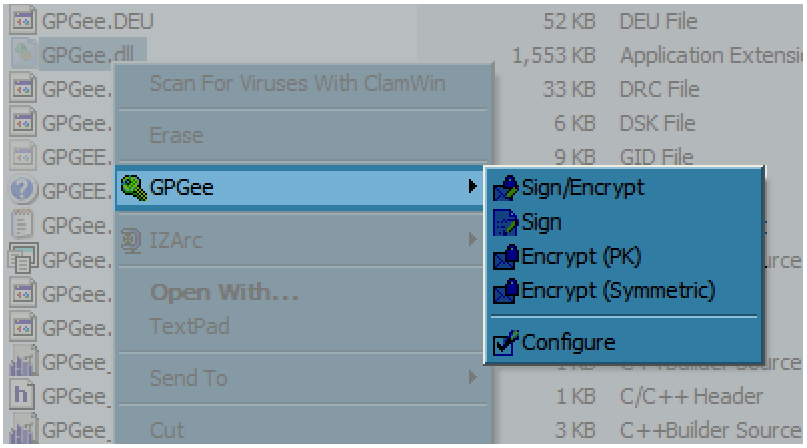
Introduction

GPGee - GNU Privacy Guard Explorer Extension v1.3.0

GNU Privacy Guard, or GPG, is the premiere open source implementation of OpenPGP encryption.   It is secure, free and open, versatile, and about as user friendly as toxic waste.   Simply counting all the command-line switches it supports would is a daunting task.   This is where front-ends come in generally, and where GPGee comes in specificially.

GPGee isn't a full front-end for GPG, rather it is a Windows explorer shell extension.   It is designed as a small helper to make day-to-day usage easier.   It registers itself as a context menu handler and gives you the ability to access GPG functionality right through the Windows file explorer.   Once it's installed, the GPGee menu will appear in the context menu that appears when you right-click on any file, as shown below:



GPGee Context Menu

The context menu that will appear for most files contains the following options:

- Sign and Encrypt

- Sign

- Encrypt

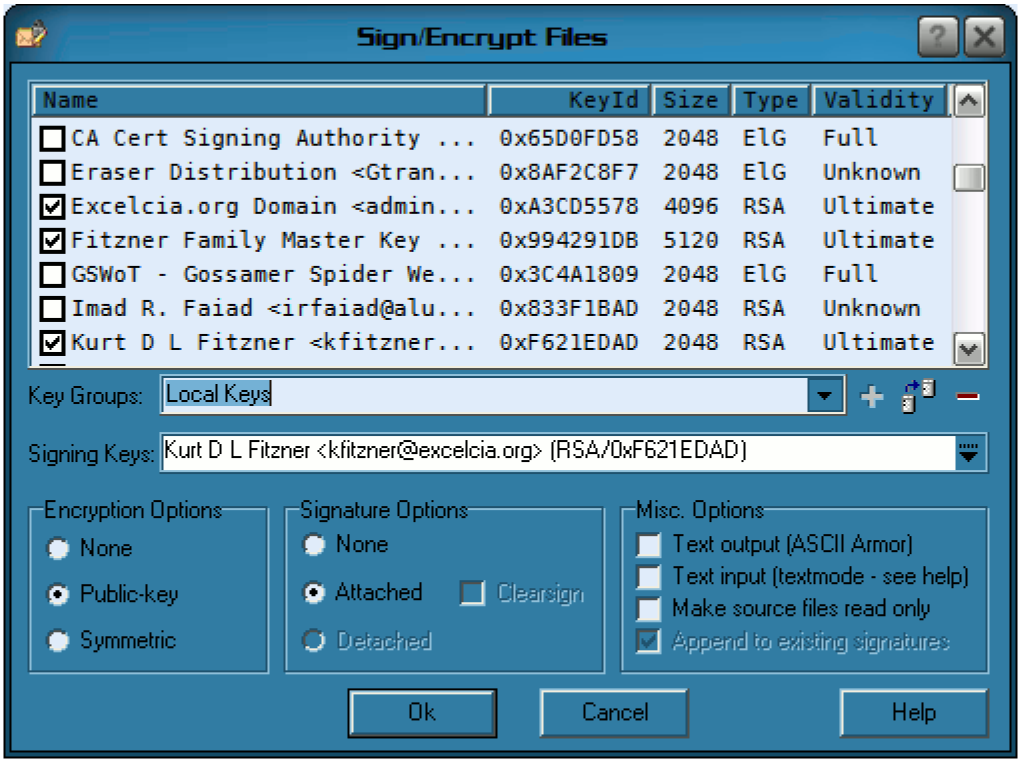
- Verify/Decrypt *

- Configure GPGee

* This option appears when only files of type ".asc", ".gpg", or ".sig" are selected in Windows explorer.

