# Pretty Good Privacy Windows Shell

## *Version 2.1*

## *User's Guide*

Prepared for:

Project Manager, STINGER Product Office
Attn: David C. Kennedy, P.E.
SFAE-MSL-FAD-SE
Redstone Arsenal, AL 35898
Commercial: (205) 876-2282
DSN: 746-2282

Prepared by:

Mike Lyman
AEgis Research Corporation
6703 Odyssey Dr., Suite 200
Huntsville, AL 35806
Phone: (205) 922-0802

# Table of Contents

# Introduction

## Overview

I designed Pretty Good Privacy Windows Shell to help users encrypt and decrypt their e-mail. It relies heavily on the Windows clipboard for transferring messages into and out of the shell. Since there is a 32k limitation to the Visual Basic text box, file encryption and decryption has also been added so that you can handle larger messages. File encryption and decryption is also useful for other purposes.

Pretty Good Privacy (PGP) is a freeware encryption/decryption program available on the Internet. It uses both public and secret keys to encrypt and decrypt text messages. Messages are encrypted to specific UserIDs using their public keys and only those specific people can decrypt the messages.

This package is actually two programs interacting with each other to get your work done. The Pretty Good Privacy Windows Shell makes things easy for you and then has Pretty Good Privacy do the hard work.

The Pretty Good Privacy Windows Shell was designed to work with version 2.6.2 of PGP. There were some reported problems with other versions of PGP that have been fixed but there may be other problems. If you encounter problems, try to get version 2.6.2.

**Pretty Good Privacy Windows Shell (The Windows program PGPSHELL.EXE)**

Pretty Good Privacy is not a Windows program and must be run from the DOS prompt with command line switches. This is inconvenient for most users and not something they want to do.

I developed the Pretty Good Privacy Windows Shell to make Pretty Good Privacy easier to use. It provides you push button control and guides you through the steps needed to make Pretty Good Privacy work. This shell still needs to go to DOS to make Pretty Good Privacy work but the shell does this for you. The Pretty Good Privacy Windows Shell will start a DOS window and execute the commands that you requested when you clicked on a button. You will occasionally need to type things while the DOS window is opened but Pretty Good Privacy prompts you for what it needs and you do not have to remember anything (except for your pass phrase to decrypt messages to you.)

**Pretty Good Privacy (The MS-DOS program PGP.EXE)**

Pretty Good Privacy does the hard work in this team but it does it as a DOS program rather than as a Windows program.

You can go to DOS yourself and use Pretty Good Privacy directly but this manual does not cover that. If you want to use Pretty Good Privacy directly, read information in README.DOC, PGPDOC1.TXT and PGPDOC2.TXT that come with PGP. You can also access the PGP.HLP file that comes with this shell. It is a Win 3.1 help file containing the exact same information as the PGP documentation.

## User's Guide Purpose

This manual only describes the Pretty Good Privacy Window Shell interface, how to use it and how the shell interacts with Pretty Good Privacy. It does not describe the Pretty Good Privacy program or its features. It assumes you are familiar with Microsoft Windows and the Windows interface.

## Finding PGP on the Internet

Finding Pretty Good Privacy on the Internet is not as straight forward as finding most files at FTP sites. You can start to find pgp262.zip at **ftp.csn.net** in the **/mpj** directory. Directions from there are contained in the README file. (The path to Pretty Good Privacy changes every thirty minutes. If you cannot find the indicated path, it may

have changed in the time it took you to read the README. This happened to me and I had to reread the updated message.)

There are also various other sites distributing Pretty Good Privacy and you can find them in several of the computer magazines. The other sites have different ways of protecting Pretty Good Privacy so be prepared for surprises.

## What's New in Version 2.1

**International users will no longer get TYPE MISMATCH error**

In version 2.0, international users experienced problems caused by the differences in decimal points used. The U.S. uses a "." while many international number systems use a "," for decimal points. This error has been corrected

**Signing Keys "-ks"**

You asked for it, you got it. You can now sign keys in your public keyring.

**Multiple UserIDs for a key**

Again, you asked for it and you got it. For people who use multiple UserIDs with their keys, those multiple UserIDs are now visible. (To see them, use the Rebuild UserID List button on the Admin Tab.) Be careful, this makes some key management tasks a little more difficult. This shell will take care of its part but be careful of PGP's prompts in the DOS Window.

## What Was New in Version 2.0

**Obvious Stuff**

***File Encryption and Decryption***

You can encrypt and decrypt files. No more 32k, text box limitation.

***Binary Armor***

Binary Armor is now supported

***Single Key Encryption/Decryption "-c" Option***

You can encrypt files using conventional, single key encryption.

