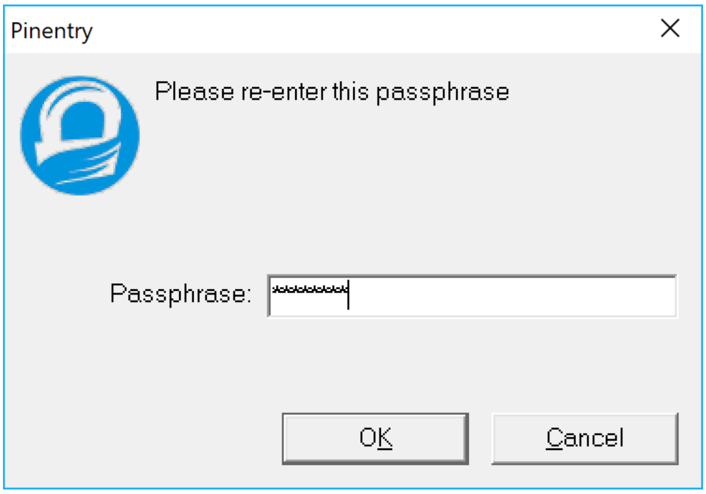


Pinentry

Please enter the passphrase to protect the imported object within the GnuPG system.

Passphrase: ********

Figure 4 - Enter local GnuPG private key password

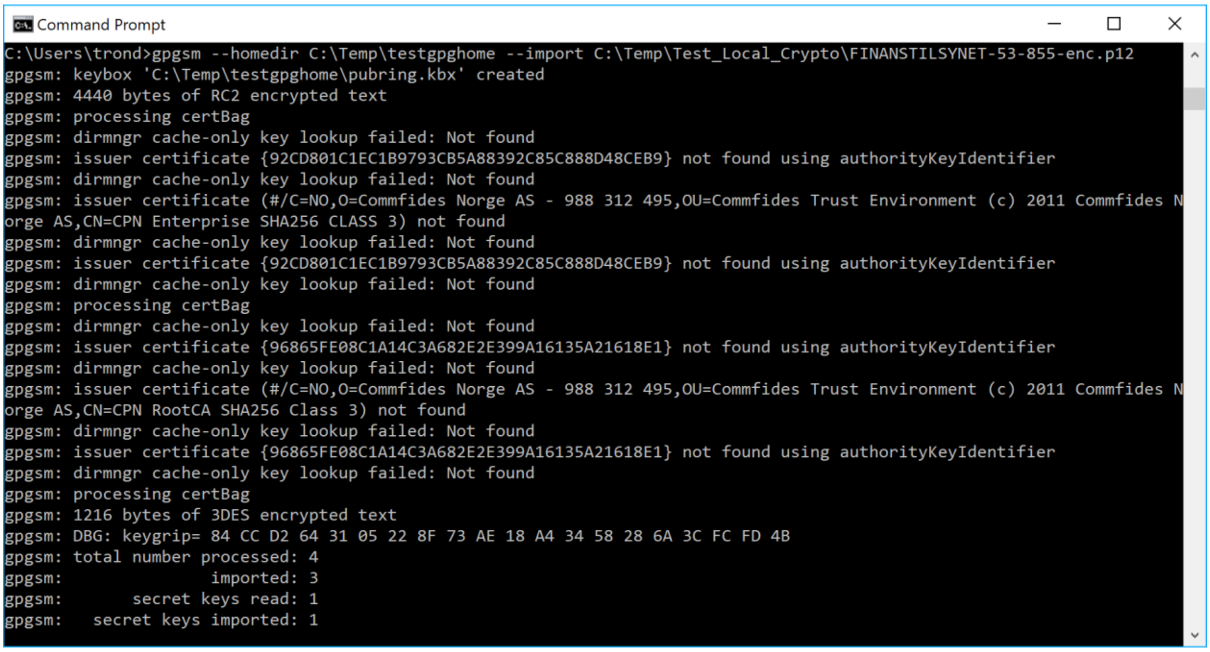


Pinentry

Please re-enter this passphrase

Passphrase: ********

Figure 5 - Re-enter local GnuPG private key password



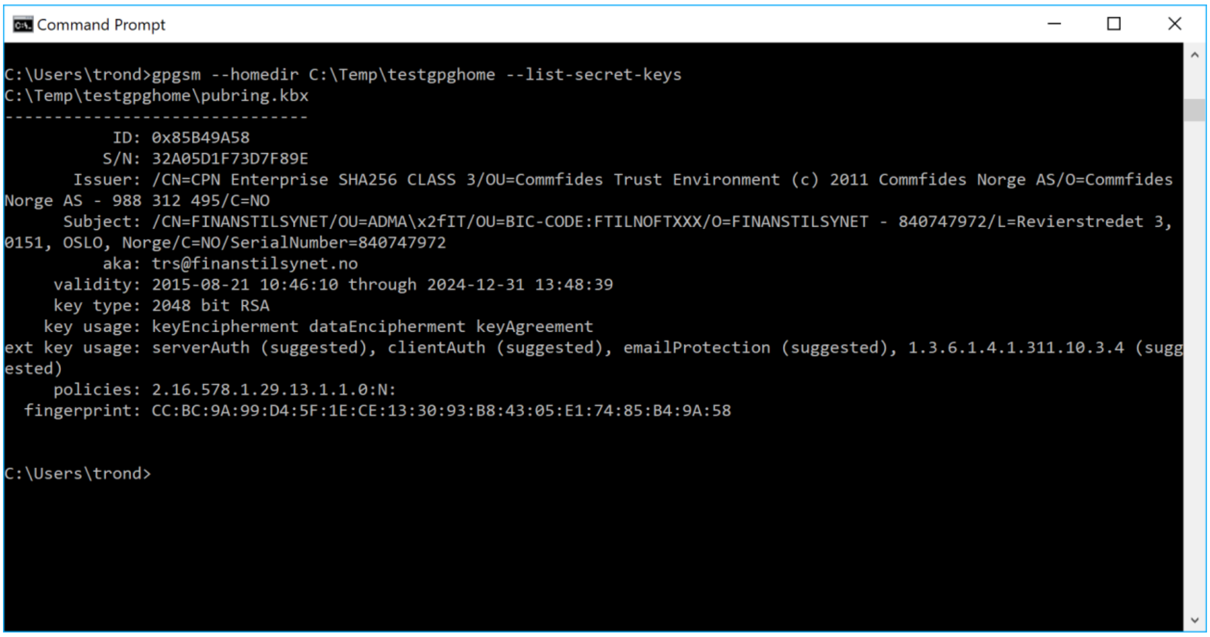
```
C:\Users\trond>gpgsm --homedir C:\Temp\testgpghome --import C:\Temp\Test_Local_Crypto\FINANSTILSYNET-53-855-enc.p12
gpgsm: keybox 'C:\Temp\testgpghome\pubring.kbx' created
gpgsm: 4440 bytes of RC2 encrypted text
gpgsm: processing certBag
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: issuer certificate {92CD801C1EC1B9793CB5A88392C85C888D48CEB9} not found using authorityKeyIdentifier
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: issuer certificate (#/C=NO,O=Commfides Norge AS - 988 312 495,OU=Commfides Trust Environment (c) 2011 Commfides N
orge AS,CN=CPN Enterprise SHA256 CLASS 3) not found
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: issuer certificate {92CD801C1EC1B9793CB5A88392C85C888D48CEB9} not found using authorityKeyIdentifier
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: processing certBag
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: issuer certificate {96865FE08C1A14C3A682E2E399A16135A21618E1} not found using authorityKeyIdentifier
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: issuer certificate (#/C=NO,O=Commfides Norge AS - 988 312 495,OU=Commfides Trust Environment (c) 2011 Commfides N
orge AS,CN=CPN RootCA SHA256 Class 3) not found
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: issuer certificate {96865FE08C1A14C3A682E2E399A16135A21618E1} not found using authorityKeyIdentifier
gpgsm: dirmngr cache-only key lookup failed: Not found
gpgsm: processing certBag
gpgsm: 1216 bytes of 3DES encrypted text
gpgsm: DBG: keygrip= 84 CC D2 64 31 05 22 8F 73 AE 18 A4 34 58 28 6A 3C FC FD 4B
gpgsm: total number processed: 4
gpgsm:               imported: 3
gpgsm:       secret keys read: 1
gpgsm:   secret keys imported: 1
```

Figure 6 - gpgsm --import command output

2.4.3 List Submitting Entity’s private key for verification

Ensure that the private key incorporated in PKCS#12-formatted certificate file are present in gpgsm’s keychain:

gpgsm --list-secret-keys



```
C:\Users\trond>gpgsm --homedir C:\Temp\testgpghome --list-secret-keys
C:\Temp\testgpghome\pubring.kbx
-------------------------------
           ID: 0x85B49A58
          S/N: 32A05D1F73D7F89E
       Issuer: /CN=CPN Enterprise SHA256 CLASS 3/OU=Commfides Trust Environment (c) 2011 Commfides Norge AS/O=Commfides
Norge AS - 988 312 495/C=NO
      Subject: /CN=FINANSTILSYNET/OU=ADMA\x2fIT/OU=BIC-CODE:FTILNOFTXXX/O=FINANSTILSYNET - 840747972/L=Revierstredet 3,
0151, OSLO, Norge/C=NO/SerialNumber=840747972
          aka: trs@finanstilsynet.no
     validity: 2015-08-21 10:46:10 through 2024-12-31 13:48:39
     key type: 2048 bit RSA
    key usage: keyEncipherment dataEncipherment keyAgreement
ext key usage: serverAuth (suggested), clientAuth (suggested), emailProtection (suggested), 1.3.6.1.4.1.311.10.3.4 (sugg
ested)
     policies: 2.16.578.1.29.13.1.1.0:N:
  fingerprint: CC:BC:9A:99:D4:5F:1E:CE:13:30:93:B8:43:05:E1:74:85:B4:9A:58


C:\Users\trond>
```

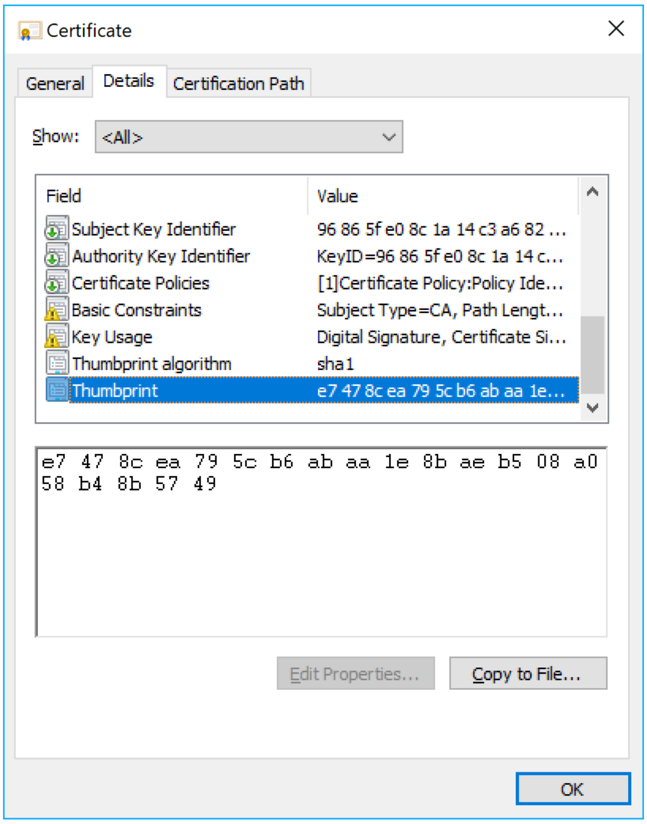
Figure 7- gpgsm --list-secret-keys comand output