Sign/Encrypt Files

When signing and/or encrypting a file, GPGee will show the following dialog:



	Name	KeyId	Size	Type	Validity
□	CA Cert Signing Authority ...	0x65D0FD58	2048	ElG	Full
□	Eraser Distribution <Gtran...	0x8AF2C8F7	2048	ElG	Unknown
☑	Excelcia.org Domain <admin...	0xA3CD5578	4096	RSA	Ultimate
☑	Fitzner Family Master Key ...	0x994291DB	5120	RSA	Ultimate
□	GSWoT - Gossamer Spider We...	0x3C4A1809	2048	ElG	Full
□	Imad R. Faiad <irfaiad@alu...	0x833F1BAD	2048	RSA	Unknown
☑	Kurt D L Fitzner <kfitzner...	0xF621EDAD	2048	RSA	Ultimate

Key Groups: Local Keys

Signing Keys: Kurt D L Fitzner <kfitzner@excelcia.org> (RSA/0xF621EDAD)

Encryption Options
○ None
● Public-key
○ Symmetric

Signature Options
○ None
● Attached □ Clearsign
○ Detached

Misc. Options
□ Text output (ASCII Armor)
□ Text input (textmode - see help)
□ Make source files read only
☑ Append to existing signatures

GPGee Sign/Encrypt Dialog

The top of the dialog contains a list of all the public keys on your keyring and can be sorted by any of its columns by clicking the column heading. This list is used to select recipients' keys that you can use to encrypt a file to. Each recipient key in the list has a checkbox beside it. Select one or more recipient keys to encrypt the file with by clicking on the check boxes. Once encrypted, any one of the recipients will be able to decrypt the file.

You can create groups of keys for quick encryption to groups of recipients. To create a group, select all the keys that you want to have be in the group, then in the "Key Groups" drop-down edit box, type the name you wish to store this group under. Beside the drop-down edit box are three buttons, one each to add (+), modify ( ), and delete (−) a key group. Click the "Add" button once you have selected the keys and entered the name to create the new group. In the future you can then select that group from the "Key Groups" drop-down box to quickly select all the keys in that group. You can later modify the group simply by selecting it, selecting and unselecting keys in the public key list as needed for the group, and clicking the "modify" button. Delete a group by selecting the group then clicking the "delete" button.

Under the public key list and key group selection is a drop-down list of all the secret keys on your keyring. This list is used to select the keys to use for signing a file when "Signature options" is ***not*** set to "None". You can select one or even several keys from this list by activating the dropdown box and clicking the checkboxes on the shown keys. GPGee will remember the last keys used to sign with and those keys will be the default keys selected the next time.

Encryption Options

- **None**: No encryption. This is the default encryption setting when "Sign" is selected from the context menu.

- **Public key**: The public keys on your keyring are used to encrypt the file(s). One or more public keys can be selected from the public key list. This is the default encryption setting when either "Sign and Encrypt" or "Encrypt" is selected from the context menu. When "Encrypt to self" is enabled in the GPGee configuration and when you are signing+encrypting a file, then in addition to the keys in the public key list, the key used to sign the file will also be added as an key to encrypt to.

- **Symmetric**: Symmetric encryption is performed on the file(s). A passphrase is needed for encryption. The same passphrase must be known by whomever wishes to decrypt the file. You do not use recipients' public keys to encrypt with this option, and the public key list will be innactive if it is chosen. If symmetric encryption is chosen, you will not be able to select any signature options as it is not normally appropriate to sign a symmetrically encrypted file.

Signature Options

- **None**: No signature. This is the default signature setting when "Encrypt" is selected from the context menu.

- **Attached** Each target file is signed and the resulting signature is attached as added data to each signed file. The signature is normally added as binary data to the file. For text files, the "Clearsign" option can also be selected. This instructs GPGee to add the signature as text to the file so as to leave the file human-readable. Only text files should be signed in this manner. Since encrypted files are always compressed before encryption and are thus always binary data, the "Clearsign" option is not available if you are also encrypting in the same operation. The "Attached" option is set by default when "Sign and encrypt" is selected from the context menu.

- **Detached** Each target file is signed and the resulting signature is saved as a separate file from the file being signed. This is useful when you wish to publish a file and don't know whether those that will obtain it will all have software to be able to extract the original unsigned file. In this case you publish the signed file in its original state and also publish a separate signature for those that wish to verify the signed file's authenticity. This is by far the most common reason for signing a file. As such, this is the default signature option when "Sign" is selected from the context menu.

Misc Options

- **Text output (ASCII Armor)**: This option instructs GPGee to produce a file that is comprised of ASCII characters only. This is useful for if the resulting file is to be included as text in an e-mail or usenet news group posting. If this option is selected, the resulting file produced from the operation will have the .asc extension added to its name to indicate it is an ASCII file. If this option is not selected, the resulting file will have the extension .gpg added to its name to indicate it is a binary GPG file.