***ASCII Wrapper Only***

You can now use PGP Windows Shell to provide an ASCII wrapper (without encryption) to binary files so you can send them by e-mail to other PGP users. This is similar to UUENCODE.

***Signatures***

PGP Windows Shell now supports signed messages.

***Clearsig***

Clearsig messages and files are now supported.

***More Key Management***

You can now perform more key management from the shell rather than the DOS prompt.

### *Clipboard Monitor*

You can have PGP Windows Shell monitor the clipboard to automatically decrypt PGP messages you copy to the clipboard.

### *Pausing Options*

You can now have the system pause after shelling to PGP. This allows you to see any output filenames, signature check messages, error messages, etc. PGP Windows Shell will pause when it needs to automatically. You can set the Shell to pause after adding keys and decrypting messages if you want it to. There is also an error checking pause option that will add a pause to every shell to PGP.

### *PGP Help File*

A Windows Help file version of PGP's documentation is now included. This is the same information included in PGPDOC1.TXT and PGPDOC2.TXT that come with PGP but compiled as a Windows 3.1 help file. Thank you Jeff Sheets for compiling this help file and allowing me to distribute it with this shell.

This file (PGP.HLP) is automatically installed with PGP Windows Shell but you can delete the file to save disk space. If you delete the help file, it will no longer show up on the help menu.

### Background Stuff

There are some changes that will not be apparent to some users.

### *Temporary Files*

In version 1.0, I forgot to wipe and delete the temporary files that the PGP Windows Shell used. Some people probably never realized this and people who were concerned probably did and took care of it themselves. The shell now wipes the plain text files. When encrypting a message, it uses PGP's -w options. When decrypting messages it rewrites the temp file with 64k of "X" and then deletes it.

### *VB Text box limitations*

The Visual Basic text box control is limited to 32k. For most cases this is the limit of the plain-text message since PGP compresses as it encrypts. Cases where this 32k limit is exceeded are now detected and the shell uses Write (WordPad) to display the results. In tests, I found a few cases where even Notepad could not hold the decrypted message.

### *Key References*

Keys were sent to PGP through the command line using the UserIDs in version 1. Since the command line has a limited length, PGP Windows Shell now uses the KeyID that goes with the UserID. The KeyID is shorter in length and means that more can be sent to PGP.

### *+nomanual option*

The +nomanual option is now used with key generation.

# How To...

## *Setup*

The setup program will try to setup to C:\PGPSHELL. The PGP Windows Shell executable is named PGPSHELL.EXE. There is a DOS base shell for PGP that is also named PGPSHELL.EXE so be careful if you also have this DOS shell.

**Setup from Download**

***Full Version***

The download version may come in many different flavors since I cannot control how people upload it to BBSs and on line services.

The download version I post comes as a self-extracting, zipped file. To unzip the setup routine, copy the pgpshlzp.exe to a temporary directory and run it. After the file unzips itself, follow the directions below except run setup from the temporary directory instead of the floppy drive.

***Upgrade***

The upgrade download I post comes as a self-extracting, zipped file. To install it, run PGPSHEL2.EXE and copy PGPSHELL.EXE over the old version and move PGP.HLP into the same directory. PGPSHEL2.EXE also includes this manual.

**Setup from Floppy**

***Installing Pretty Good Privacy Windows Shell***

Pretty Good Privacy Windows Shell comes with an automated setup program. To run the setup program put the setup disk in your 3½ floppy drive.

### Windows 3.1

From the Program Manager:

- Click on the File Menu and select the run command.
- Type the 3½ drive letter and setup.exe. ex: a:\setup.exe.

Or from the File Manager:

- Switch to the 3½ drive and double-click on setup.exe.

### Windows 95

- Start the Control Panel
- Double Click on Add/Remove Programs
- Select the Install/Uninstall tab
- Click on the Install button and the Install Wizard will start
- The Wizard will scan your disks for setup programs
- When the Wizard finds the setup.exe on the 3½ drive click on the Finish button

### Both

The setup program will run and ask you for the directory you want to install to. Accept the default or type in another directory.

The program will copy the necessary files to your hard disk. Some files will go into the directory you type in above, some will go to you WINDOWS\SYSTEMS directories.

The setup program will then create a program icon.

## Configuring Pretty Good Privacy Windows Shell

### *AUTOEXEC.BAT - Needed by PGP*

PGP requires special environmental variables to work. PGP Windows Shell does not, it will create a PGPSHELL.INI file to store the information it needs. If you have already configured you system for PGP, you can skip this section.

