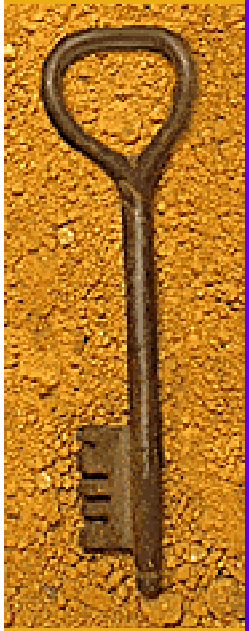
Using Pretty Good Privacy

One User's Experience

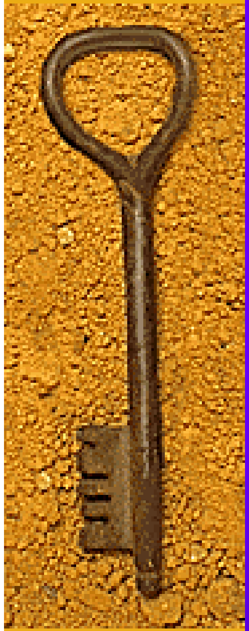
SHARE Nashville

Session 5516



Abstract

- PGP or "Pretty Good Privacy" has been described as "one of the most revolutionary pieces of software ever created." Join us as we take a look at both the history and usage of this exceptional program. We will look at both the Windows and UNIX versions of PGP, and see how PGP allows us to both digitally sign and encrypt our correspondence. We'll also take a look of the status of PGP on MVS, and what ports are available for use on the S/390.



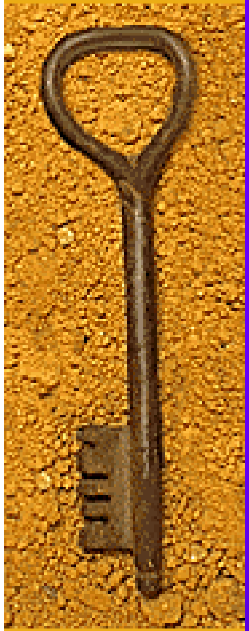
The Speaker

Harold Pritchett

The University of Georgia

1-706-542-0190

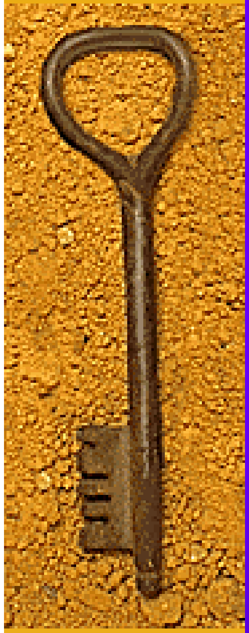
harold@uga.edu



Disclaimer

Everybody has lawyers:

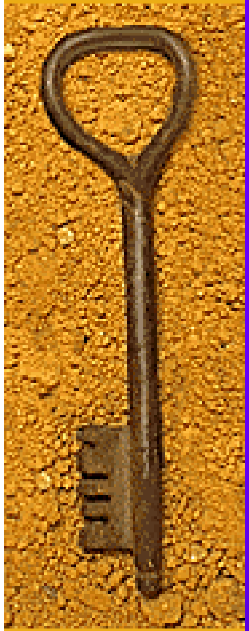
The ideas and concepts set forth in this presentation are solely those of the author, and not of the companies and or vendors referenced within and these organizations do not endorse, guarantee, or otherwise certify any such ideas or concepts in application or usage. This material should be verified for applicability and correctness in each user environment. No warranty of any kind available.



Privacy, Why do we need it?

“Show me a human being who has no secrets from her family, her neighbors, or her colleagues, and I'll show you someone who is either an extraordinary exhibitionist or an incredible dullard. Show me a business that has no trade secrets or confidential records, and I'll show you a business that is not very successful..

André Bacard, 1995



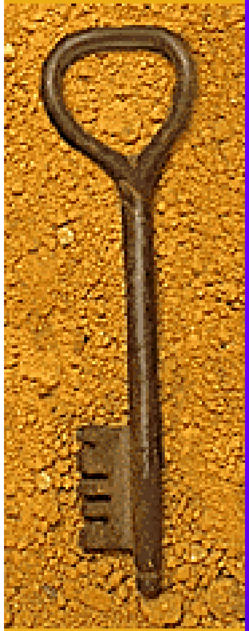
Privacy, Why do we need it?

“When privacy is outlawed, only outlaws will have privacy”

Phillip Zimmerman

“Privacy, it’s not just for criminals and terrorists”

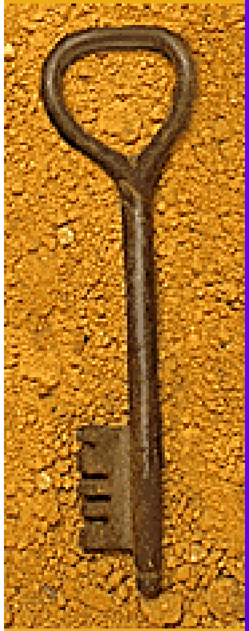
Harold Pritchett, 2001



Privacy, Why do we need it?

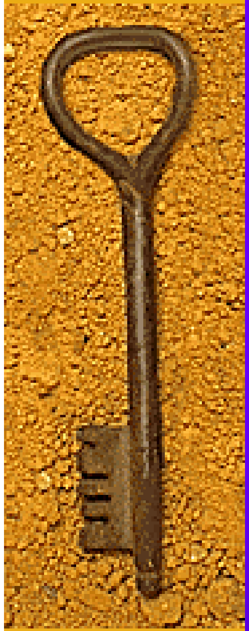
"They that give up essential liberty to obtain a little temporary safety deserve neither liberty nor safety.“

Benjamin Franklin, 1759



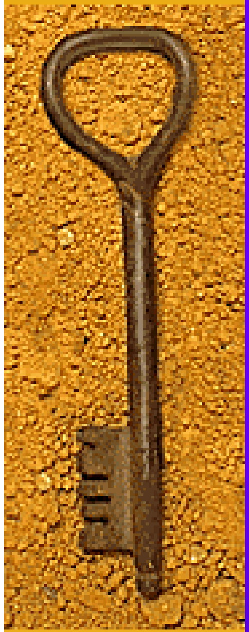
Introduction

- Who am I?
- What makes me qualified to talk about PGP?
  - 25 Years working with computers
  - Almost 10 Years experience with PGP
    - My first PGP key is dated December 1992
  - UNIX Security Administrator
  - Computer Incident Response Team for UGA



Problems needing a solution

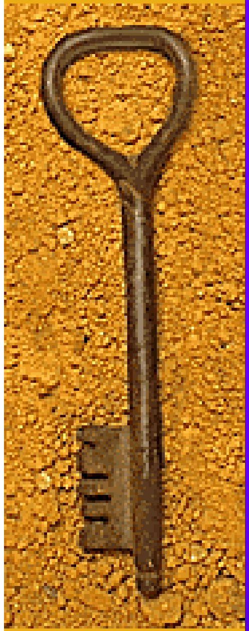
- Encryption
- Authentication
- Key distribution



Cryptographic Basics

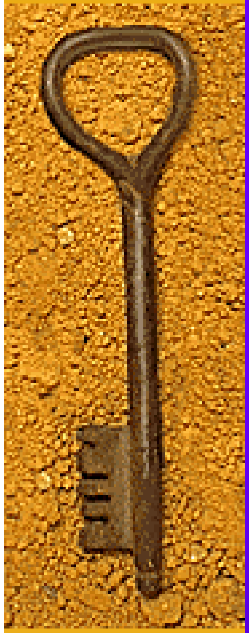
- It’s Mathematics!
- All modern cryptography is based upon very high level mathematics. I am not a mathematician, nor do I play one on TV. There are many excellent books and locations on the World Wide Web which explains this stuff for those with an interest. Parts of this section of this talk are based upon one of these sites:

  www.scramdisk.clara.net/pgpfaq.html



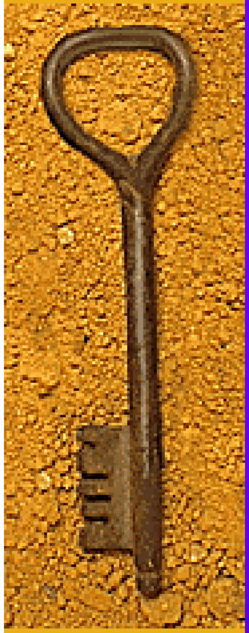
A few definitions

- The basic building blocks of cryptography
  - Checksums
  - Hash Functions
  - Symmetric Key Cryptography
  - Asymmetric Key Cryptography



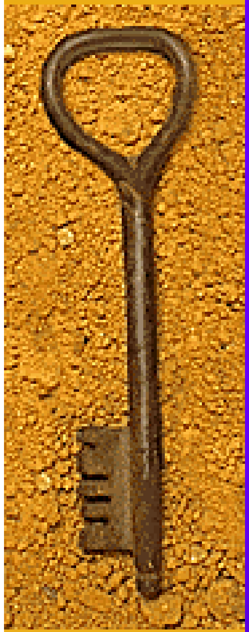
Checksums & Hash Functions

- A checksum or CRC function is a simple, non-cryptographic mechanism for detecting transmission errors
- A hash function is a number derived from a string of text, in such a way that it is extremely unlikely that some other text will generate the same hash value



Checksums & Hash Functions

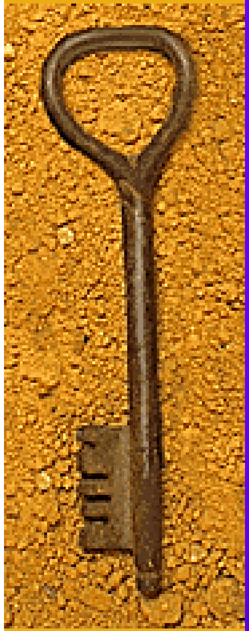
- Hash functions are really just more complex checksums
- Hash functions are responsible for two primary tasks in PGP
  - Creation of digital signatures
  - Conversion of the pass phrase into a cryptographic key



Examples

Hash Functions

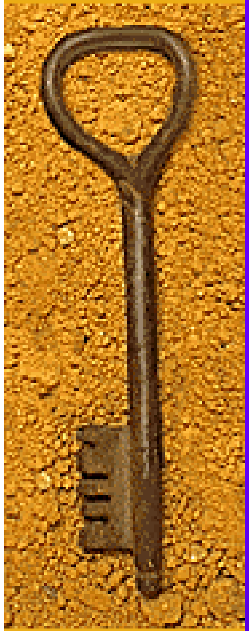
- MD5
  - Message digest 5 – Ron Rivest – 1991 – rfc 1321
  - 128 bit
- SHA1 (Secure Hash Algorithm)
  - Developed by NIST - 1995 – FIPS Pub 180-1
  - 160 bit – developed in secret
- RIPEMD160
  - Dobbertin, Bosselaers, and Preneel – 1991
  - 160 bit – published algorithm



Cryptographic Systems

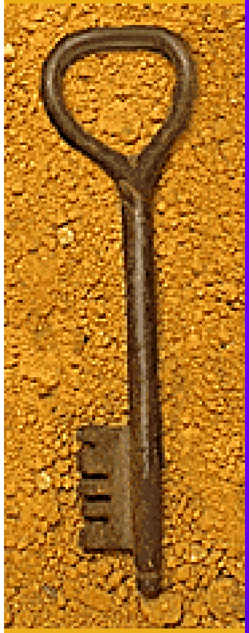
- Symmetric
  - Same key used to encrypt and decrypt
  - Fast
  - More secure if right key length and algorithm
  - Big key distribution problem

- Asymmetric
  - Different keys used to encrypt and decrypt
  - Slow
  - Allows for digital signatures
  - No key distribution problem



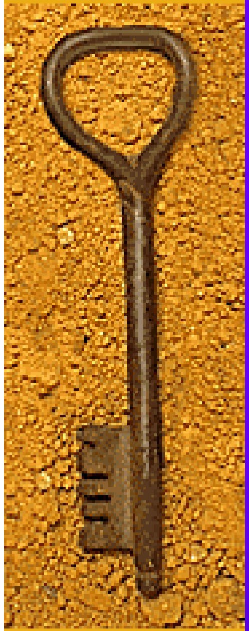
Some examples of Symmetric key Algorithms

- DES (Data Encryption Standard – 56 Bits)
- Triple-DES
- RC2, RC4
- IDEA (International Data Encryption Algorithm – 128 Bits)
- CAST
- Skipjack
- Twofish
- Blowfish



Some examples of Symmetric key Algorithms

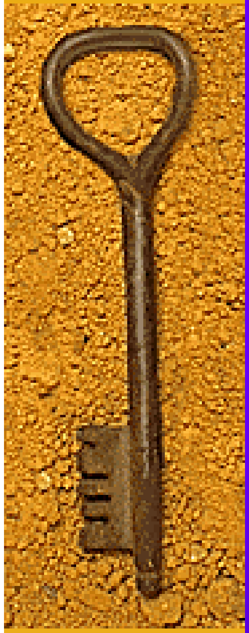
- ♦ AES – Advanced Encryption Standard
  - – New replacement for DES
  - – Announced in 2001 as the new standard and formally incorporated into FIPS 197
  - – Uses the Rijndael encryption formula
  - – Developed by Belgian cryptographers Joan Daemen and Vincent Rijmen
  - – Supports keys of 128, 192, and 256 bits



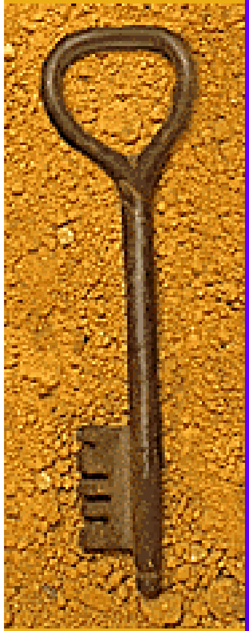
Some examples of Asymmetric key Algorithms

- RSA (Rivest, Shamir, and Adleman)
- Diffie-Hellman (DH)
  - El Gamal
- Merkle-Hellman (Knapsack)
  - Depreciated

PGP – what is it?

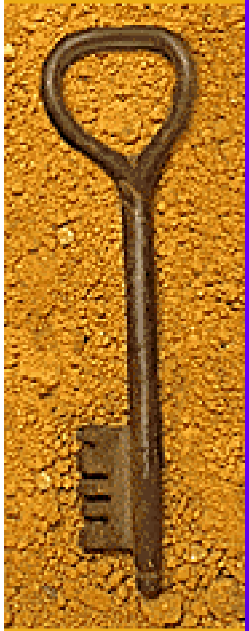


PGP is a program that uses encryption to protect the privacy of your electronic mail and the files you store on your computer. It can also be used to make digital signatures allowing you to prove that your files and electronic mail have not been altered.



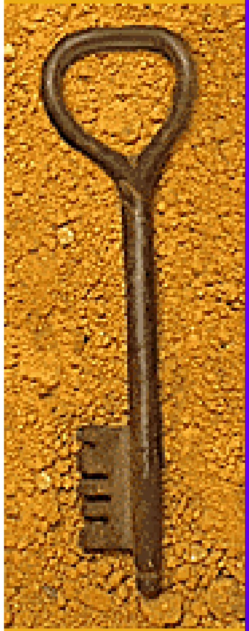
A very brief history of PGP

- 1991 – Phillip Zimmerman releases PGP 1.0 to the world under the GNU license
- 1992 – Version 2.0 is released
  - Uses IDEA for encryption
  - Written by an international team of programmers
  - Infringed on the RSA patent
- 1993 – ViaCrypt PGP 2.4
  - Commercial version



A very brief history of PGP

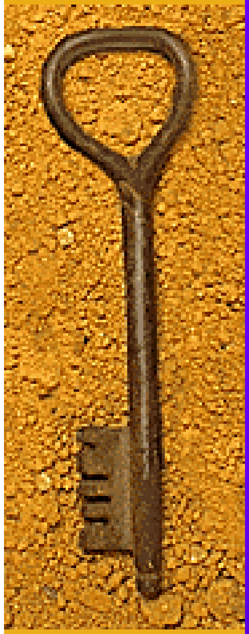
- 1994 – PGP 2.6
  - Distributed by MIT
  - Used RSAREF 2.0
  - License changed to disallow commercial use
  - Distributed with source
  - Not backwards compatible with earlier versions
  - Legal



Cryptography and the US Government

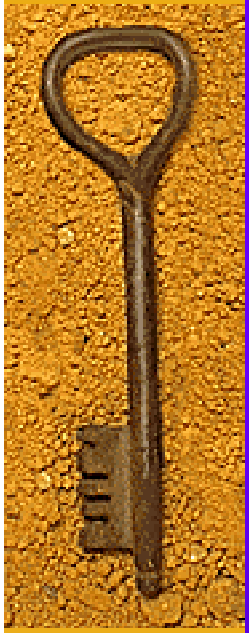
- Cryptography considered to be “Munitions”
- Cryptography could not be exported until very recently (July, 2000)
- Complete regulations can be found at:

  www.bxa.doc.gov/Encryption/Default.htm



Cryptography and international law

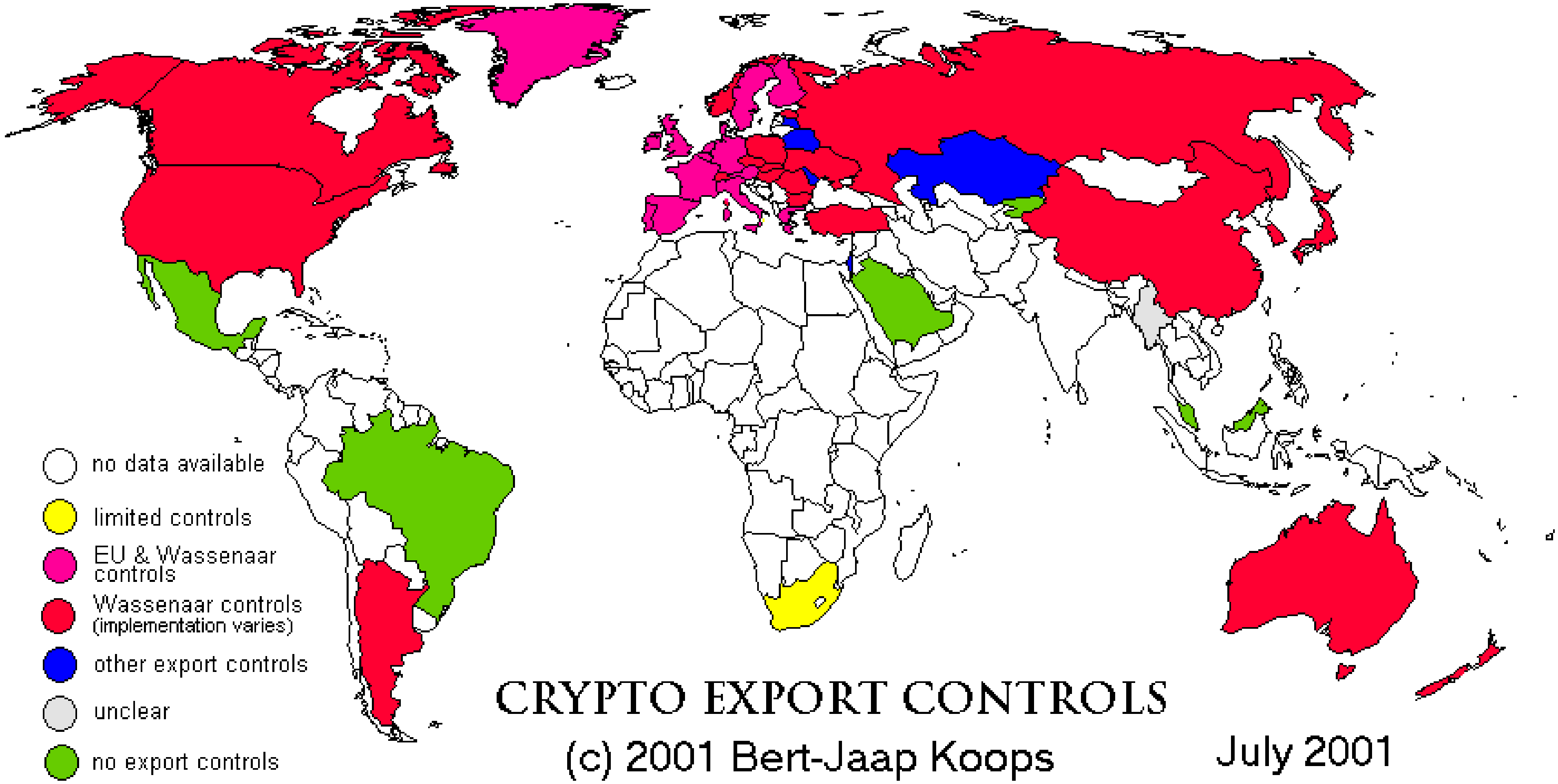
- It varies from country to country
- The crypto law addresses
  - Exporting cryptography
  - Importing cryptography
  - Using cryptography



Cryptography and International Law

- ♦ The next three slides provide an overview of the current situation. These are copyright Bert-Jaap Koops and are used with permission. Detailed information can be found on his web site at:

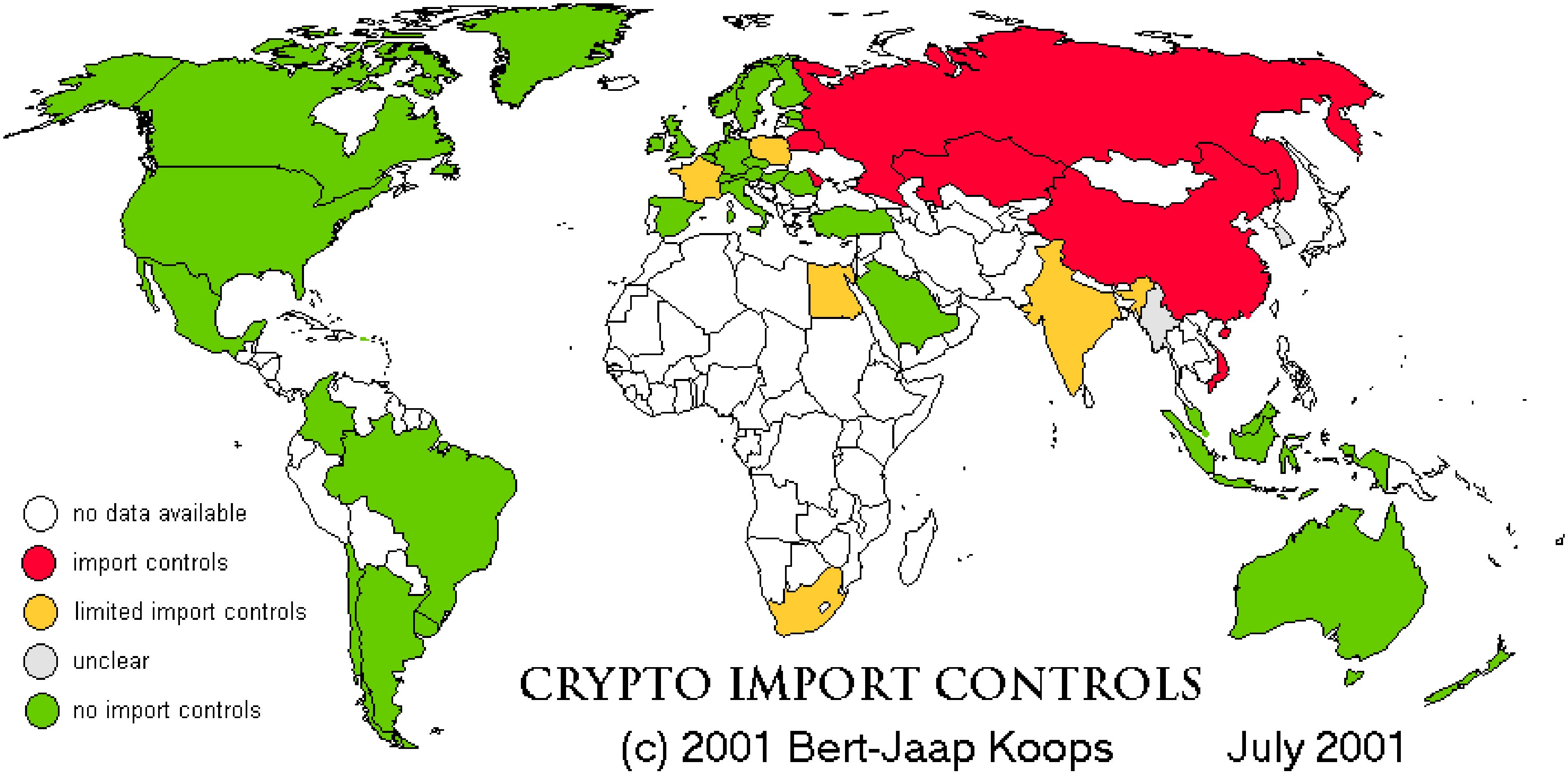
  cwis.kub.nl/~frw/people/koops/lawsurvy.htm



CRYPTO EXPORT CONTROLS

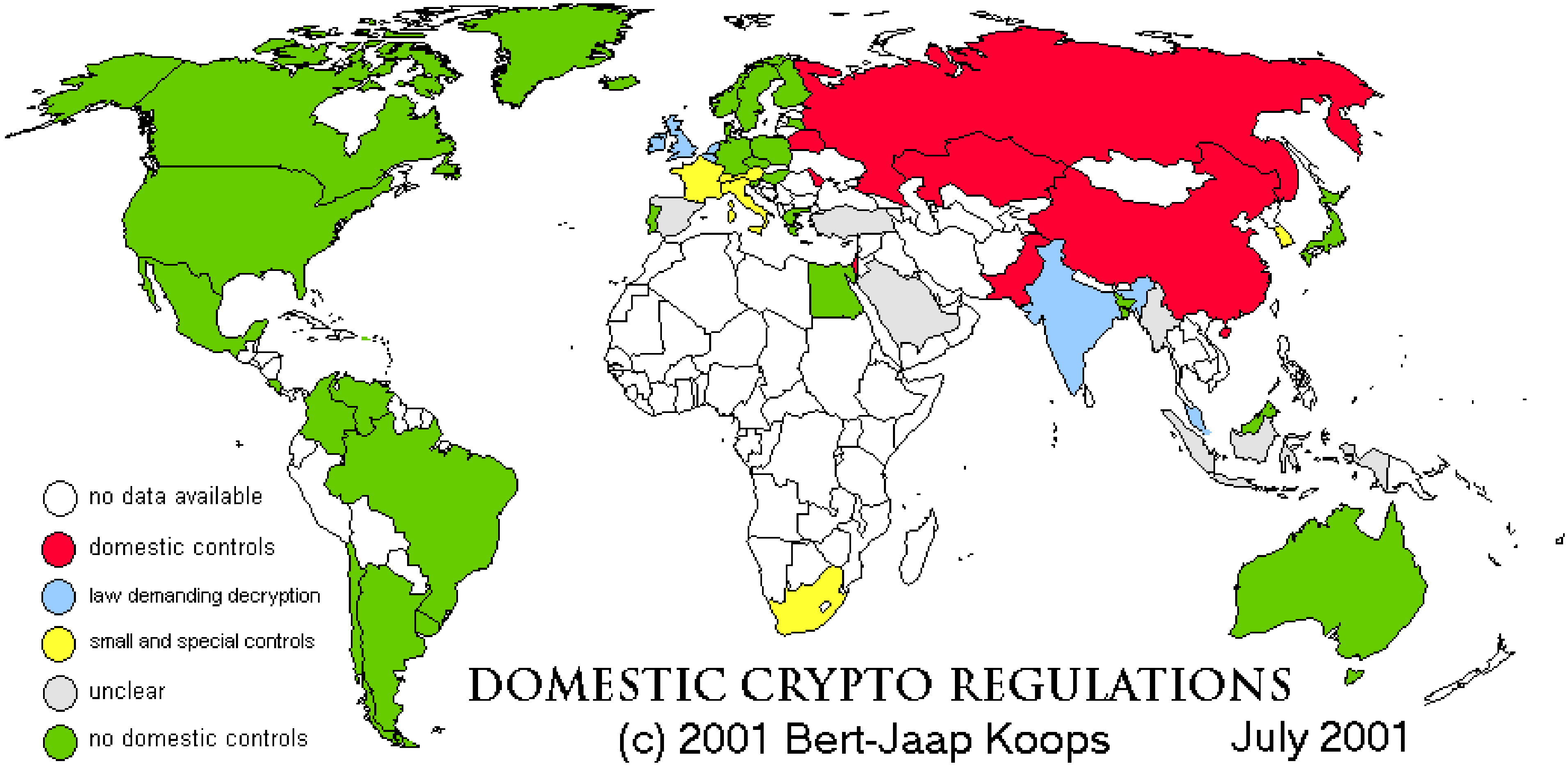
(c) 2001 Bert-Jaap Koops

July 2001



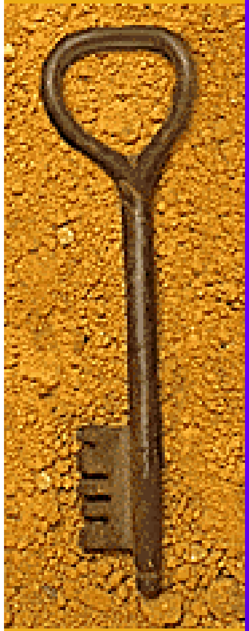
CRYPTO IMPORT CONTROLS

(c) 2001 Bert-Jaap Koops July 2001



DOMESTIC CRYPTO REGULATIONS

(c) 2001 Bert-Jaap Koops July 2001



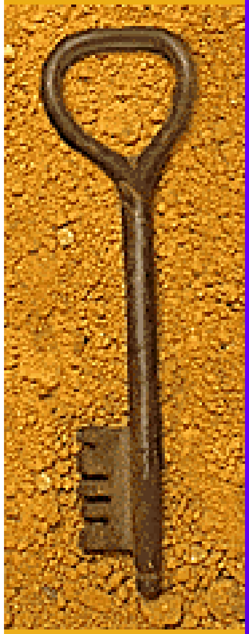
Cryptography and International Law

- Detailed information can be found on the web site at:

  cwis.kub.nl/~frw/people/koops/lawsurvy.htm

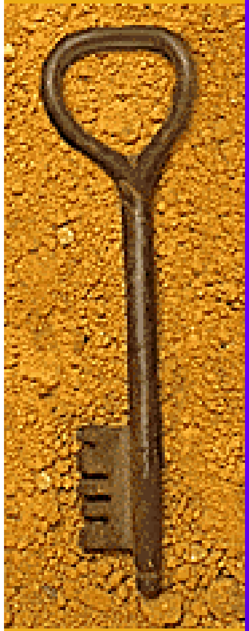
- Detailed information on the Wassenaar Agreement on Export Controls for Conventional Arms and Dual Use Goods and Technologies can be found at:

  www.wassenaar.org



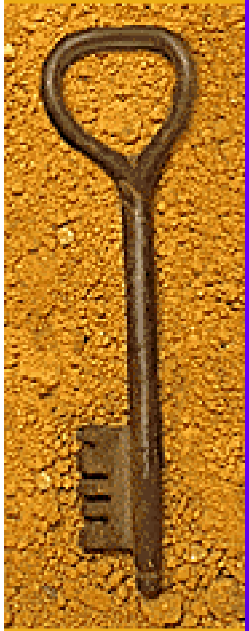
Cryptography and International Law

- As with all web sites, there are no guarantees of correctness. The actual law is what the courts of a specific country say it is.



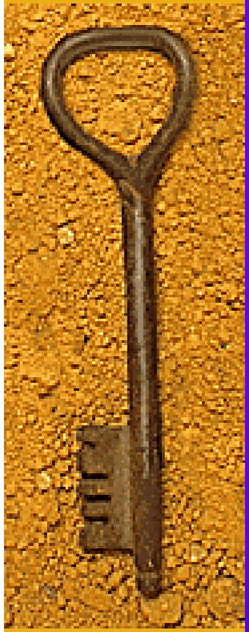
Some Terminology

- Keys
  - Public
  - Private
  - Secret
  - Session



More Terminology

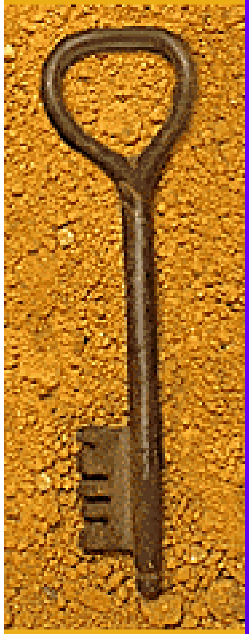
- Key certificates contain:
  - Key
  - Date created
  - One or more e-mail addresses
    - Zero or more digital signatures



More Terminology

- Key rings
  - Public key ring
  - Private key ring
- Pass phrases
- Digital signatures
- Signatures on key certificates

More Terminology



- Key Length

  Length of the key in bits

- Key Type

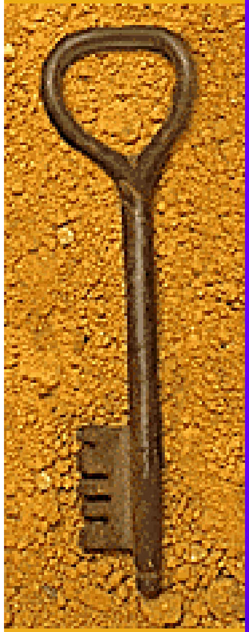
  RSA or Diffie/Hellman

- KeyID

  64 least significant bits of the public key, written as a hexadecimal number

- Key Fingerprint

  MD5 hash of the public key, written in Hex

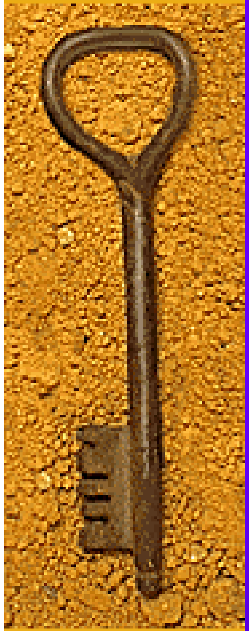


PGP Web of Trust

- Every key has two properties associated with it
- Key Validity

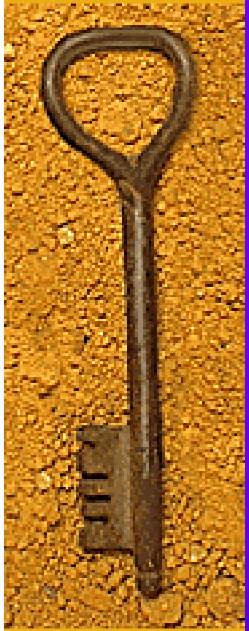
  Is the key valid? Does it really belong to the person whose name appears on it?
- Trust

  Do you trust the person to whom the key belongs to introduce others?



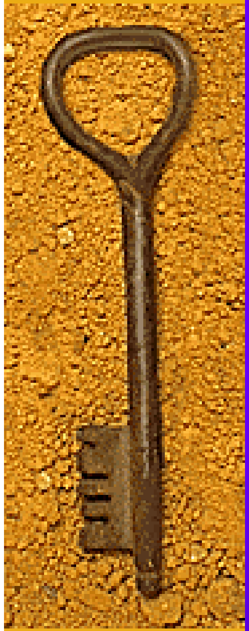
PGP Web of Trust

- What makes a key valid?
- Signatures
  - Exportable
  - Non-exportable (PGP Version 7)
- Your signature
- Signature of a “trusted introducer”



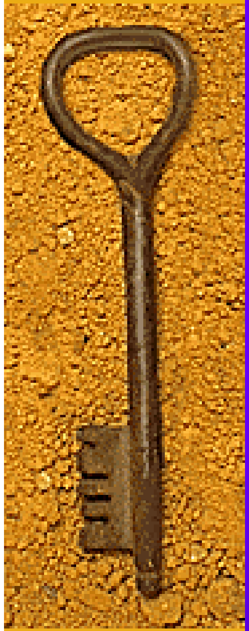
PGP Web of Trust

- What is a Trusted introducer?
- You know them!
  - Full trust
  - Partial trust
  - No trust
- Inherited trust
- Implicit trust



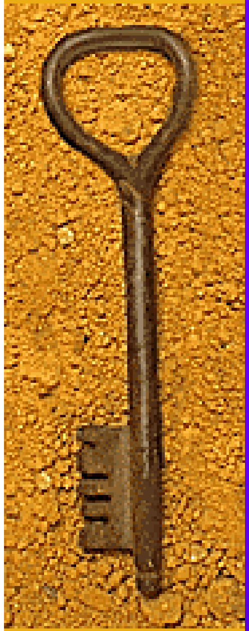
Using PGP

- Command line interface
- Various GUI interfaces to PGP exist
  - Windows
  - X-windows for UNIX
  - Macintosh



Examples

- All examples will be of the command line version of PGP. At the end of the talk, there will be a demo of the windows based GUI client.

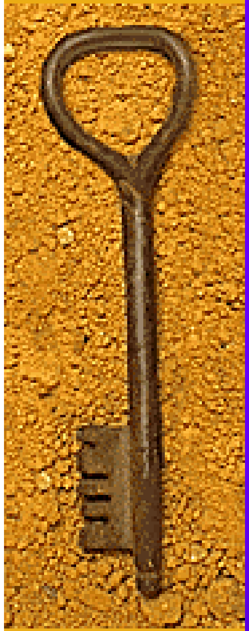


Creating Keys

- The first thing you must do is create your public/private key pair.

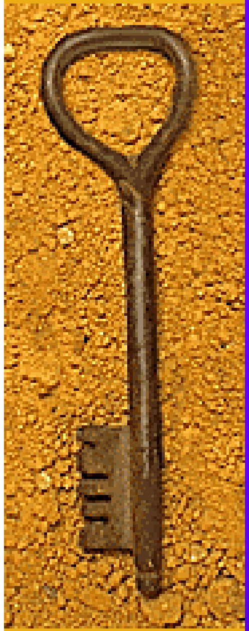
pgp –kg

- Answer prompts
- Reasonable key length is 2048
- Pick a good pass phrase
  - www.pgpi.org/doc/faq/passphrase/



Identifying Keys

- ♦ Keys on your key ring(s) may be identified in one of two ways
- ♦ By UserID
  - – Enter as much of the userid as required to be unique
- ♦ By KeyID
  - – Enter the Hex KeyID – 0x1a2b3c4d



Managing Keys

- Extracting a key from your public key ring

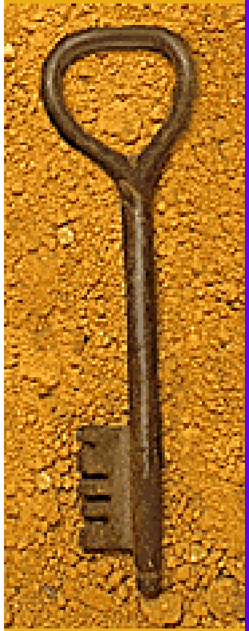
```
pgp –kx KeyID KeyFile
```

- Viewing the keys on your public key ring

```
pgp –kv UserID KeyFile
```

- Viewing more information on the keys on your public key ring

```
pgp –kvc keyfile
```

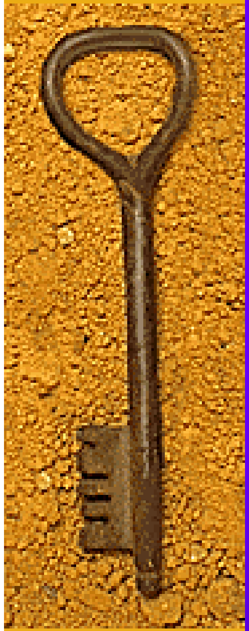


Managing Keys

- Adding a key to your public key ring

```
pgp –ka keyfile
```

When you add a key to your ring you will be asked if you wish to sign it, and what level of trust you wish to assign to it. Do NOT do either unless you are POSITIVE the key belongs to the person who says it’s theirs.



Encrypting Files

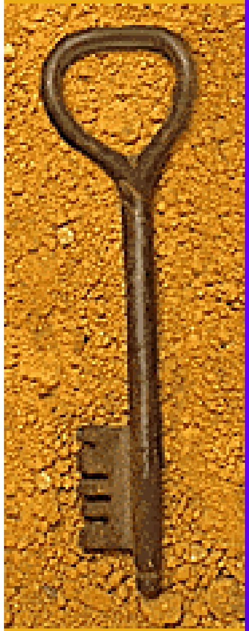
- Encrypt a file to a single recipient

```
pgp –e filename userid
```

- Encrypt a file for multiple recipients

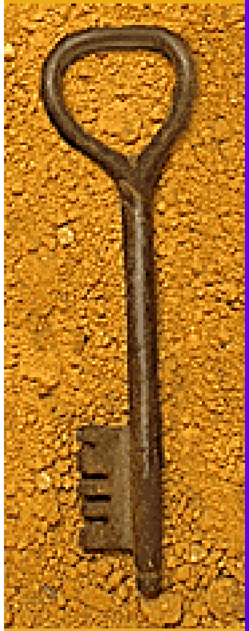
```
pgp –e filename userid1 userid2 userid3
```

You will be prompted for your pass phrase



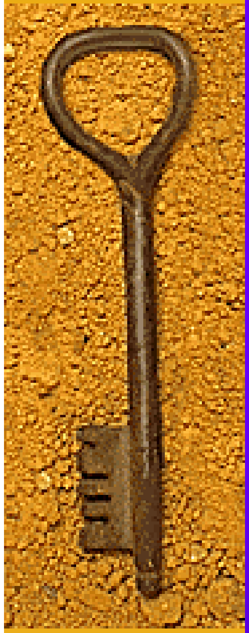
Encrypting files

- You have to have copies of all the recipients public keys on your public key ring
- This will encrypt the input file and create a file named “filename.pgp”



Encrypting files

- If the file contains ascii text, add the –t operand
- If the output file is to be sent via e-mail, add the –a option to get ascii output
  - In this case, the output file will be named “filename.asc”

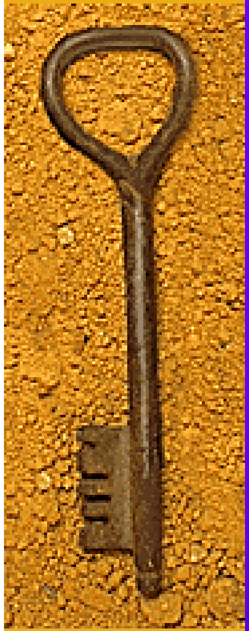


Encrypting files

- Therefore, the usual command to encrypt a file containing text which is to be sent via e-mail to someone else is:

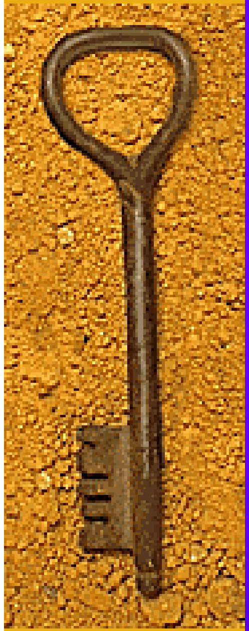
pgp –eat filename userid1 userid2 …

And the output file will be filename.asc



Encrypting E-Mail

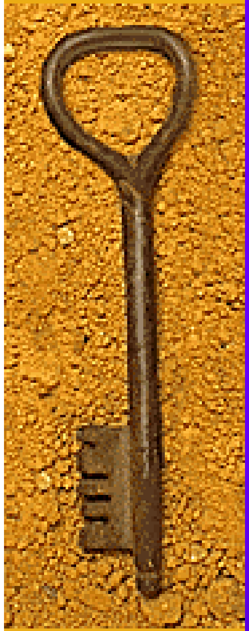
- The old-fashioned way. Create a text file, encrypt this file using the technique on the previous slide and then mail the resulting encrypted file to the recipient(s)
- Use a “PGP aware” mail package



Encrypting E-Mail

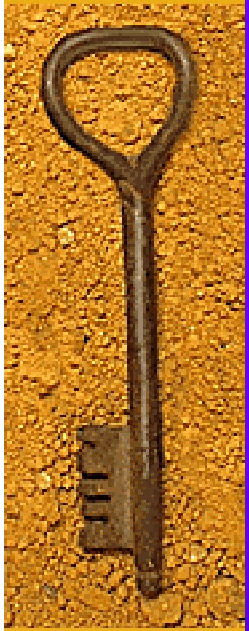
- Use a “PGP plug-in” for your e-mail package

- Use the “Windows” clipboard



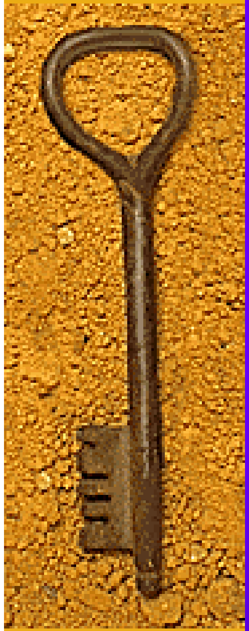
Signing E-Mail

- Digital signatures provide two functions
- Integrity
  - Has the message been altered after it was composed?
- Authentication
  - Did the message really come from who it is supposed to be from?



Signing E-Mail

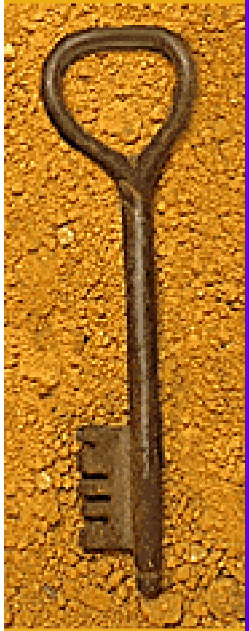
- Signing of e-mail consists of creating a message digest of the message, and then encrypting this message digest with your own PRIVATE key.
- Since only you have the PRIVATE key, then only you could have signed it
- pgp –sta filename



Signing and Encrypting E-Mail

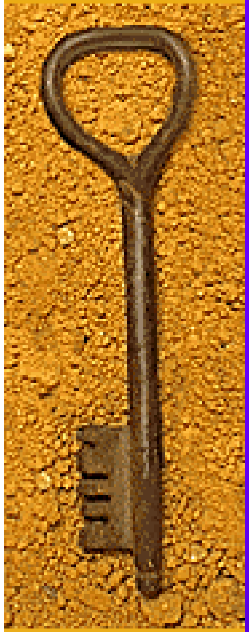
- You can sign and encrypt a message at the same time

```
pgp –seat filename userid1 userid2
```



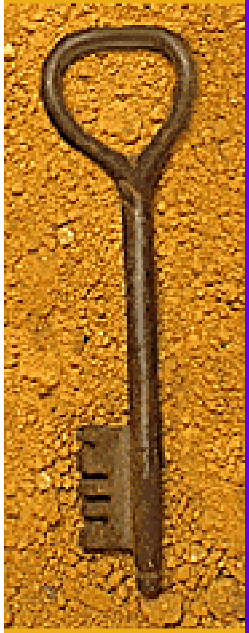
Signing Keys

- Done as a part of the “Web of Trust”
- Two types of signatures
  - Non-exportable
    - Good enough for me personally
    - Can not be exported with the public key
  - Exportable
    - Can be relied upon by the whole world
    - Only if you Absolutely, positively know that the key belongs to the person claiming it.



Distributing Keys

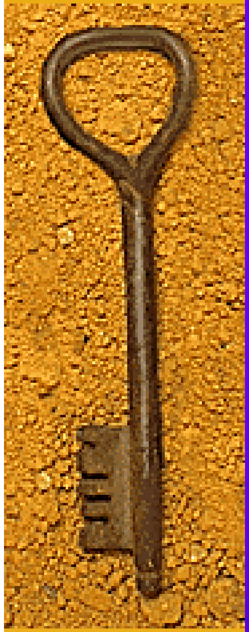
- Only PUBLIC keys are distributed
  - E-Mail
  - UNIX finger command
  - Web pages
  - Key servers



Other Key Actions

- Revoking
  - Done to a PRIVATE key
  - Indicates that the key has been compromised
  - Cannot be un-done
  - Revoked key can not be used to encrypt or sign
  - Requires access to private key and pass phrase

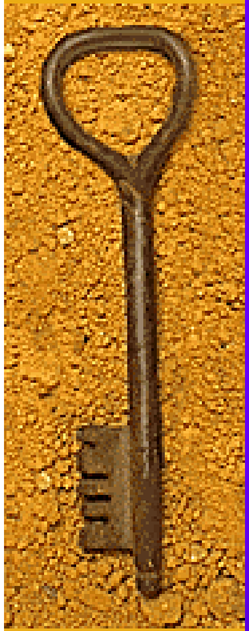
```
pgp –kd userid
```



Other Key Actions

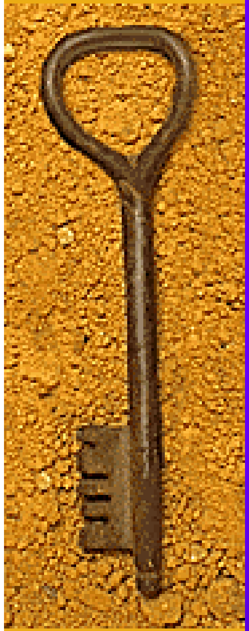
- Revoking
  - – Done to a PUBLIC key
  - – Indicates that you no longer trust this key
  - – Can be un-done
  - – Revoked key can not be used to encrypt
  - – No access to private key and pass phrase

pgp –kd userid



Other Key Actions

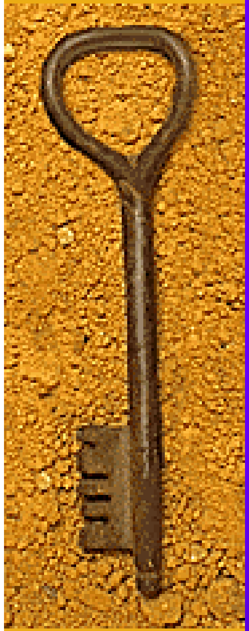
- Escrowing
  - Saving a copy of your private key and pass phrase
  - Why?
    - You might forget your pass phrase
    - You might get hit by a truck
    - Do you want all your encrypted files to die with you?



Other Key Actions

- Escrowing
  - How
    - Copy your secret key ring to a floppy
    - Create a text file on the floppy containing your pass phrase
    - Optionally, encrypt all files with lawyer/solicitor’s public key
    - Put floppy in envelope and seal. Sign name across all seams/flaps of envelope
    - Lock floppy in safe/bank vault

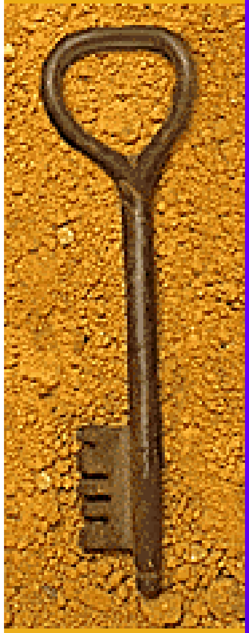
      or
    - Give floppy to lawyer/solicitor



PGP on the S/390

- PGP E-Business Server
  - Commercial Product
  - Full implementation of PGP from NAI

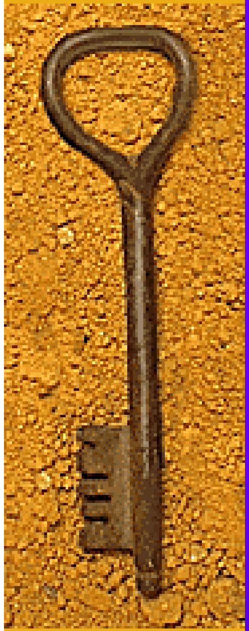
www.pgp.com/products/ebusiness-server-os390/



PGP on the S/390

- ♦ OS/390 Open Server
  - – Freeware
  - – Full implementation of PGP 2.6.3is
  - – