

Deploying PGP Encryption and Compression for z/OS Batch Data Protection to (FIPS-140) Compliance

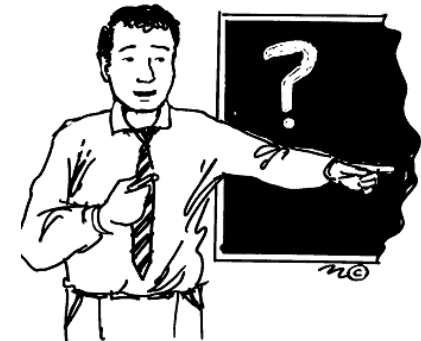
Patrick Townsend
Software Diversified Services/Townsend Security

August 9, 2011
Session Number 9347

PGP Command Line 9 for z/OS Topics

PGP

- History
- Business Motivators
 - Compliance Drivers
- What Is PGP?
- How Does it Work?



PGP for z/Series Mainframe

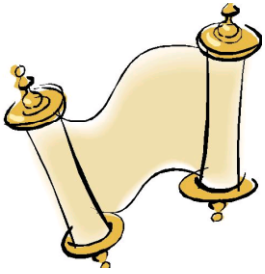
- Installation
- Batch Samples
- Additional Technical Slides
- Key Management & Encryption

Technical Support

Introductions

Patrick Townsend

CTO - Townsend Security



History of PGP

- 1998 – Townsend and Phil Zimmerman agree on port to IBM Enterprise servers
- 1999 – First implementation of PGP on IBM AS/400 servers with NAI
- 2001 – First integrated FTP and PGP application on IBM AS/400
- 2004 – McAfee takes over name and control of PGP
- 2005 – PGP Corporation takes control of PGP Command Line 9
- 2005 – Townsend releases PGP Command Line 9 for zLinux and Linux on IBM i
- 2008 – Townsend releases PGP Command Line 9 on System z USS
- 2009 – Townsend releases PGP Command Line 9 on IBM i
- 2010 – Townsend releases PGP Command Line 9 for System z z/OS
- 2010 – Townsend supports VSAM file encryption for System z z/OS

What are the business motivators ?

- Compliance regulations require data encryption: PCI, HIPAA /HITECH, FFIEC, etc.
- Trading partners require encryption (banks, insurance companies, etc.)
- Corporate security policies require encryption of valuable assets
- Corporate risk management requires brand protection

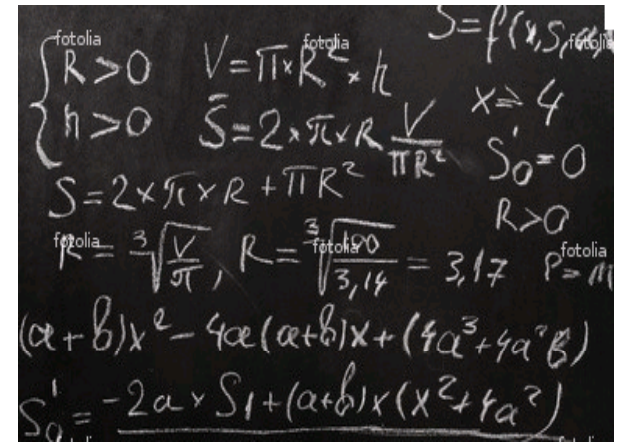
Compliance regulations that require encryption

- PCI Data Security Standards (PCI DSS)
- HIPAA / HITECH Act for medical industry
- State Privacy Laws (45 states)
- GLBA / FFIEC for banking industry
- Proposed Federal Privacy Law
(passed House, in the Senate)
- FERPA for educational institutions