- **Text input (textmode)**: This setting is the same as using the --textmode switch on GnuPG's command line. It tells GnuPG that all input files for this operation are text files. This sets a flag in the output file that allows GnuPG on the recipient's computer to convert the file to the local text format when it is decoded, or to ignore line-ending differences during signature verification.

- **Make source files read-only**: Some software will alter the contents of a file even when the file is simply read and not saved. Microsoft Excel is a notable example where it will change a spreadsheet file to reflect the date it was last accessed. This is something you probably don't want to have happen after a file is signed, as the act of reading the file will change it and invalidate the signature. This option will cause GPGee to mark all the files it processes during the current operation as read-only to prevent their being changed inadvertantly after signing.

- **Append to existing signatures**: There may be times when you want to sign a file with more than one key, but where the signers use different computers. GPGee can take a file that already has a detached signature and append another signature to it. This way one person can sign the file, pass along the file and signature, and another person can add a signature to it. The checkbox for this is a tri-state - it has three settings. Unchecked means never append - GPGee will silently overwrite any existing signatures. Dimly checked or greyed-checked (the default) tell GPGee to ask each time it sees an existing signature. Checked means that GPGee will always append without asking. Signature appending is only possible when making detached signatures.

Ok Button

If the dialog's Ok button is greyed out and cannot be clicked, it means that GPGee doesn't have enough information to proceed. For example, if you have selected public-key encryption but have not yet selected any recipients from the public key list.

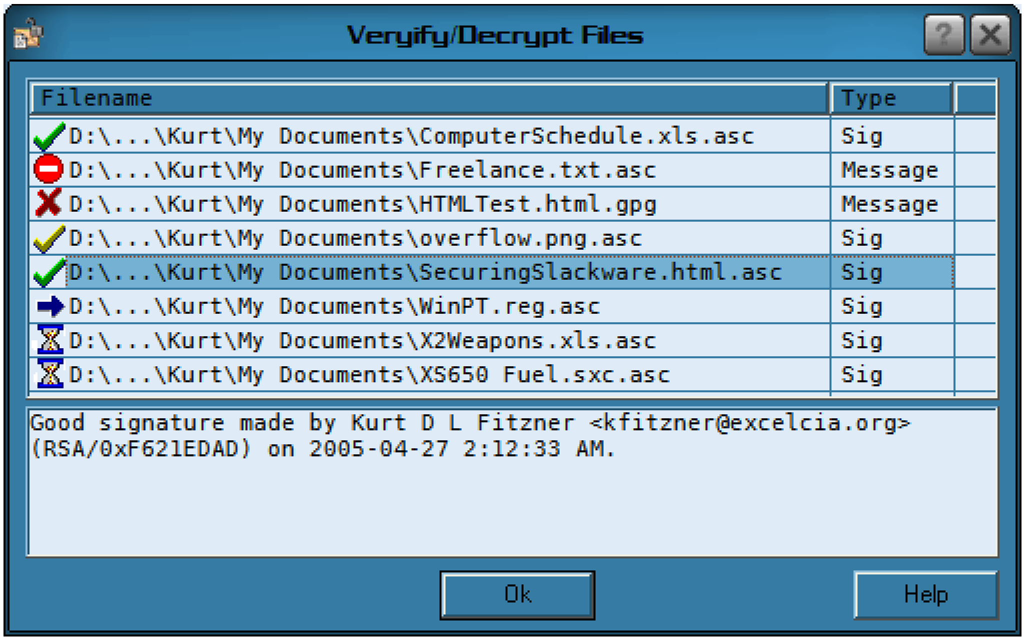
Passphrase

Once you have clicked Ok, if you are signing a file or if you are using symmetrical encryption, then you will be prompted for a passphrase. In the case of signing, the passphrase is to unlock the secret key that you will be signing with. The data for the secret key you have selected will be displayed on the dialog that asks for your passphrase. In the case of symmetrical encryption, the passphrase you select becomes the key for the encrypted file. In this case, the exact passphrase you enter at the time of encryption will be needed again by whomever wishes to decrypt the file.

Verify/Decrypt Files

GPGee's file verification and decryption is mostly an automated process. Whenever files that have an extension of ".asc", ".sig", or ".gpg" are selected in Windows explorer, the GPGee context menu will show the option for verifying/decrypting files.