You can set an MS-DOS environment variable to let PGP know where to find its special files. Use your favorite text editor to add the following line to your AUTOEXEC.BAT file:

SET PGPPATH=C:\PGPSHELL

You must also add C:\PGPSHELL to your PATH statement in the AUTOEXEC.BAT file. Substitute your own **PGP** directory name if different from "C:\PGPSHELL". This environmental variable is needed by PGP and not by PGP Windows Shell.

Another environmental variable you should set in MS-DOS is "TZ", which tells MS-DOS what time zone you are in, which helps PGP create GMT timestamps for its keys and signatures. If you properly define TZ in AUTOEXEC.BAT, then MS-DOS gives you good GMT timestamps, and will handle daylight savings time adjustments for you. Here are some sample lines to insert into AUTOEXEC.BAT, depending on your time zone:

| | |
|---|---|
| For Los Angeles: | SET TZ=PST8PDT |
| For Denver: | SET TZ=MST7MDT |
| For Arizona: | SET TZ=MST7 (Arizona never uses daylight savings time) |
| For Chicago: | SET TZ=CST6CDT |
| For New York: | SET TZ=EST5EDT |
| For London: | SET TZ=GMT0BST |
| For Amsterdam: | SET TZ=MET-1DST |
| For Moscow: | SET TZ=MSK-3MSD |
| For Auckland: | SET TZ=NZT-13 |

Now reboot your system to run AUTOEXEC.BAT, which will set up PGPPATH and TZ for you.[1]

### *Pretty Good Privacy Windows Shell*

Upgrading from Version 1.0

PGP Windows Shell 2.0 needs to store some information differently than Version 1 did. The first time you run Version 2.0 it will rebuild your UserID list and then ask you for your UserID again.

The first time you run Pretty Good Privacy Windows Shell, it will attempt to configure itself. It will display a message box asking you if you want to configure it. You should answer yes. (The program will not work right if you do not.)

The first thing it will ask you is if PGP.EXE is in the same directory as the PGP Windows Shell. If it is, answer Yes and that is that. If you already had Pretty Good Privacy installed on you machine before installing the Windows Shell and did not install to the same directory, answer no and show the shell where Pretty Good Privacy is located. (If PGPShell cannot find PGP.EXE it will again ask you to help locate it.)

After finding PGP.EXE, the Windows Shell will ask you if you have generated your public and private keys yet. Answer yes or no. If you answer no, it will let you generate your keys. (See below)

[1] Taken from Pretty Good Privacy Version 2.6.2 Installation Guide by Perry Metzger, Colin Plumb, Derek Atkins, Jeffrey I. Schiller and others (SETUP.DOC)

After dealing with your keys, the Pretty Good Privacy Windows Shell will build your initial list of UserIDs that you have keys for. (If this is the first time you have used Pretty Good Privacy the list will probably only contain your UserID.)

After building the UserID list, the program will ask you to identify your UserID so that it can automatically encrypt messages so that you can decrypt them.

The program will then tell you that it is configured.

#### *Message Display Font*

You can change the message display font by clicking on the Options menu and selecting the Display Font command. The system will show you a font dialog box. Choose the font and the size you want and click OK. Your choice will be saved and used from now on.

#### PIF and Properties Settings

Since PGP is a DOS program, you may need to create PIF (Win 3.1x) or Properties (Win95) settings so that PGP works correctly.

#### *Windows 3.1x PIF Settings*

Windows 3.1x users - create a PIF for TEMP.BAT in the PGP Window Shell directory if you don't like the behavior of the shelled process. These settings should mirror your settings for PGP. Suggested settings are Memory allocation of -1, -1 (as much as the DOS session wants and can get), Windowed display (not full screen), Background and Exclusive, Close Window on Exit, Lock App Memory (under advanced options) (Thanks, Jack Trades)

#### *Windows 95 Properties Settings*

The Windows 95 defaults work fine but they leave DOS windows open after the batch process ends. Before closing the window, click on the Properties button and click on the Close Window On Exit option box. You can also set any other properties you want set for the TEMP.BAT. (the only other option I change is Font Size for personal taste) They will remain in effect even though TEMP.BAT is deleted once the window is closed. You will notice a MS DOS shortcut pointing to TEMP.BAT, this is where Win95 maintains non-default properties.

## *Using Pretty Good Privacy Windows Shell*

#### Starting Pretty Good Privacy Windows Shell

To start Pretty Good Privacy Windows Shell double click on the Pretty Good Privacy Windows Shell Icon.

#### Quitting Pretty Good Privacy Windows Shell

To quit Pretty Good Privacy Windows Shell you can use one of the following:

- click on the File menu and select the Exit command or,
- double click on the system menu box or,
- click on the sytem menu box and select the Close command or,
- press Alt + F4 or
- (Windows 95 only) click on the Window close button.

#### Encrypting a Message

Pretty Good Privacy encrypts messages so that only the person you are sending the message can decrypt it. You can encrypt a single message to multiple people at the same time rather than re-encrypting the message for each person.

You can also generate Clearsig messages. Clearsig messages are signed, plain-text messages. (Human readable messages that have an authentication/verification signature attached.)

The following sections will walk you through the encryption process.

Start by selecting the Encrypt a Message tab if you are not already there.

***Typing the Message***

You have a couple of options for generating your message:

- Type the message in the Step 1: Type or Paste Your Message Here: text box or,
- Type your message in another editor, copy the text to the Windows Clipboard and paste the message into the Step 1: Type or Paste Your Message Here: text box.

The text box will automatically wrap your text to the next line. Then Enter key will end the current line and move to the next line. The Tab key will not work.

***Selecting UserIDs to send the message to***

To encrypt a message you must specify who you are sending the message to. Only the people you send the message to can read the encrypted message. (Pretty Good Privacy Windows Shell will automatically encrypt any message to your UserID so that you can also decrypt the encrypted text. If this was not done, you would not be able to decrypt the messages you created.)

You do not have to select any UserIDs if you use the Clearsig button.

### Selecting a single UserID

To select a single UserID, find the UserID in the Select Recipients list box and click on it. The UserID that becomes highlighted is the one that the message will be encrypted to.

### Selecting multiple, continuous UserIDs

To select more than one UserID from the list, when all the UserIDs are continuous, you can:

- Click on the first UserID
- Hold down the Shift key and click on the last UserID

Or you can:

- Click on the first UserID and hold the mouse button down
- Drag the cursor over all the UserIDs you want
- Release the mouse button on the last UserID

### Selecting multiple, non-continuous UserIDs

To select more than one UserID from the list, when all the UserIDs are not continuous , you can:

- Hold the Ctrl key and click on each UserID you want

Or you can:

- Select a continous list of UserIDs as described about and
- Hold the Ctrl key and click on each UserID in the select list that you do not want

***Encrypting the message***

The Encrypt Message button will be disabled until there is a message in the text box and you have selected at least one UserID to encrypt the message to. Once the button is enabled, all you have to do is click on the button with the mouse.

The Pretty Good Privacy Windows Shell shells out to DOS to encrypt the program and then returns to Windows. Your encrypted message will appear in the Your Encrypted Message text box.

***Just Signing the Message (Clearsig)***

The Clearsig button will be disabled until there is a message in the text box. Once there is a message to sign, click on the Clearsig button.

The PGP Windows Shell will shell out to PGP to sign the message. PGP will prompt you to enter you pass phrase so that it can generate the signature.

***Copying the encrypted message to paste into e-mail***

To use the encrypted message you must copy it to the clipboard and paste it into you e-mail application. To copy the message click on the Copy Message Button. The Pretty Good Privacy Windows Shell will copy the encrypted message to the Windows clipboard. You can then past the encrypted message into the text of an e-mail message and send as you normally would.

***Encryption Options***

### Binary or ASCII Armor

You can choose to use ASCII or Binary armor for your message. Be aware that most e-mail systems cannot handle binary data so PGP Windows Shell defaults to ASCII armor. You can change the armor type for messages by:

- Click on the Options Menu
- Select ASCII Armor (Messages) or Binary Armor (Messages)

The currently selected option is indicated by a check next to in on the menu. The (Messages) options apply only to messages encrypted through the Encrypt A Message tab. The (Files) options apply to files encrypted through the Encrypt & Decrypt Files tab.

### Signatures

You can sign a message so that the receiver can tell if the message has been tampered with.

To enable signatures:

- Click on the Options Menu
- Select Signed Messages & Files

To disable signatures:

- Click on the Options Menu
- Select Unsigned Messages & Files

The currently selected option is indicated by a check next to in on the menu.

***Limitations***

### Binary Armor

The Windows clipboard cannot properly handle the binary armor so you will be asked to specify a file name to save the encrypted message to.

### 32k Text Box Limit

The text box being used is limited to 32k in size. If your encrypted message exceeds this limit, you will be asked to specify a filename to save the message to.

### Command Line Length

If you choose too many UserIDs to encrypt your message too, the PGP Windows Shell will not be able to send all of the UserIDs to PGP through the command line. You will be forced to break up the UserIDs into smaller groups.

**Decrypting a message**

To Decrypt a message you must click on the Decrypt a Message tab.

***Copy the encrypted message to the clipboard***

To decrypt a message you must bring the encrypted message into the Pretty Good Privacy Windows Shell. You must copy the encrypted message from the e-mail message to the clipboard. You must include the "-----BEGIN PGP MESSAGE-----" and the "-----END PGP MESSAGE-----" for the decryption to work.

***Paste the message into the Paste Your Encrypted Message Here text box***

Position the cursor into the Paste You Encrypted Message Here text box and past the encrypted message into it. (You can use Ctrl + v, the Paste command in the Edit menu, or under Windows 95, right click in the text box and select Paste.)

***Decrypt the Message***

The Decrypt Message button will be disabled until a message is in the encrypted message box. When it is enabled, click on it with the mouse. The program will shell out to DOS to decrypt the message. Pretty Good Privacy will ask you to enter you Pass Phase. Type your phrase in and Pretty Good Privacy will decrypt the message. The decrypted message will appear in the decrypted message box.

If the message was not encrypted to you, the decryption will fail and no message will appear in the decrypted message box.

***Options***

### Copy the Decrypted Message to the Clipboard

You may copy the message into the Windows clipboard to paste it into another application.

### Save the Decrypted Message to a file

You may save the decrypted message as a text file. Click on the Save Message button and a standard Save As file dialog box will open. Select a directory and type in the filename. Click OK and the message will be saved.

### Print the Decrypted Message

The Print button will print the message to the default printer.

### Monitor the Clipboard

You can have the PGP Windows Shell monitor the Window's clipboard. If it detects a PGP encrypted message copied to the clipboard, it will automatically decrypt the message and copy the decrypted message back into the clipboard.

This will not happen if you copied the message to the clipboard from within PGP Windows Shell.

This could be annoying so you have the option of turning this feature off.

To enable or disable the clipboard monitor:

- Click on the Options Menu
- Click on the Monitor Clipboard item

If the menu item has a check mark next to it, the feature is turned on, no check mark indicates the feature is off.

### Pause after Decryption and Key Add

You can select this option if you need to see the results of a signature check. If there is a check mark next to this command under the Options menu, this pause is active.

To enable or disable this pause:

- Click on the Options Menu
- Click on the Pause after Decryption and Key Add item

***Limitations***

### 32k Text Box Limit

The text box being used is limited to 32k in size. If your decrypted message exceeds this limit, you will be asked to specify a filename to save the message to.

**Encrypting & Decrypting Files**

You can also encrypt and decrypt files rather than typing messages and using the clipboard. This allows you to avoid the 32k limit of the text box and to encrypt non-text files (documents, pictures, executables, etc.)

You can encrypt and decrypt files from the same tab. You can also use conventional, single key encryption.

***Starting***

To start you must specify the file that you wish to work with. This is the same for encrypting and decrypting files. You can either type the name of the file or click on the Browse button and use a common open file dialog box to find the file.

***Encrypting Files***

### Specify the file

See Starting above.

### Set the options

See Setting Options below

### Selecting UserIDs to encrypt the file to

To encrypt a file you must specify who you are encrypting the file to. Only the people you encrypt the file to can decrypt the file. (Pretty Good Privacy Windows Shell will automatically encrypt any file to your UserID so that you can also decrypt the encrypted file. If this was not done, you would not be able to decrypt the messages you created.)

#### *Selecting a single UserID*

To select a single UserID, find the UserID in the Select Recipients list box and click on it. The UserID that becomes highlighted is the one that the file will be encrypted to.

#### *Selecting multiple, continuous UserIDs*

To select more than one UserID from the list, when all the UserIDs are continuous , you can:

- Click on the first UserID Hold down the Shift key and click on the last UserID

Or you can:

- Click on the first UserID and hold the mouse button down
- Drag the cursor over all the UserIDs you want
- Release the mouse button on the last UserID

#### *Selecting multiple, non-continuous UserIDs*

To select more than one UserID from the list, when all the UserIDs are not continuous , you can:

- Hold the Ctrl key and click on each UserID you want

Or you can:

- Select a continous list of UserIDs as described about and
- Hold the Ctrl key and click on each UserID in the select list that you do not want

### Encrypting the File

To encrypt the file:

- Click on the Encrypt button

The shell will start PGP and the file will be encrypted.

Files encrypted using binary armor will have the original file name except with the .pgp extension.

ASCII armor files will have the .asc extension. If the resulting .asc file is over a certain size limit, the file will be broken up into multiple files with .as1 - .asn extensions. This is a function of PGP and is designed to allow the ASCII armor files to be sent by e-mail which often has a 64k limit. This behavior can be changed by setting PGP configuration options. (See the PGP documentation for more information.)