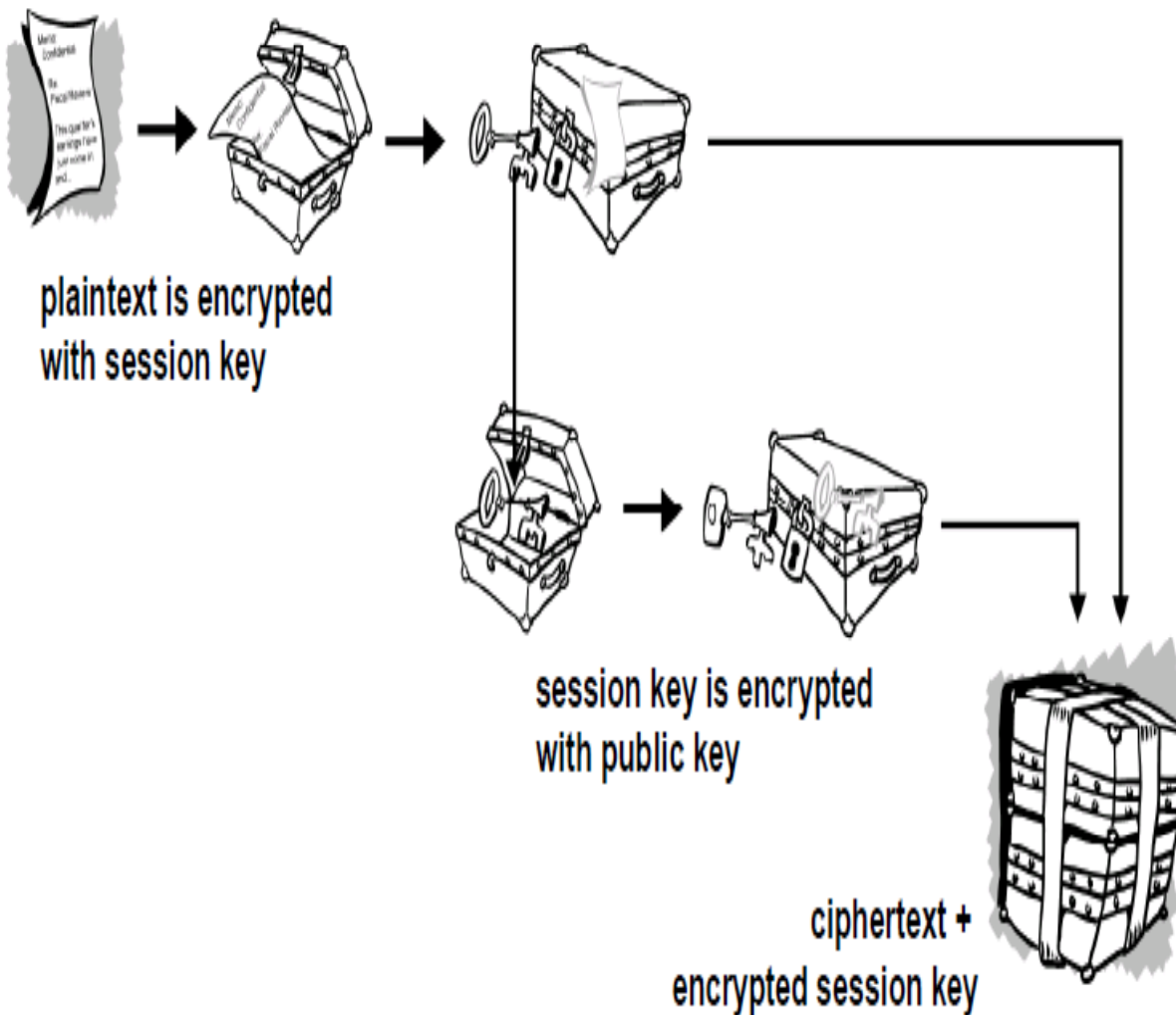
What is PGP Command Line 9 ?

- Whole file encryption
- Based on open standards (RFC 2440)
- Public / Private key infrastructure
- Enforces strong encryption (AES)
- Insures file integrity
- Non-repudiation of sender
- Cross platform implementation (Windows, Linux, Unix, IBM z, IBM i)
- Widely used in banking, finance, insurance, medical, and other industries
- Compression



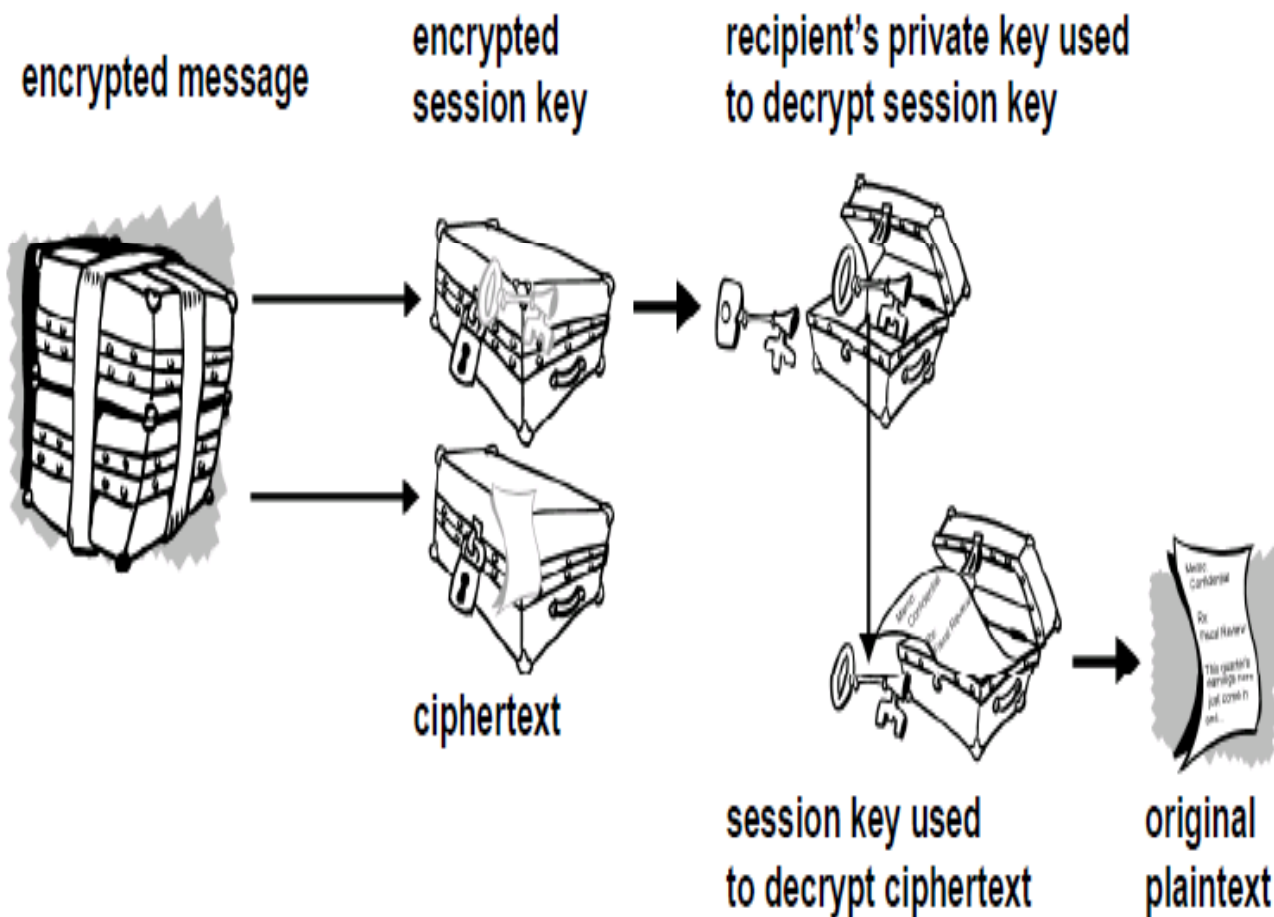
$$\begin{aligned} & \begin{cases} R > 0 \\ h > 0 \end{cases} \quad V = \pi R^2 \times h \quad S = 2\pi R^2 + \pi R^2 \frac{V}{\pi R^2} \\ & S = 2\pi R^2 + \pi R^2 \frac{V}{\pi R^2} \quad x \geq 4 \quad S_0 = 0 \\ & R = \frac{3}{\sqrt[3]{\frac{V}{\pi}}} \quad R = \frac{3}{\sqrt[3]{\frac{100}{3.14}}} = 3.17 \quad R > 0 \quad P = 11 \\ & (a+b)x^2 - 4a(a+b)x + (4a^3 + 4a^2b) \\ & S_0' = -2a \times S_1 + (a+b)x(x^2 + 4a^2) \end{aligned}$$

How does PGP work - Encryption?



- Compresses files first – reduces patterns
- Session Key – one Time Secret Key
- Session Key encrypted to recipients public key

How does PGP work -Decryption?



➤ Decryption works in reverse

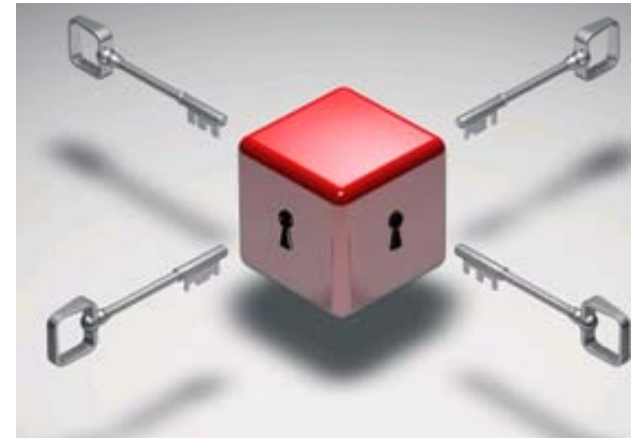
➤ Private Key used to recover temporary session key

PGP Key Servers

PGP Universal Server

PGP Key Management Server

- PGP Encryption Key Management
- Email and application integration
- Automated administration
- Software appliance



Native z/OS and USS Implementations

Two implementations for IBM System z users:

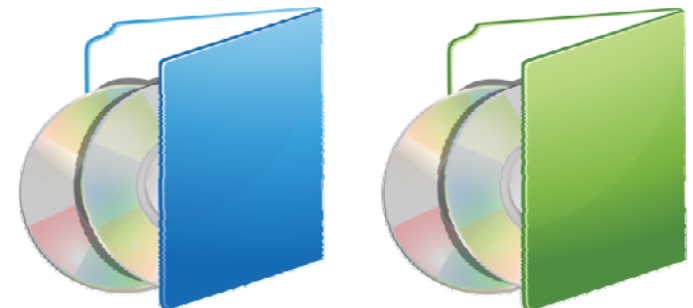
- ☐ Native z/OS implementation
- ☐ Unix System Services (USS) full portation
- ☐ USS supports OMVS command line and JCL
- ☐ Native z/OS works better in batch JCL
and with RACF / ACF2 security
- ☐ Additional Decryption Keys (ADK)
- ☐ Self Decrypting Archives (SDA's)
(Windows, Linux, Mac, OSX, UNIX)



z/OS and USS

The implementation of PGP Command Line 9 supports any version of z/OS from 1.7 forward. The product is delivered with two separate operating environments.

- z/OS Batch and JCL native executable
- USS executable through JCL and OMVS command line



Complete PGP Implementation

PGP Command Line 9 for the IBM z/OS platform is a complete implementation with support for:

- Multiple PGP key files
- PGP Universal Key Server
- Additional Decryption Keys (ADK)
- Self-decrypting archives
(Windows, Linux, Mac OSX, UNIX)

This product was not hobbled to accommodate the different file structures or operating systems of z/OS.



Cross-Platform Support

PGP Command Line 9 for the IBM z/OS can encrypt files for a variety of target platforms including:

- Windows
- Linux (Red Hat, SuSE, etc.)
- UNIX (AIX, Solaris, HP-UX, etc.)
- IBM i
- IBM z

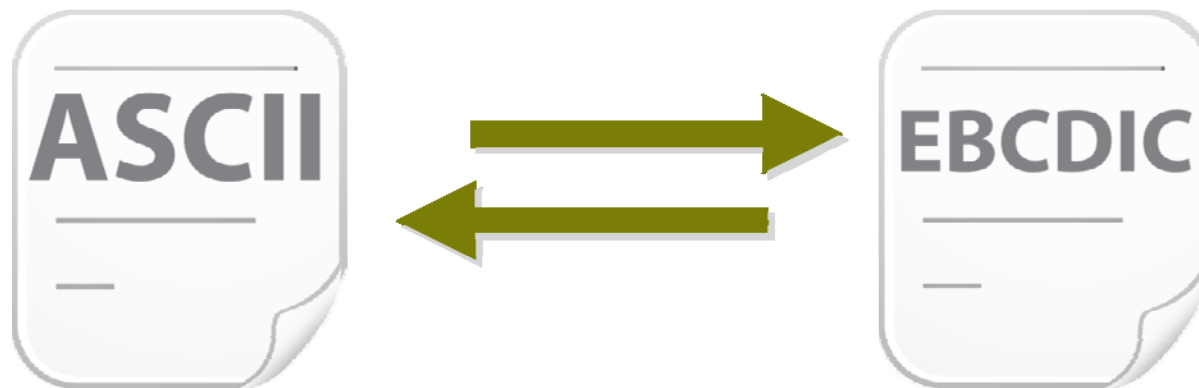
PGP Command Line 9 can decrypt files from all of these environments, and is compatible with any OpenPGP implementation.



ASCII and EBCDIC Character Conversion

PGP Command Line 9 uses an internal ASCII character set format for meta information. The payload can be either EBCDIC or ASCII data.

PGP Command Line 9 includes utilities to convert files in EBCDIC to the ASCII character set, or decrypted ASCII files to EBCDIC.



Supported File Systems

- PDS
- PDSE
- VSAM
- Sequential Files



PGP File Format

- File Format is ASCII
- Regardless of the format of encrypted data, a PGP encrypted dataset can be copied in Binary Mode to another platform and decrypted there
- Files encrypted elsewhere can be copied in Binary mode to a dataset and decrypted
- ASCII basis of the PGP file format refers to it's Internal Structure and not the Payload
- PGP does not translate or interpret the contents of the encrypted file
- The original input to encrypt can be EBCDIC and when decrypted will be EBCDIC even if decrypted on a ASCII platform (Windows)

```
100110001110110000111101110011110011
00011101100001111101110011110011000111
0110000111101110011111001100011101100
0011110111001111001100011101100001111
101100011100011110111001111001100011
10110001111011001111001100011101100011
000111101110011110011000110110000111
11011001111001100011011000011110111
00111100110001101100011110110011110111
00111100110001101100011110110011110111
0001111011001111001100110110000111
110110011110011000111011000011110111
001111001100011101100001111011001111
100110001110110000111101100111100110
00111011000011110110011100011110111
001111001100011101100001111011001111
100110001110110000111101100111100110
```

PGP in Batch with JCL

Both native z/OS and USS versions of PGP Command Line 9 can run in JCL batch environments. An example of JCL code -
Encrypting a Text File:

```
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT      BCVR1.014.PGPCL9.JCL (PGPENCRF) - 01.22      Columns 00001 00072
Command ==> Scroll ==> CSR
000019 //PGPPRVKY DD DSN=BCVR1.014.PGPCL9.DATU (PRIVKEY1), DISP=OLD
000020 /*
000021 //ENCIN DD DSN=BCVR1.014.PGPCL9.INFB80 (TSTF1), DISP=SHR
000022 //ENCOUT DD DSN=BCVR1.014.PGPCL9.TSTDATU (TSTF1ENC), DISP=SHR
000023 /*
000024 //SYSIN DD *
000025 --encrypt
000026 DD:ENCIN
000027 -r
000028 Alice
000029 -o
000030 DD:ENCOUT
000031 /*
000032 //COPYSTEP EXEC PGM=CONVERT
000033 //STEPLIB DD DSN=BCVR1.014.PGPCL9.LOADLIB, DISP=SHR
000034 // DD DSN=CEE.SCEERUN, DISP=SHR
000035 // DD DSN=CEE.SCEERUN2, DISP=SHR
000036 //SYSIN DD DSN=##STDOUT, DISP= (OLD, DELETE)
000037 //SYSPRINT DD SYSOUT=*
000038 /*
```

PGP in Batch with JCL

JCL Example on how to Decrypt a Text File

```
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT      BCVR1.014.PGPCL9.JCL (PGPDECRF) - 01.13      Columns 00001 00072
Command ==> _____ Scroll ==> CSR
000024 //DECIN      DD  DSN=BCVR1.014.PGPCL9.TSTDATU(TSTF1ENC),DISP=SHR
000025 //DECOUT     DD  DSN=BCVR1.014.PGPCL9.OUTFB80(TSTF1DEC),DISP=SHR
000026 /*
000027 //SYSIN      DD *
000028 --decrypt
000029 DD:DECIN
000030 --passphrase
000031 alices passphrase
000032 -o
000033 DD:DECOUT
000034 /*
000035 //COPYSTEP EXEC PGM=CONVERT
000036 //STEPLIB DD  DSN=BCVR1.014.PGPCL9.LOADLIB,DISP=SHR
000037 //          DD DSN=CEE.SCEERUN,DISP=SHR
000038 //          DD DSN=CEE.SCEERUN2,DISP=SHR
000039 //SYSIN      DD DSN=&&STDOUT,DISP=(OLD,DELETE)
000040 //SYSPRINT    DD SYSOUT=*
000041 /*
***** ***** Bottom of Data *****
```

PGP in Batch with JCL

JCL Example on how to Sign a File

```
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT      BCVR1.014.PGPCL9.JCL (PGPSIGN) - 01.14      Columns 00001 00072
Command ==> Scroll ==> CSR
000019 //ENCIN      DD  DSN=PGP.CL9.INFB80(TSTF1),DISP=SHR
000020 //ENCOUT     DD  DSN=PGP.CL9.TSTDATU(TSTSIGN1),DISP=SHR
000021 /*
000022 //SYSIN      DD *
000023 --sign
000024 DD:ENCIN
000025 --signer
000026 Alice
000027 --passphrase
000028 alices passphrase
000029 -o
000030 DD:ENCOUT
000031 /*
000032 //COPYSTEP EXEC PGM=CONVERT
000033 //STEPLIB      DD  DSN=PGP.CL9.LOADLIB,DISP=SHR
000034 //             DD  DSN=CEE.SCEERUN,DISP=SHR
000035 //             DD  DSN=CEE.SCEERUN2,DISP=SHR
000036 //SYSIN      DD  DSN=&&STDOUT,DISP=(OLD,DELETE)
000037 //SYSPRINT     DD  SYSOUT=*
000038 /*
```

PGP in Batch with JCL

JCL Example on how to Export a Key

```
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT      BCVR1.014.PGPCL9.JCL (PGPEXPXY) - 01.11      Columns 00001 00072
Command ==> _____ Scroll ==> CSR
000014 //PGPPREFS DD DSN=PGP.CL9.DATU (PGPPREFS), DISP=SHR
000015 //PGPRANDS DD DSN=PGP.CL9.DATU (RANDSEED), DISP=SHR
000016 //PGPPUBKY DD DSN=PGP.CL9.DATU (PUBKEY1), DISP=OLD
000017 //PGPPRVKY DD DSN=PGP.CL9.DATU (PRIVKEY1), DISP=OLD
000018 //*
000019 //AKEYOUT DD DSN=PGP.CL9.TSTDATU (AKEYOUTX), DISP=SHR
000020 //*
000021 //SYSIN DD *
000022 --export
000023 Alice
000024 -o
000025 DD:AKEYOUT
000026 /*
000027 /*
000028 //COPYSTEP EXEC PGM=CONVERT
000029 /* convert the ascii STDOUT output of pgp to ebcdic and
000030 /* print it. STDOUT is in the dataset specified by SYSIN DD
000031 //STEPLIB DD DSN=PGP.CL9.LOADLIB, DISP=SHR
000032 // DD DSN=CEE.SCEERUN, DISP=SHR
000033 // DD DSN=CEE.SCEERUN2, DISP=SHR
```

PGP z/OS Documentation

The primary user reference is the standard PGP command Line 9 user guide. This documentation provides guidance on all PGP functions such as adding and exporting keys, encrypting and decrypting files, signing files, and so forth.

The PGP Command Line 9 z/OS User Guide provides specific information about installing and configuring PGP on your IBM z server. You will use this document to get started with PGP.



PGP Installation on IBM z

PGP Command Line 9 for z/OS is provided as a ZIP compressed file. You will unzip this file and review the Readme file for information about how to transfer the enclosed binary install file to the z/OS platform. You will also find the PGP Command Line z/OS User Guide in this archive in PDF format.

The license file will be included with the download. Please be sure to read the instructions on how to apply a temporary or permanent license after you install the product.



PGP and RACF / ACF2 / TOP SECRET

The z/OS PGP application is a native z/OS executable and can be put under RACF (ACF2, Top Secret) control. This gives you the ability to control access to the PGP application and monitor its use.

The USS implementation of PGP Command Line 9 can also be put under RACF control as a USS application.

RACF

Technology Roadmap – z/OS

- PGP Command Line 10
 - Stand alone Key Retrieval
 - Symmetrical Key Retrieval Support
 - Private Key Storage on the Key Server
- Hardware Acceleration



Commercial PGP Encryption Facility vs. PGP CL9

- PGP CL9(supports Additional Decryption Keys (ADK). OpenPGP does not. Companies that need to encrypt files to their own keys to meet eDiscovery requirements should use PGP
- PGP CL9 supports Self Decrypting Archives, OpenPGP does not
- PGP CL9 is a native z/OS batch application. IBM EF OpenPGP requires Java
- PGP CL9 integrates with PGP Key Servers and Symantec's Universal Key Server. OpenPGP does not integrate with Key Servers



Any Questions?

Contact Us

Software Diversified Services

sales@sdsusa.com

www.sdsusa.com

763-571-9000