When activated, a dialog will appear that looks as follows:



Veryify/Decrypt Files

Filename	Type
✓ D:\...\Kurt\My Documents\ComputerSchedule.xls.asc	Sig
⛔ D:\...\Kurt\My Documents\Freelance.txt.asc	Message
✗ D:\...\Kurt\My Documents\HTMLTest.html.gpg	Message
✓ D:\...\Kurt\My Documents\overflow.png.asc	Sig
✓ D:\...\Kurt\My Documents\SecuringSlackware.html.asc	Sig
➡ D:\...\Kurt\My Documents\WinPT.reg.asc	Sig
⌛ D:\...\Kurt\My Documents\X2Weapons.xls.asc	Sig
⌛ D:\...\Kurt\My Documents\XS650 Fuel.sxc.asc	Sig

```
Good signature made by Kurt D L Fitzner <kfitzner@excelcia.org>
(RSA/0xF621EDAD) on 2005-04-27 2:12:33 AM.
```

GPGee Verify/Decrypt Dialog

GPGee automatically determines what type of GnuPG file each file is and will verify or decrypt each one as appropriate. When needed, GPGee will prompt you for a passphrase to decrypt symmetrically-encrypted files or to unlock your secret key for public-key encrypted files. Status icons displayed beside each file give a quick reference to what GPGee is doing and what the results were. At the bottom of the dialog is a memo area that will contain text to explain what GPGee has done for each file; status or signature information for successfully processed files, or error explanations for files that had problems.

The following are all the possible status icons:

⌛
GPGee has not yet started working on this file

This file is currently being processed

⛔
GPGee did not attempt to process this file. An explanation as to why will be in the memo box.

✓
The operation was completed successfully - extra information on what was done and the results will be displayed in the memo box.

✓
The operation completed successfully, but there were possible irregularities or warnings that should be looked at. For example, successfully verifying a signature produced with an expired or revoked key.

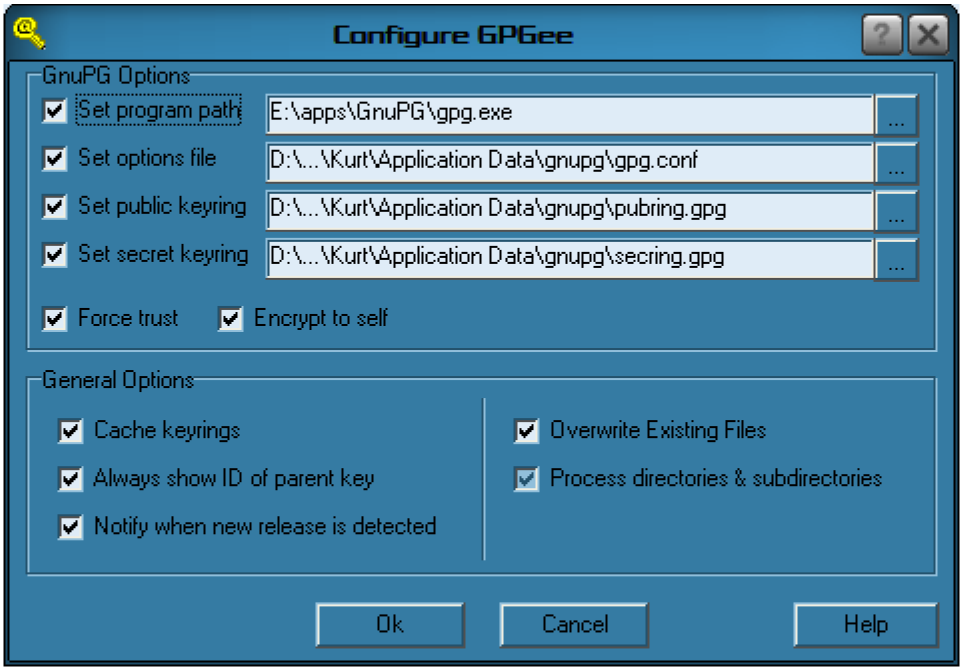
✗
The operation was attempted but failed. Possible reasons include a bad password or an invalid signature.

Any time a passphrase is needed, GPGee will prompt you for it. In any single set of operations, each passphrase that you enter will be cached internally in memory that is protected from being written to the Windows swap file.

Passphrases are never cached longer than the current operation however, so grouping as many files together to process at once is the most efficient way to keep from having to enter your passphrase multiple times.   After the operation is finished, all passphrases are erased from memory before that memory is freed for use by other programs.

Configure

GPGee's configuration dialog is activated from the GPGee context menu and looks as shown below:



GnuPG Options

☑ Set program path: E:\apps\GnuPG\gpg.exe

☑ Set options file: D:\...\Kurt\Application Data\gnupg\gpg.conf

☑ Set public keyring: D:\...\Kurt\Application Data\gnupg\pubring.gpg

☑ Set secret keyring: D:\...\Kurt\Application Data\gnupg\secring.gpg

☑ Force trust ☑ Encrypt to self

General Options

☑ Cache keyrings

☑ Always show ID of parent key

☑ Notify when new release is detected

☑ Overwrite Existing Files

☑ Process directories & subdirectories

GPGee Configuration Dialog

GPG Options

There are four individual paths that can be set in the GPG Options to override your computer's default GPG settings. Under normal circumstances, GPGee will locate all these files automatically. If, however, you have unique needs, you can choose any or all of the below options:

- **Set program path**: Enables setting a specific location for the GPG executable on your computer.