#### *Decrypting Files*

### Specify the file

See Starting above.

### Set the options

See Setting Options below

### Decrypting the File

To decrypt the file:

- Click on Decrypt button

If you specified renaming the decrypted file or leaving the signature attached, the PGP Windows Shell will ask you for a file name to save to using a common save file as dialog box. Once you specify the name, PGP will attempt to decrypt the file.

If you chose to keep the original file name, PGP will attempt to decrypt the file.

PGP will ask you for your pass phase or the pass phase of the file for conventional encryption, before it can decrypt the file.

*Conventional Encryption*

### Specify the file

See Starting above.

### Set the options

See Setting Options below

### Encrypting the File

Since convention encryption is single key, you do not have to specify any recipients. To encrypt the file:

- Click on the Conventional Encryption button

PGP will ask you for a pass phase and ask you to verify it. Once the pass phrase is entered, the file will be encrypted.

Files encrypted using binary armor will have the original file name except with the .pgp extension.

ASCII armor files will have the .asc extension. If the resulting .asc file is over a certain size limit, the file will be broken up into multiple files with .a01 - .an extensions. This is a function of PGP and is designed to allow the ASCII armor files to be sent by e-mail which often has a 64k limit. This behavior can be changed by setting PGP configuration options. (See the PGP documentation for more information.)

*Setting Options*

### Armor Options

You can specify either ASCII Armor or Binary Armor by clicking on the appropriate button.

ASCII armor files are saved with the .asc extension unless the resulting file is too long. (See the PGP documentation for the discussion of this feature.)

Binary armor files are saved with the .pgp extension.

### Signatures Options

You can specify Signed or Unsigned encryption by selecting the appropriate button or menu item under Options. This affects file and message encryption.

### Source File (Encryption) Options

You can tell PGP to either wipe and delete the source file or to keep the source file intact.

### Destination Files (Decryption) Options

You can either rename the decrypted file or let PGP try to keep the original file name

### Other Options

All of these options are one shot only. They are reset to off after they are used.

#### *Separate Signature File*

If you want to create a separate signature file, click on this button. Select the file you want to sign and click on Encrypt. The signature file will be the original file name but with a .sig extension.

#### *Leave Signature Attached to Decrypted File*

If you want to leave a signature attached to a file but read the decrypted text, click on this button and decrypt like normal.

### *ASCII Wrapper Only*

Sometimes you might want to create just an ASCII wrapper for a binary file so you can send it through e-mail. This option is a similar function to UUENCODE. The file is not encrypted but it is compressed and becomes ASCII text rather than binary data. Files that are too large are broken into smaller chunks.

To create just an ASCII wrapper, click on this button and then encrypt (no recipients are needed).

### *Clearsig (-sta)*

Sometimes you want to send a plain-text but signed message. Use this option. The resulting text is human readable but has a signature attached at the end.

#### *Limitations*

The only limitation I know of for file encryption and decryption is the length of the command line. If the command line gets too long, you will be asked to select a smaller group to encrypt the file to.

PGP Windows Shell will pause to let you see the output file name. Please check this before you press a key to close the DOS window. PGP Windows Shell cannot intercept this output file name. If you specified an output file name prior to decrypting a file, there will not be a pause.

#### Pausing The PGP Shell Process

If you have your PIF or Properties settings close the DOS windows when the shell process is over, there is little time to see any messages PGP may be giving you. To get around that there are a few pausing mechanisms built in.

#### *Normal Pausing*

The PGP Windows Shell will automatically pause before returning to Windows when you Encrypt or Decrypt a file. This lets you see the output file name. The only time this does not happen is when you have specified an output file name for a decrypted file.

#### *Message Decryption and Key Adding Pauses*

If you receive a signed message, you need to be able to see PGP signature check message. When you add a new key, you may want to see the message about the keys that may or may not have been added. To let you do this, you may set the Pause after Decryption and Key Add under the Options menu.

#### *Error Check Pauses*

PGP is a complicated program and it is picky about what it receives from the user. If you are having problems getting things to work, you may set the Error Checking pause under the Options menu. This inserts a pause into every shell to PGP process and lets you see what PGP is telling you.

#### Save Options

When you choose the Save Options menu item or click on the Save File Options button on the Encrypt & Decrypt Files tab the following options are saved to the PGPSHELL.INI file under the User section. The Save Options menu item saves the options located under the Options menu, the Save Files Option button saves the options located on the Encrypt & Decrypt Files tab.

- Message Armor (as ASCII_ARMOR_Messages key)
- File Armor (as ASCII_ARMOR_Files key)
- Monitor Clipboard (as Monitor_Clipboard key)
- Signed or Unsigned (as Signed key)
- Rename output file (as Rename key)
- Wipe source file (as Wipe key)

- Decrypt and Add Key Pause (as Pause key)
- Error Check Pause (as ErrorCheck key)

A value of -1 indicates the option is on, a 0 indicates it is off.

Your display font is automatically saved when you set it. The font is saved as Display Font and Font Size under User. If it is not there, your display font is still set to the default.

Your UserID is saved as the UserID key under User and is saved based on its KeyID.

**Admin**

Key administration is an important part of Pretty Good Privacy. It is a detailed subject covered by Pretty Good Privacy's documentation (PGPDOC1.TXT and PGPDOC2.TXT) so I won't go into much detail here.

Key administration functions are located on the Admin Tab. Most of the important Key admin functions are included. Functions not found here have to executed manually from the DOS prompt.

***Generate Your Key***

To generate you public and private keys, click on the Generate Your Key button. The shell will shell out to DOS and have PGP.EXE walk you through the key generation process. Follow the on screen prompts. The shell now uses the +nomanual option so you can still generate a key without having the PGP users manuals on your system.

***Set Your UserID***

Set Your UserID is strictly a PGPShell function. By setting you UserID, the Pretty Good Privacy Windows Shell will automatically encrypt all of your messages so that you decrypt them. If your UserID is not set, you will not be able to decrypt messages you send.

This UserID is also the one that PGP Windows Shell will use to sign your messages and files with.

### *Any button working with your UserID/Key will work on this UserID*

Setting your UserID is usually done when you first configure PGPShell. This button is included in case you change your key and UserID or use multiple UserIDs.

### *To set your UserID*

- Click on the Set Your UserID button

A list box and two buttons will appear.

- Select your UserID from the UserIDs displayed
- Click on the Set UserID button (it is disabled until you select a UserID in the list box)

***Edit Your UserID/Pass Phrase***

Sometimes you may need to edit or change your UserID or pass phase. To do this:

- Click on the Edit Your UserID/Pass Phrase button

The program will shell to PGP which will guide you through the editing process. This command will function with the UserID you set with the Set/Change Your UserID button.

***Revoke Your Key***

If you believe that your UserID has been compromised you can revoke your UserID. To do this:

- Click on the Revoke Your Key button

PGP will revoke your UserID. You should immediately generate a new key and send the revoked key to all the people you have exchanged keys with. (Your revoked key will still show up (as revoked) on the list to send it using Send a Public Key below.)

***Add a Public Key***

You can't encrypt a message to someone unless you have their public key. This button lets you add public keys to your keyring.

You will usually receive somebody's public key as an e-mail message or as a file. The add a public key function allows you to use either one.

To add a public key:

- Click on the Add a Public Key button (the PGPShell will display the Add A Public Key dialog box.)

If the key came as an e-mail message:

- Copy the key signature to the clipboard, include the "-----BEGIN PGP PUBLIC KEY BLOCK-----" and "-----END PGP PUBLIC KEY BLOCK-----"

Paste the key signature into the Option 1 text box Or if the key came a file on a disk:

- Type the name and path to the file in the Option 2 text box

Or, click on Browse and locate the file in the Open File dialog box Next:

- Click on the Add Key button

If you have additional keys to add continue to paste the keys into the Option 1 box or type the filename into the Option 2 box and clicking the Add Key button.

When you are done:

- Click on Close

When the system asks you to standby while it rebuilds your UserID list:

- Click OK
- Or Click Cancel (if you do this you save a few seconds but you will not be able to encrypt messages to those new public keys until you rebuild the UserID list)

***Send a Public Key***

For people to encrypt messages to you, you must send them your public key. You can also share other people's public keys.

To send a public key:

- Click on the Send a Public Key button
- Select the UserID for the key you wish to send
- Click on the Extract Key button (if will be disabled until you select a UserID)

The Windows Shell will shell out to DOS and extract the key from your keyring. When it is done, the key is displayed in a text box.

- Click on the Copy Key button to copy the key to the clipboard
- Paste the key into you e-mail messages

***Sign a Key (-ks)***

Sometimes you want to sign a key. To do this:

Click on the Sign A Key button

A list box and two buttons will appear.

- Select the UserID to sign
- Click on the Sign It button (it is disabled until you select a UserID in the list box)

The system will use PGP to sign the key on your keyring. You will have to type your pass phrase when PGP asks for it.

***Remove a Key/UserID***

Sometimes you will have to remove keys and UserIDs from your keyrings and UserID list.

To remove keys and UserIDs:

- Click on the Remove a Key/UserID button

A list box and two buttons will appear.

- Select the UserID to remove
- Click on the Delete button (it is disabled until you select a UserID in the list box)

The system will then rebuild your UserID list.

***Disable/Enable a Key***

If you need to disable a key instead of remove it:

To remove keys and UserIDs:

- Click on the Disable/Enable a Key button

A list box and two buttons will appear.

- Select the UserID to disable/enable
- Click on the Disable/Enable button (it is disabled until you select a UserID in the list box)

The system will then rebuild your UserID list.

To enable a UserID, follow the same procedure.

Disabled UserIDs will appear in the list boxes with "***DISABLED***" after them.

***View a Key's Fingerprint***

If you wish to view a key's fingerprint:

- Click on the View a Key's Fingerprint button

A list box and two buttons will appear.

- Select the UserID to view
- Click on the View Fingerprint button (it is disabled until you select a UserID in the list box)

The shell will have PGP generate the fingerprint and PGP Windows Shell will display it.

***Edit a Key's Trust Factor***

To edit the trust parameters for a key:

- Click on the Edit a Key's Trust Factor button

A list box and two buttons will appear.

- Select the UserID to edit
- Click on the Edit Trust button (it is disabled until you select a UserID in the list box)

PGP will walk you through the editing process.

***Rebuild UserID List***

Pretty Good Privacy automatically maintains its UserID lists in your keyrings. Pretty Good Privacy Windows Shell can not directly access that list and must ask Pretty Good Privacy for the list so that the Windows Shell can maintain its list properly.

Every time you make a change to the keys and UserIDs from the Windows Shell, it will automatically rebuild the UserID list (unless you click on the Cancel button).

If you make changes to the keys and UserIDs using Pretty Good Privacy directly from the DOS prompt, you will have to rebuild the Windows Shell's UserID list.

To do this:

- Click on the Rebuild UserID List button
- Click on the OK button on the dialog box that pops up

# A Note from the Programmer

## *It's Freeware!*

This seemed like a utility that many people might like to have and since the STIGNER Project Office is a DOD office and the American taxpayer paid for this program, we have released this as a freeware utility.

## *Why did I write this?*

Dave Kennedy at the STINGER Project Office at Redstone Arsenal needed a way to e-mail information that needs a little more security than unencrypted e-mail could provide so he had us get Pretty Good Privacy for him and make it easy to use. At the time we didn't know how many PGP shells there were out there and did not realize that we could have saved time by downloading one of those shells. Several users have said that this shell is the easiest to use of the ones they have tried so we have continued to develop it.

## *A living, breathing program*

This is a living, breathing program. Bugs (what bugs??, not in my software) will be fixed and the product will be improved.

This second version includes most of PGP's command line switches but still leaves out some of the more esoteric commands.

I still hope to have a 32 bit version done but Visual Basic 4 has been out for awhile now and I don't see it happening too soon. Since it uses some VBXes which are 16 bit controls, it will take a little work replacing them to make it a pure 32 bit shell.

I plan to add better file handling features in the next version. I want add multiple file encryption at a time. One beta user requested automatic reassembling of multiple .as1, .as2 etc. files. I hope to quickly wrap this into Windows 95's MS Exchange for folks who use the WordMail option after they install Word 7.

The long-term goal is for this program to be able to send and receive encrypted e-mail directly with most major e-mail systems.

## Thanks to My Beta Testers

Version 1 of the program was written in less than a week and released. It did not do much and did not require much testing.

Version 2 does more and polished what version 1 did. It required more stress testing and I had a group of about 10 folks testing and evaluating Version 2 for a couple of months. Without their input, this would not be the shell that it is. Thank you.

## Where is the Latest Version?

I will always try to have the latest version posted on my company's Web pages. It will be located off of the main page. I will keep it there until we start to run into storage space problems or I have not made any changes in a long time. Our URL is:

- http://iquest.com/~aegisrc

## Let me hear from you

If you give me an e-mail address, I will try to let you know when and where new versions are available.

If you have problems or suggestions, let me know. I may not get back to you directly but I will try to fix the problems and incorporate suggestions in the next version.

You can reach me at: Mike Lyman
AEgis Research Corporation
6703 Odyssey Dr., Suite 200
Huntsville, AL 35806
Phone: (205) 922-0802
e-mail: mlyman@aegisrc.com
MSN: Mike_Lyman
CompuServe: 71563,526