2.4.4 Configure trust for required chain of certificates

Together, 1) Finanstilsynet's encryption certificate issued by Commfides, containing the public key required for report file encryption, 2) the associated Commfides Issuer CA certificates and 3) the parent Commfides Root CA certificate constitute a certificate chain that must be imported into each and every submitting entity's local GnuPG keyring in order for the cryptographic operation required in the TRS II and CPRS report file interchange to work. These certificates are available for the submitting entities to download from the /Public/Cert/ folder, residing under each submitting entity's respective home directory on the TRS II/CPRS STFP servers. The certificates may be obtained one-by-one, by downloading the following DER encoded binary X.509 files:

- *FINANSTILSYNET-53-855-enc.cer (fingerprint: CC:BC:9A:99:D4:5F:1E:CE:13:30:93:B8:43:05:E1:74:85:B4:9A:58)*
- *COMMFIDES-CPN-Enterprise-SHA256-CLASS-3.cer (fingerprint: DC:38:AC:1C:B3:2A:5F:85:08:14:09:89:98:DA:D1:35:83:16:F4:86)*
- *COMMFIDES-CPN-RootCA-SHA256-Class-3.cer (fingerprint: E7:47:8C:EA:79:5C:B6:AB:AA:1E:8B:AE:B5:08:A0:58:B4:8B:57:49)*

Alternatively, a single file with the entire certificate chain, *FINANSTILSYNET-53-855-CERT-CHAIN.p7b* may be downloaded and installed/imported in the local GnuPG infrastructure as a whole, in one operation.
When downloaded to the submitting entity's local system, the fingerprint for the rsa keys incorporated in these certificate files can be verified using the Windows certmngr tool or, alternatively, the Certificate Snap-in of the Microsoft Management Console:



Show: <All>

Field	Value
Subject Key Identifier	96 86 5f e0 8c 1a 14 c3 a6 82 ...
Authority Key Identifier	KeyID=96 86 5f e0 8c 1a 14 c...
Certificate Policies	[1]Certificate Policy:Policy Ide...
Basic Constraints	Subject Type=CA, Path Lengt...
Key Usage	Digital Signature, Certificate Si...
Thumbprint algorithm	sha1
Thumbprint	e7 47 8c ea 79 5c b6 ab aa 1e...

```
e7 47 8c ea 79 5c b6 ab aa 1e 8b ae b5 08 a0
58 b4 8b 57 49
```

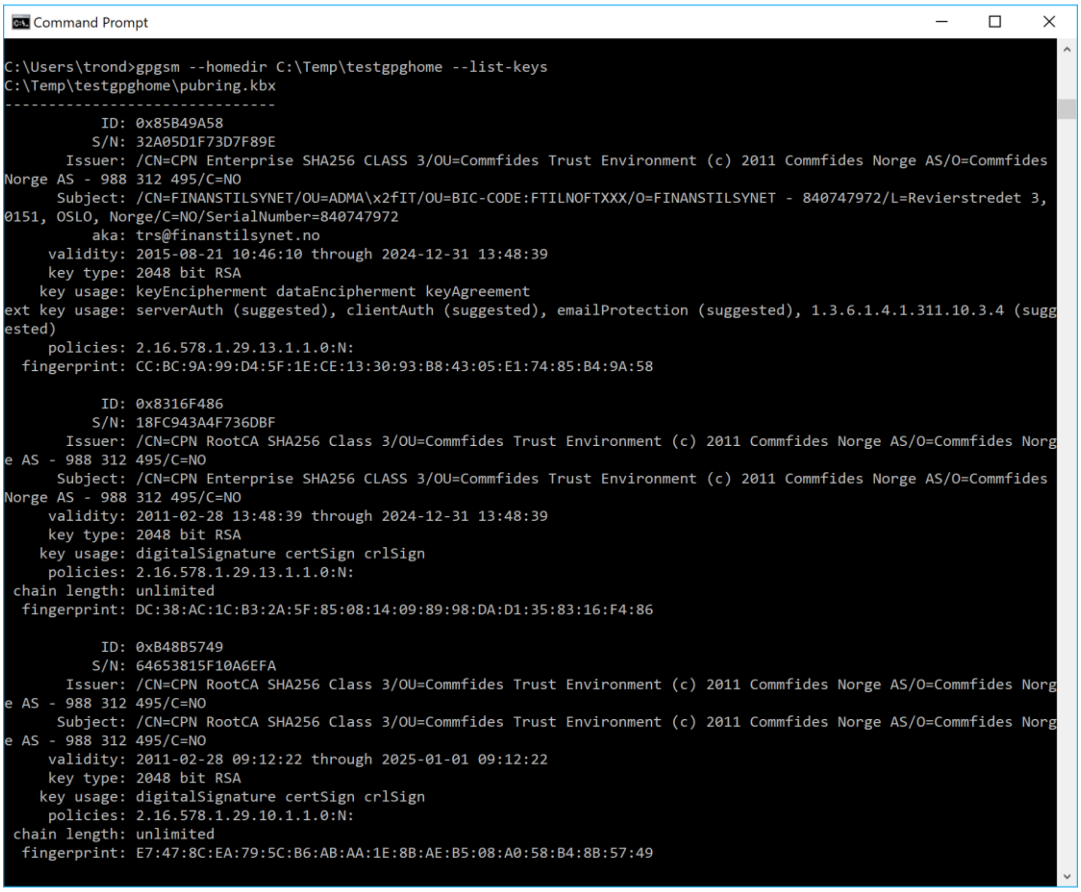
Figure 8-Windows certmngr

Regardless of whether the certificates are obtained in one single PKCS#7 file or they are downloaded one-by-one in the form of DER encoded binaries, the encryption keys incorporated in the certificates are imported into the local GnuPG infrastructure using the --import command of the gpgsm utility, according to the following pattern:
gpgsm--import <path\filename>[.p7b][.cer]

2.4.4.1 List public keys for verification

Having imported the certificate file(s) according to the preceding steps, ensure that the Root CA certificate, the Issuing CA certificate and the Recipient (Finanstilsynet's) certificate are present in gpgsm's keychain.