- **Set options file**: Enables setting a specific location for the GPG options file (normally called gpg.conf). Setting a specific location for this file within GPGee can allow you to have one options file for normal usage, and a different options file for using from within GPGee.

- **Set keyring**: Enables setting a specific location for your public key keyring (normally called pubring.gpg).

- **Set secret keyring**: Enables setting a specific location for your secret key keyring (normally called secring.gpg).

Once any of the above check boxes are activated, GPGee will display a file requester dialog that you can use to select the location of the applicable file. To reselect the file, click on the button to the right of the file name field which is labelled with an elipses (three dots ...).

- **Force trust**: Force GPG to allow encryption to any and all public keys on your keyring regardless of what the trust setting is on that key.

- **Encrypt to self**: When a file is being signed+encrypted, this option will cause the key used for signing the file to be added as an encryption recipient. This allows the user signing+encrypting the file to be able to later decrypt and access the file again. In the case of multiple subkeys, the first subkey found that is usable for encryption is the one that is used.

General Options

- **Cache Keyrings**: If GPGee takes a long time to load your keyrings, then you can turn on key caching. Before you can turn it on, however, you must first manually specify the location of both your public and secret keyrings (in the GPG Options section) since GPGee uses the timestamp on your keyrings to tell when changes have been made. Once key caching is enabled, you will only experience a delay the first time GPGee loads a keyring. After that GPGee will use its cache until it detects a change in the keyring files.

- **Always show ID of parent key**: Some GnuPG keys have a master or parent key (usually the signing key) and one or more subkeys (usually the encryption key). When this setting is checked, GPGee will always show the key ID of the master or parent key, even if a sub key is being used. Most people will probably want to leave this checked.

- **Notify when new release is detected**: This setting causes GPGee to check each time it is activated (maximum one check per day) to see if there is a new release available. No information is sent during this check. The check is performed by retrieving the text file at URL http://gpgee.excelcia.org/GPGeeCurrentVersion.txt checking its contents against GPGee's internal version string. When you are notified of an update, you will be given the option at that time of telling GPGee not to notify you again until the *next* new release is detected. This allows you to stop GPGee from continuing to notify you about a minor release that you dedided not to upgrade to.

- **Overwrite Existing Files**: During decryption and verification operations, it is possible that the output file for the operation could already exist. What GPGee does in that case depends on this setting. When the box is unchecked, GPGee will refuse to overwrite files. If the box is checked, GPGee will always overwrite files. If the box is checked but grayed, GPGee will ask each whether or not to overwrite a file.

- **Process Directories and Subdirectories**: This setting affects how GPGee responds to directories that are in the list of files that were right-clicked on. When this option is checked, the GPGee will recurse through any directories it finds and process the files and subdirectories within them. When the option is unchecked, GPGee will ignore directories (and the files in them). When this option is checked but grayed, GPGee will ask you if it should process directories if it encounters one.

Changelog

GPGee version 1.3.1 build 192 (20 Apr 2006)

- Bugfix: The sign/encrypt form wasn't correctly scaling to non-default Windows DPI settings.

- Bugfix: Certain very large (large as in many signatures, user ids, subkeys, etc) keys would cause GPGee to hang when a signature was verified. Caused by GnuPG providing duplicate keys under certain conditions. GnuPG's maintainers won't fix the problem, so a partian workaround was incorporated in GPGee's back end library. If there are any irregularities in the display of keys (certain keys missing), please report this.

- Bugfix: The installer was copying the fonts to the correct font folder, but not registering them. This caused, for example, the encryption recipient list to be blank if GPGee was run after installation until a reboot was performed.

- Bugfix: Files with more than 13 signatures would cause GPGee to hang if they were verified.

- Changed the order of the encryption options and signature options in the sign/encrypt form. It was confusign some people that the encryption options were first when it's called "sign/encrypt".

- Bugfix: Fixed typo in the German language file (FORM_CONFIG_CHECK_PROGRAMPATH).

- Added "Encryption Key Selection" label to the encryption key selection control.

GPGee version 1.3.0 build 179 (13 Jan 2006)

- Added OpenPGP smartcard support. Thanks to g10 code for donating the smartcards! These are very cool - they eliminate 99% of the ways to attack a key because the key never leaves the card!

- GPGee is now subkey-friendly. Previously, even if you had a subkey, GPGee would force GnuPG to sign with your main key. Now GPGee allows GnuPG to select which subkey is used. This is useful if you have an existing RSA key that you want to add subkeys to later (for example, if you get a smartcard and make subkeys on the card).

- Added support for the .pgp extension. This is based on how PGP verion 6.5.8 used the extension. If there are any problems with more modern versions, please contact me.

- The sign/encrypt window is now sizeable.

- The private keys in the sign/encrypt window are now alphabetically sorted.

- The file list in the verify/decrypt window will now scroll to show the file currently being worked on.

- Changed the configuration method for overwriting from "yes/no/ask" radio buttons to a tri-state check box (yes=checked, no=unchecked, ask=checked but grayed). This is more consistant with the signature append checkbox in the sign/encrypt form, and the way all future "yes/no/ask" type settings will be done.

- Added a configuration entry to allow the user to disable directory recursing, or to force GPGee to ask.

- Changed the way directory recursion works. Previously it would process all the directories at the time the user right-clicked. If a directory with lots of subdirectories was right-clicked on, it could take several minutes for the context menu to appear, simpy because GPGee was reading every filename. This was unacceptable behavior, so now directories are processed when GPGee is actually selected and the user clicks "Ok".

- Changed the way a cancel works for the verify/decrypt passphrase dialog. It used to cancel just the passphrase and processing of other files would continue. It now causes all the queued operations to be cancelled as well.

- Bugfix: When a file was signed and encrypted, the signature result message was overwriting the decryption result message. Now it is appended and both are displayed.

- Bugfix: Fixed a bug where encrypt-to-self may not have worked properly for all files processed in one operation.

- Bugfix: Fixed a bug where GPGee's symmetrical encryption always used the CAST5 algorithm, regardless of what the gpg.conf file specified. This was actually two bugs - GPGee's backend library (MyGPGME) had a bug where you could not clear the default algorithm, and GPGee itself had a bug where it didn't even try.

- Bugfix: Fixed a bug in passphrase caching where, when signing files with more than one key, GPGee would not consult the cache after the first passphrase for a file.

- Bugfix: Fixed a bug in passphrase caching where a mistyped passphrase would be entered into the cache and referenced instead of a corrected one. If I haven't mentioned this before, I **hate** passphrase caching. It's something that *sounds* simple but, in practice, is very tricky to do right.

- Merged most of Timo Shulz's changes and bugfixes to the MyGPGME library made since May last year. Timo isn't working on MyGPGME any more, so I have decided to support it myself. Some of his last changes cleaned up some memory leaks, so hopefully this will help GPGee.

GPGee version 1.2.3 build 146 (12 Dec 2005)

- Bugfix: Fixed a bug that was introduced in the last release that prevented the configure menu from appearing when a non-gpg file was right-clicked.

- Bugfix: The help file incorrectly identified the 1.2.2 release as a debug release, which it wasn't, though it perhaps should have been.

GPGee version 1.2.2 build 144 (11 Dec 2005)

- GPGee will now (finally) recurse any directories in your Windows Explorer file selection.

- Added a new menu entry for symmetrical encryption and renamed the old one to "Encrypt (PK)" to differentiate it. The sign/encrypt dialog will immediately proceed to asking for a passphrase when this option is selected, since you don't need to enter in any encryption or signing keys for symmetrical encryption.

- There shouldn't be as much delay for the context menu to appear when large numbers of files are selected.

- The Ok button on the sign/encrypt form is now the default - which means when the sign/encrypt form comes up, you can just hit enter for Ok.

- Changed GPGee's memory manager to FastMM. This should speed things up in general, as it really is lightning fast when compared to Borland's memory management junk. There have been reports of GPGee leaking memory. While Windows Explorer certainly leaks memory on my machine, I can't trace this to GPGee - Explorer leaks memory whether I have GPGee attached or not. The new memory manager is rock solid, so it might help.

- Bugfix: The installer had a problem with upgrading old versions - it would not properly detect when a DLL was in use and prompt for a reboot. It will now do this properly. This was actually previously released as 1.2.1a since there was no change to the underlying code.

- Internal change to the way strings are handled for visual form elements like labels or the column headers in string tables. These strings are also now stored in a resource file and loaded at runtime. This is to make translation easier - there is only one file with strings that need translating.

- Thanks to Werner Koch we now have a German translation. GPGee should automagically load the german language module (GPGee.DEU) now.

GPGee version 1.2.1 build 115 (8 Sep 2005)

- Bugfix: The automated version checking had more problems. It seems the ISP I use cached member web pages. Because the version check is a simple string match, and because the caching meant that everyone was still downloading the old version string, people were getting messages about upgrading to a previous version. **sigh** The URL is now changed to my personal web server and the version check now looks at the build number so it will never tell you to "upgrade" to an earlier release. This was compounded by the next, related bug...

- Bugfix: Version checking now only really happens once a day, not every time GPGee is activated. No, really, I mean it this time. Honest.

- Bugfix: Hang error during sign+encrypt operations fixed. The new passphrase system for sign+encrypt was being interrupted by a gpgme_set_passphrase() call that was part of the old system. More poor testing on my part.