gpgsm --list-keys



```
C:\Users\trond>gpgsm --homedir C:\Temp\testgpghome --list-keys
C:\Temp\testgpghome\pubring.kbx
-------------------------------
           ID: 0x85B49A58
          S/N: 32A05D1F73D7F89E
       Issuer: /CN=CPN Enterprise SHA256 CLASS 3/OU=Commfides Trust Environment (c) 2011 Commfides Norge AS/O=Commfides
Norge AS - 988 312 495/C=NO
      Subject: /CN=FINANSTILSYNET/OU=ADMA\x2fIT/OU=BIC-CODE:FTILNOFTXXX/O=FINANSTILSYNET - 840747972/L=Revierstredet 3,
0151, OSLO, Norge/C=NO/SerialNumber=840747972
          aka: trs@finanstilsynet.no
     validity: 2015-08-21 10:46:10 through 2024-12-31 13:48:39
     key type: 2048 bit RSA
    key usage: keyEncipherment dataEncipherment keyAgreement
ext key usage: serverAuth (suggested), clientAuth (suggested), emailProtection (suggested), 1.3.6.1.4.1.311.10.3.4 (sugg
ested)
     policies: 2.16.578.1.29.13.1.1.0:N:
  fingerprint: CC:BC:9A:99:D4:5F:1E:CE:13:30:93:B8:43:05:E1:74:85:B4:9A:58

           ID: 0x8316F486
          S/N: 18FC943A4F736DBF
       Issuer: /CN=CPN RootCA SHA256 Class 3/OU=Commfides Trust Environment (c) 2011 Commfides Norge AS/O=Commfides Norg
e AS - 988 312 495/C=NO
      Subject: /CN=CPN Enterprise SHA256 CLASS 3/OU=Commfides Trust Environment (c) 2011 Commfides Norge AS/O=Commfides
Norge AS - 988 312 495/C=NO
     validity: 2011-02-28 13:48:39 through 2024-12-31 13:48:39
     key type: 2048 bit RSA
    key usage: digitalSignature certSign crlSign
     policies: 2.16.578.1.29.13.1.1.0:N:
 chain length: unlimited
  fingerprint: DC:38:AC:1C:B3:2A:5F:85:08:14:09:89:98:DA:D1:35:83:16:F4:86

           ID: 0xB48B5749
          S/N: 64653815F10A6EFA
       Issuer: /CN=CPN RootCA SHA256 Class 3/OU=Commfides Trust Environment (c) 2011 Commfides Norge AS/O=Commfides Norg
e AS - 988 312 495/C=NO
      Subject: /CN=CPN RootCA SHA256 Class 3/OU=Commfides Trust Environment (c) 2011 Commfides Norge AS/O=Commfides Norg
e AS - 988 312 495/C=NO
     validity: 2011-02-28 09:12:22 through 2025-01-01 09:12:22
     key type: 2048 bit RSA
    key usage: digitalSignature certSign crlSign
     policies: 2.16.578.1.29.10.1.1.0:N:
 chain length: unlimited
  fingerprint: E7:47:8C:EA:79:5C:B6:AB:AA:1E:8B:AE:B5:08:A0:58:B4:8B:57:49
```

Figure 9 - gpgsm --list-keys command output

2.4.4.2 Locate the trustlist file

Open the file trustlist.txt., per default residing under the gnupg homedir. Unless another location has been explicitly specified using the --homedir option with the gpgsm utility, the trustlist.txt should be located at %Appdata%\Roaming\GnuPG

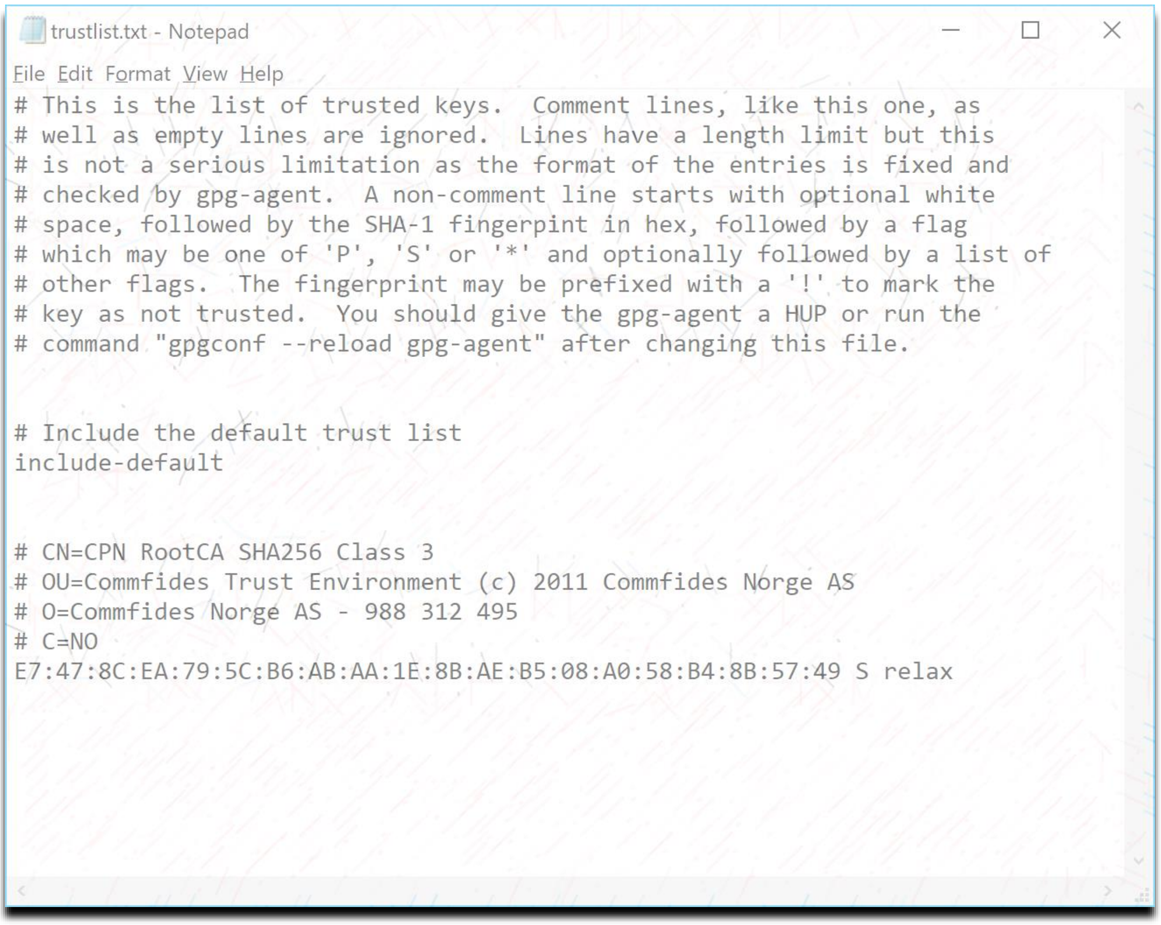
If the trustlist.txt does not exist, simply create a new trustlist.txt file and add the following lines:
# Include the default trust list
include-default

2.4.4.3 Add the fingerprint of the Commfides Root CA certificate to the trustlist

In the trustlist.txt ,add an entry at the end of trustlist.txt with the fingerprint of the Commfides Root CA certificate, followed by the character string"'S relax"
Ie, in this example, add the following text at the end of the file:

# CN=CPN RootCA SHA256 Class 3
# OU=Commfides Trust Environment (c) 2011 Commfides Norge AS
# O=Commfides Norge AS - 988 312 495
# C=NO
E7:47:8C:EA:79:5C:B6:AB:AA:1E:8B:AE:B5:08:A0:58:B4:8B:57:49 S relax

Having saved the trustlist.txt, it should (opened in notepad) look as follows:



```
# This is the list of trusted keys.  Comment lines, like this one, as
# well as empty lines are ignored.  Lines have a length limit but this
# is not a serious limitation as the format of the entries is fixed and
# checked by gpg-agent.  A non-comment line starts with optional white
# space, followed by the SHA-1 fingerpint in hex, followed by a flag
# which may be one of 'P', 'S' or '*' and optionally followed by a list of
# other flags.  The fingerprint may be prefixed with a '!' to mark the
# key as not trusted.  You should give the gpg-agent a HUP or run the
# command "gpgconf --reload gpg-agent" after changing this file.


# Include the default trust list
include-default


# CN=CPN RootCA SHA256 Class 3
# OU=Commfides Trust Environment (c) 2011 Commfides Norge AS
# O=Commfides Norge AS - 988 312 495
# C=NO
E7:47:8C:EA:79:5C:B6:AB:AA:1E:8B:AE:B5:08:A0:58:B4:8B:57:49 S relax
```

Figure 10 - trustlist.txt content

2.4.4.4 Reload gpgsm configuration to use the updated trustlist

Having updated the trustlist.txt according to the procedure described in 2.4.4.3, issue the following command:

gpgconf –reload

The new trustlist, incuding the key identified by the fingerprint just added, will now be loaded and ready to use.

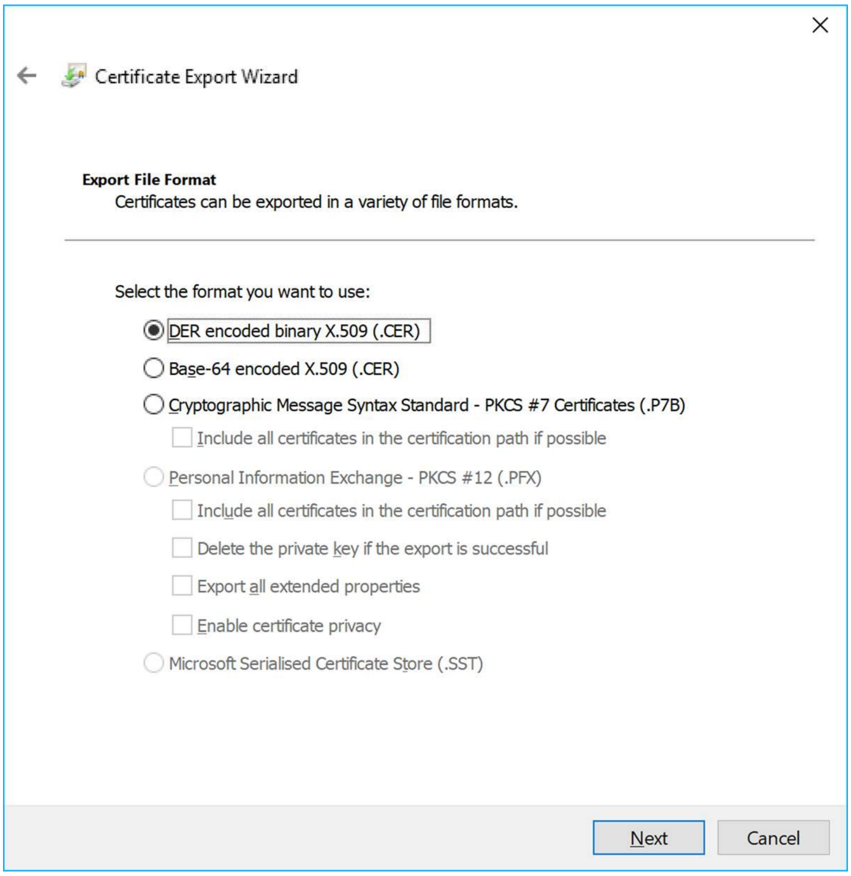
2.4.5 Making the submitting entities' public key available to Finanstilsynet.

In order for Finanstilsynet to verify digital signatures applied to report files from a given submitting entity, according to the procedure described in 3.2, certificates with the submitting entity's public signing key will have to be exported from the submitting entity's local GnuPG infrastructure and uploaded to Finanstilsynet.

<SEIC>.cer

2.4.5.1 Exporting the submitting entity's signing certificate.

Using the Windows *certmgr* tool or, alternatively, the *Certificate Snap*-in of the Microsoft Management Console, export the submitting entity's public signing key in a DER encoded binary X.509 certificate file, named according to the following file naming pattern:



Export File Format

Certificates can be exported in a variety of file formats.

Select the format you want to use:

● DER encoded binary X.509 (.CER)

○ Base-64 encoded X.509 (.CER)

○ Cryptographic Message Syntax Standard - PKCS #7 Certificates (.P7B)

□ Include all certificates in the certification path if possible

○ Personal Information Exchange - PKCS #12 (.PFX)

□ Include all certificates in the certification path if possible

□ Delete the private key if the export is successful

□ Export all extended properties

□ Enable certificate privacy

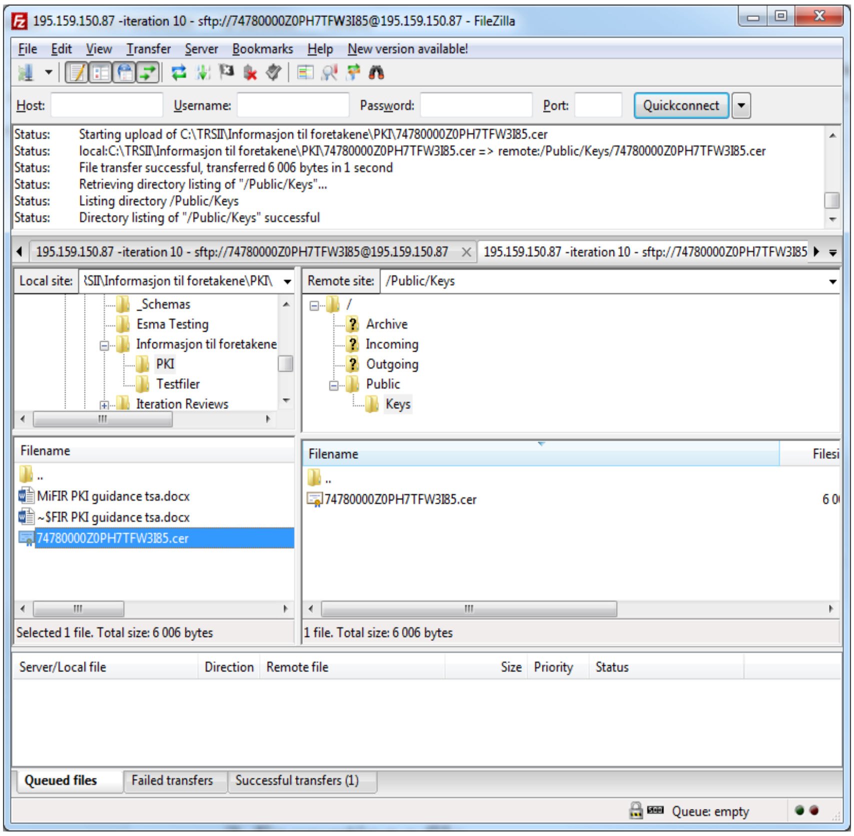
○ Microsoft Serialised Certificate Store (.SST)

Figure 11 – Exporting submitting entity's signing key, using Windows MMC

2.4.5.2 Uploading the exported signing certificate to Finanstilsynet.

Using any preferred SFTP enabled client software, i.e. Filezilla FTP Client, and the STFP logon credentials provided by Finanstilsynet:

1) log on to the SFTP server constituting the submitting entities' file exchange interface towards the TRS II and/or the CPRS, and
2) upload the exported certificate file to the /Public/Keys folder, residing under the home directory of the submitting entity's SFTP user.



195.159.150.87 -iteration 10 - sftp://74780000Z0PH7TFW3I85@195.159.150.87 - FileZilla

```
Status:	Starting upload of C:\TRSII\Informasjon til foretakene\PKI\74780000Z0PH7TFW3I85.cer
Status:	local:C:\TRSII\Informasjon til foretakene\PKI\74780000Z0PH7TFW3I85.cer => remote:/Public/Keys/74780000Z0PH7TFW3I85.cer
Status:	File transfer successful, transferred 6 006 bytes in 1 second
Status:	Retrieving directory listing of "/Public/Keys"...
Status:	Listing directory /Public/Keys
Status:	Directory listing of "/Public/Keys" successful
```

Local site: ꞉SII\Informasjon til foretakene\PKI\

Remote site: /Public/Keys

- _Schemas
- Esma Testing
- Informasjon til foretakene
  - PKI
  - Testfiler
- Iteration Reviews

- /
  - Archive
  - Incoming
  - Outgoing
  - Public
    - Keys

Filename
..
MiFIR PKI guidance tsa.docx
~$FIR PKI guidance tsa.docx
74780000Z0PH7TFW3I85.cer

Selected 1 file. Total size: 6 006 bytes

Filename	Filesi
..	
74780000Z0PH7TFW3I85.cer	6 0

1 file. Total size: 6 006 bytes

Queue: empty

Figure 12 – Uploading submitting entities singing certificate, using FileZilla.

3 Encrypting a file

3.1 Issue the encrypt command

Encryption parameters:

- **OCSP & CRL checking**: Whenever the encrypt command is issued, gpgsm could, dependent of the dirmngr configuration, query the OCSP responder and a CRL list to determine the validity of the certificates used. This is done by gpgsm calling dirmngr behind-the-scene. If the revocation checking cannot be completed, the operation will fail. In order to avoid performing such checks, add the parameters --disable-ocsp and --disable-crl-checks to the command line.
- **Recipient**: This parameter value must provide a unique reference to the public certificate of the recipient. This reference either be an ID, a Subject name, an email address, a key/certificate fingerprint, a keygrip, or practically any other information detail incorporated in a certificate that have been successfully imported into the GnuPG infrastructure. However, ensure to choose a value which is unique, or else gpgsm will return an error that the value given is ambiguous. For this example, the ID of a certificate is used to identify the receipient, as it is unique within the keychain. For more information about specifying a recipient, please confer the GnuPG manual pages, https://www.gnupg.org/documentation/manuals/gnupg/ .

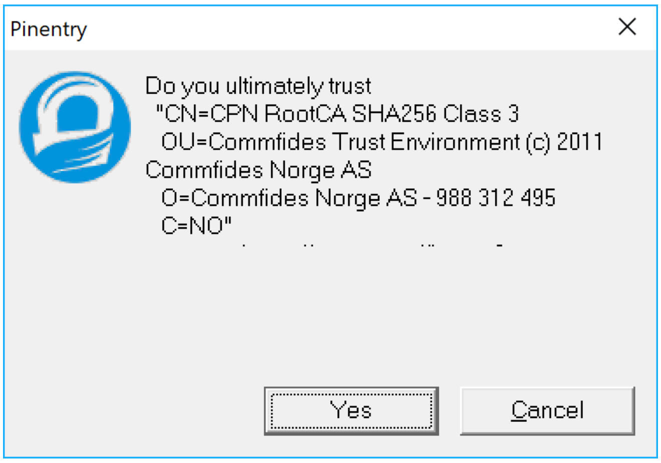
The gpgsm command line utilized for encrypting a transaction report file destined for the TRS II system at Finanstilsynet should be formatted according to the following pattern:

```
gpgsm --recipient <recipient certificate ID> --encrypt
TR_<SEIC>_<ORI>_<YYYYMMDD>_<RFSEQ>.ZIP > <output encrypted filename>
```

For example:

```
gpgsm --recipient 0x85B49A58 --encrypt testdatafile.zip > encryptedfile.gpg
```

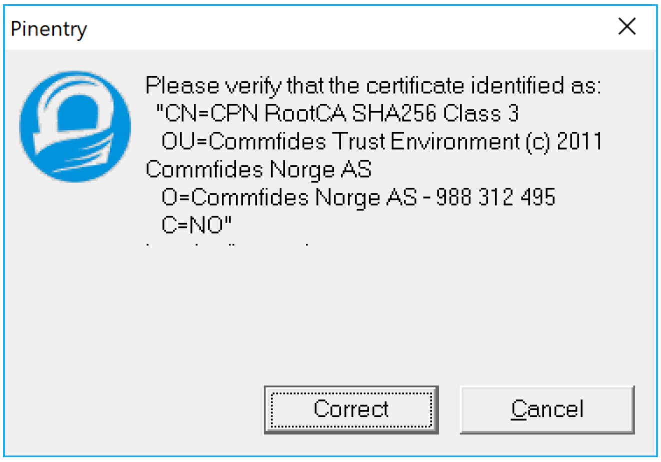
However, if the Root CA certificate has not been marked as trusted (2.4.4.3), gpgsm will prompt the user to explicitly confirm this certificate's general trustworthiness:



Do you ultimately trust
"CN=CPN RootCA SHA256 Class 3
OU=Commfides Trust Environment (c) 2011
Commfides Norge AS
O=Commfides Norge AS - 988 312 495
C=NO"

Click "Yes".

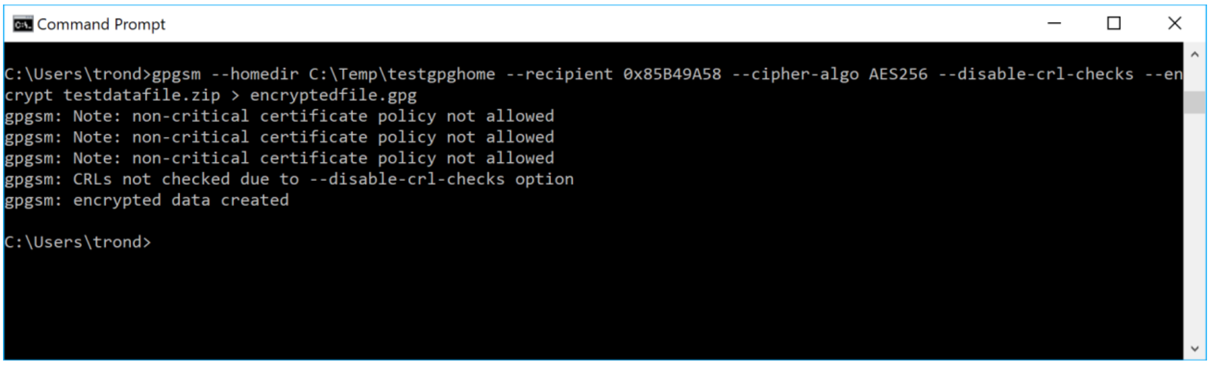
By clicking "Yes", the following window dialog will appear on the desktop:



Please verify that the certificate identified as:
 "CN=CPN RootCA SHA256 Class 3
  OU=Commfides Trust Environment (c) 2011
Commfides Norge AS
  O=Commfides Norge AS - 988 312 495
  C=NO"

Click “Correct”.

The final output will look as follows:



```
C:\Users\trond>gpgsm --homedir C:\Temp\testgpghome --recipient 0x85B49A58 --cipher-algo AES256 --disable-crl-checks --en
crypt testdatafile.zip > encryptedfile.gpg
gpgsm: Note: non-critical certificate policy not allowed
gpgsm: Note: non-critical certificate policy not allowed
gpgsm: Note: non-critical certificate policy not allowed
gpgsm: CRLs not checked due to --disable-crl-checks option
gpgsm: encrypted data created

C:\Users\trond>
```

A new file called encryptedfile.gpg will have been created in the current folder.

3.2 Issue the sign command

Signing a file will apply a digital signature from a given key stored in the local GnuPG keyring to an existing file. The output from the sign operation is a file which contains the original file and the generated signature.
Signing parameters:

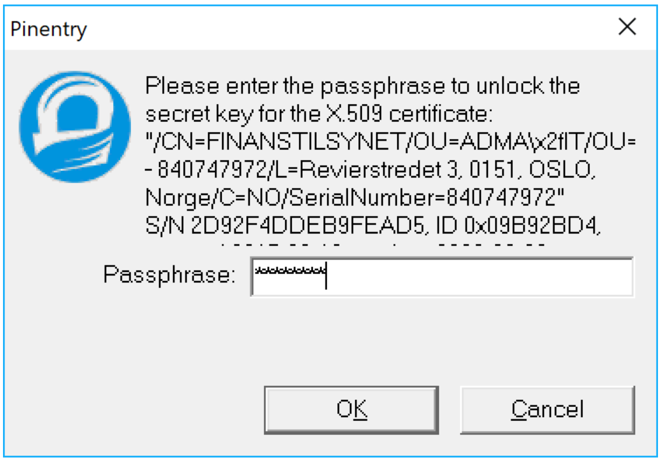
- **OCSP & CRL checking**: Whenever the sign command is issued, gpgsm could, dependent of the local dirmngr configuration, query the OCSP responder and a CRL list to determine the validity of the certificates used. This is done by gpgsm calling dirmngr internally behind-the-scene. If the revocation checking cannot be completed, the operation will fail. In order to avoid performing such checks, add the parameters --disable-ocsp and --disable-crl-checks to the command line.

- **Signer key**: By default, gpgsm will use the first key found in the local secret keychain. The ID of the key may, however, be overridden using the --local-user option, although it is recommended to use the latter alternative, in order to avoid confusion.

When, i.e., a submitting entity is applying it’s digital signature to a file destined for the TRS II transaction report system the gpgsm command line should be formatted according to the following pattern:
gpgsm --local-user <signer key ID> --sign <filename to sign> >
TR_<SEIC>_<ORI>_<YYYYMMDD>_<RFSEQ>.<TYPE>

For example:
gpgsm --local-user 0x0C54A996 --sign encryptedfile.gpg > TR_5967007LIEEXZXJUBK44_01_20171117_0006.zip

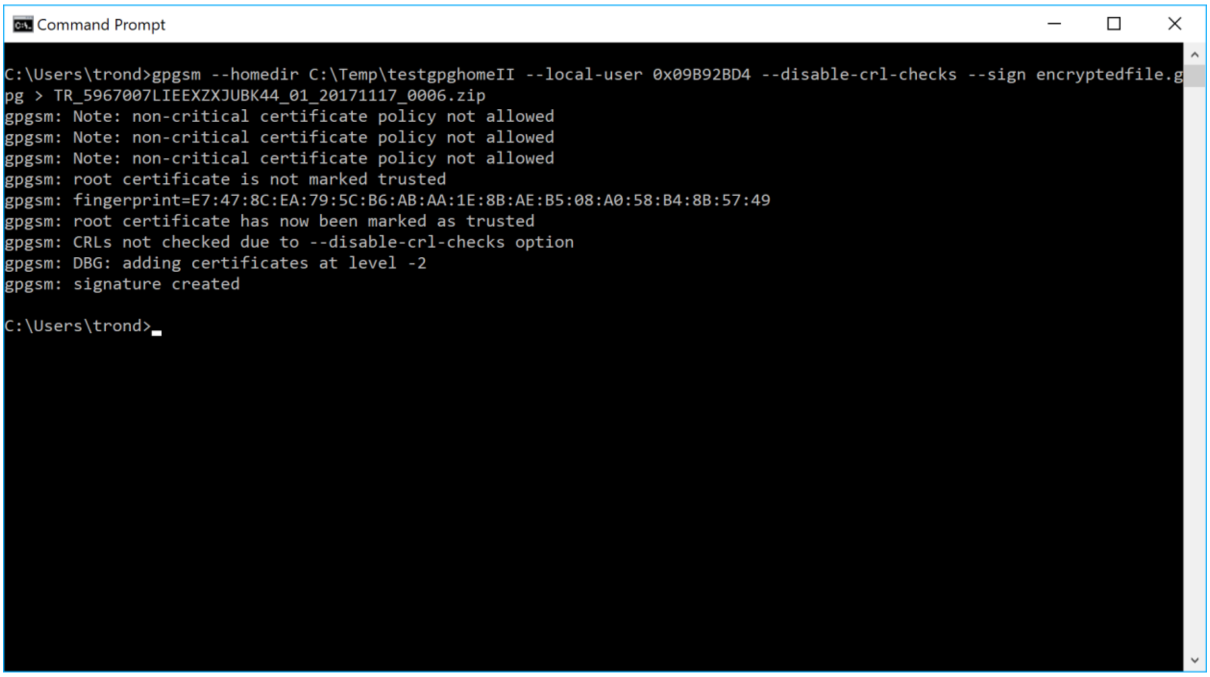
If the submitting entity's private key has been protected with a password, gpgsm will then prompt the user to enter it. The output may look as follows:



Please enter the passphrase to unlock the secret key for the X.509 certificate:
"/CN=FINANSTILSYNET/OU=ADMA\x2flT/OU=
- 840747972/L=Revierstredet 3, 0151, OSLO,
Norge/C=NO/SerialNumber=840747972"
S/N 2D92F4DDEB9FEAD5, ID 0x09B92BD4,

Passphrase: ********

The final output will look as follows:



```
C:\Users\trond>gpgsm --homedir C:\Temp\testgpghomeII --local-user 0x09B92BD4 --disable-crl-checks --sign encryptedfile.g
pg > TR_5967007LIEEXZXJUBK44_01_20171117_0006.zip
gpgsm: Note: non-critical certificate policy not allowed
gpgsm: Note: non-critical certificate policy not allowed
gpgsm: Note: non-critical certificate policy not allowed
gpgsm: root certificate is not marked trusted
gpgsm: fingerprint=E7:47:8C:EA:79:5C:B6:AB:AA:1E:8B:AE:B5:08:A0:58:B4:8B:57:49
gpgsm: root certificate has now been marked as trusted
gpgsm: CRLs not checked due to --disable-crl-checks option
gpgsm: DBG: adding certificates at level -2
gpgsm: signature created

C:\Users\trond>_
```

A new file called TR_5967007LIEEXZXJUBK44_01_20171117_0006.zip, encrypted, signed and ready for upload to the TRS' SFP server, should have been created in the current folder.

Annex 1 - Reference documents

Ref	Title	Ver	Author/Issuer	Date
1	EU Trusted Lists	N/A	The Directorate-General for	18/3/2013