- Added support for signature appending. In the same vein as signing a file with multiple keys, this feature allows you to append a signature to an existing one. This means that multiple people on multiple computers can sign a file.

- Some further source code documenting and prettying up. I find that I catch bugs when I "comb the code". I guess it's sort of like finding all the gnats and lice in my hair when I comb it.

- Bugfix: Some further internal passphrase securing. It's not likely that the issues changed here are security problems, but it's better to be safe than sorry later.

GPGee version 1.2.0 build 107 (6 Sep 2005)

- Bugfix: Fixed a bug that probably prevented encrypt-to-self from ever working. Whoops. I think the MyGPGME library might automatically do that in any case, which means the options in GPGee is useless. One of these days I'll investigate that further.

- Added support for verification of multi-key signatures.

- Added support for creation of signatures with more than one signing key. This also necesitated a change to Timo's MyGPGME library.

- Added support for making the source files read-only after a sign/encrypt operation. Useful for when a file's "reader" wants to update the file when it is read. This is done by Excel, for example - any loaded spreadsheet is modified with the last accessed timestamp. This isn't particularly good when a file has been signed, because the act of reading the file then invalidates the signature.

- Upgraded the error reporting for sign/encrypt. Previously many errors might have been logged, but no notification messagebox was given to the user.

- Bugfix: Fixed the "Could not convert variant of type (string) into type (Double)" bug that was introduced in 1.1.3. I have to admit, this bug was due to bad testing on my part.

GPGee version 1.1.3 build 100 (31 Aug 2005)

- Added support for automatic version checking. Now that GPGee downloads are taking off, we might need that. I couldn't justtify having it turned on by default, so to encourage people to use it, I also make a very visible option in the installer to turn it on.

- Bugfix: A bug in the help file makes the text black-on-black for some color schemes. I *think* I have fixed this, but email back to the person who reported the bug bounced so I don't know for sure.

- Minor change in licensing. Whereas before GPGee was licensed under version 2 "or any later version" of the GPL, now it is locked into version 2. Call this the paranoid in me, but I found I was comfortable with version 2 and not comfortable with someone other than me being able to change the licensing terms of my software sometime in the future. If version 3 comes along and it's better, then I'll change the terms myself then.

GPGee version 1.1.2 build 94 (29 July 2005)

- Passphrase caching has been upgraded. Before, you would never be asked for the same passphrase twice in a row when verifying/decrypting multiple files, but if you were asked for a passphrase for key A, then for key B, if it was needed again for key A you would be asked for it. In other words, it only cached the most recently asked for passphrase. The new way caches all passphrases asked for within a single operation. This caching does not persist past that operation, though. In addition, it has been made safer by ensuring all memory that passphrases are stored in is now locked so that Windows won't page it out to virtual memory. After an operation, each passphrase is overwritten before it is freed. In this way the hope is to keep passphrases from ever floating around in unallocated memory or getting written to the hard drive.

- Passphrase overwriting has been strengthened. Previously the memory where a passphrase was stored was overwritten, but someone looking at it after the fact would be able to tell how many characters the passphrase was. Now, after the passphrase is overwritten once, the length is increased to a fixed size (128 bytes) and it is overwritten again. It's doubtful anyone has a passphrase more than 128 characters, so this seems to me to be a safe size.

- Bugfix: A bug where the passphrase from the passphrase dialog was not overwritten after the dialog was freed has been fixed.

- Bugfix: The new "Always show ID of parent key" setting was inadvertantly set as a read-only setting in the compiled-in registry configuration file. The "read-only" column was set to true instead of the default column. This causes GPGee to throw an exception when someone uses the configuration dialog.

GPGee version 1.1.1 build 89 (17 July 2005)

- Added an option to display encryption keys with the ID of the parent key and made this the default. Prior to this, encryption subkeys were shown with their own key ID which confused some people. Anyone wanting the old behavior will have to enter the configuration and change the "Always show IF of parent key" setting.

- Added some code that should speed up the initial display of the sign/encrypt dialog when there are lots of keys.

GPGee version 1.1.0 build 86 (17 May 2005)

- The message digest algorithm is now displayed when a signature is verified.

- The key caches can now be held resident. There were reports from some people with very large keyrings of delays of up to fifteen seconds while GPGee loaded the keyrings. If you specify the location of the keyring files, GPGee can be set to keep the cache in memory and reuse it until the keyring changes.

- Bugfix: The workaround used to circumvent the GPGME duplicate key bug didn't work in all cases. A better one is now in place that should do the trick.

- Bugfix: Clearsigned messages weren't being verified correctly.

- Bugfix: Bad signature reports no longer attempt to print the non-existant signature date.

- Bugfix: Not a bug per se - an old "#pragma link" was left lying around for a library that I tested with GPGee briefly. It was linking in code that wasn't needed. Removing it saved 200k off the size of the .dll. Yay.

GPGee version 1.0.0 build 74 (11 May 2005)

GPGee version 1.0.0D build 75 (11 May 2005) (Debug/logging version)

- Acknowledgement: I want to add a special thank-you to Gerfried "moali" Maier. He did a lot of testing for me and was instrumental in quashing many of the bugs below. Thanks!

- Added key groups - you can now create quick-select groups of keys for encryption.

- Bitmaps associated with each context menu item for better visual association.

- Logging (in debug builds) now uses high resolution performance counters on CPUs that have it. Log entry timestamp resolution is now accurate to the millisecond.

- Bugfix: Several minor memory leaks plugged.

- Bugfix: GPGee context menu no longer appears when only directories are selected.

- Bugfix: In the verify/decrypt form, the secret key cache was being built before the GPGee configuration settings on gpg.exe executable and keyring locations were being set.

- Bugfix: There is a bug in the MyGPGME library where keys appear twice in a keycache. A workaround to prevent GPGee from listing the same key twice has been added until this bug is fixed in the library.

- Bugfix: The changes made (in version 0.3.0) to Timo Schulz's MyGPGME library to allow manually setting the gpg.exe/options/keyring paths were simply not workable. I had attempted to allow setting them on a per-context basis. There are just too many times in the library, though, where a new context is created out of thin air to do something. Every time a gpg command was executed by the library with this new context you lose the paths settings. I've now changed this to a global mechanism where the paths are set for the whole library. This should fix all the times where a user would manually set a path in GPGee's configuration dialog and it wasn't being honored.

- Bugfix: An exception raised while GPGee is in verify/decrypt mode will no longer cause the "Cancel" button to be unresponsive.

GPGee version 0.4.1 build 51 (6 May 2005)
GPGee version 0.4.1D build 52 (6 May 2005) (Debug/logging version)

- When verifying a signature produced with an expired key, the date of the signature is now checked against the expiry date of the key. If it appears the signature was produced after the key expired then the user is shown a warning to this effect. Previously GPGee would only say "Good expired-key signature..." and leave it to the user to do this checking.

- A key's validity is now taken into account when verifying signatures. Signatures produced with keys marked with undefined or never validities now cause a warning message and the result "checkmark" will be yellow. Signatures made with marginal keys produce the warning message, but the result checkmark is green (I felt marginal deserved a warning, but marginal means barely which is still good so no yellow checkmark).

- Extensive logging added (only active when a logging-enabled debug build is created)

- Bugfix: Passphrases were not being correctly wiped after use. This means it was more (though still not very) likely that a passphrase could have ended up stored on a page file.

- Bugfix: The context-sensitive for the configuration dialog is now updated with the new 0.4 options.

- Bugfix: Exceptions now propagate properly and end with a messagebox to the user giving the error message. Before, certain exceptions would cause Explorer to crash.

- Source: A whole bunch of extra comments have been added to the source with better theory-of-operation explanations.

- Source: All message strings are now encapsulated inside string table resources to allow for translations to other languages. All we need now are translators.

GPGee version 0.4.0 build 19 (27 April 2005)

- Bugfix: Unchecking a file-location option on the configuration dialog failed to clear that option when the user clicked Ok.

- The last used signature key is now saved to be the default for the next time a file is signed.

- Symmetric encryption now working - before it would ask for the passphrase but not actually encrypt. Whoops.

- Added unique icons to each dialog to show the dialog's function. Changing the function of the sign/encrypt dialog (sign, encrypt, sign+encrypt) changes the icon. Useless, but makes it look a tiny bit more professional.

- Bugfix: Several unlikely but theoretically possible error conditions during initialization of the GPGME library and key caches could have caused a crash.

- Added a configuration option for Encrypt to Self. When this option is selected, any file that is signed and encrypted will add the key of the signer to the list of encryption recipients.

- Added file verification and decrypting. The process is mostly automated, only asking for user input when a passphrase is needed.

- Added a configuration option for overwriting files. You can choose during decryption to always, never, or prompt to overwrite files.

- All dates now converted from GMT to the local time zone, taking into account daylight savings (if present).

GPGee version 0.3.1 build 15 (12 April 2005) - Bug fix release

- Bugfix: GPGee was not honoring the Force Trust configuration setting.

GPGee version 0.3.0 build 13 (1 April 2005)

- Bugfix: Opening a dialog more than once resulted in an unitialized dialog.

- Added support to the MyGPGME library for configuring the location of gpg.exe, gpg.conf, and the keyring files.

- Added a configuration dialog.

- Added context-sensitive help.

GPGee version 0.2.0 build 5 (21 March 2005)

- Initial public release (0.1 was a dismal failure - it was quickly put out of my misery). Support for signing/encypting. Lots of bugs.

License

This software is released under the terms of version 2 of the GNU General Public License:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this

section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS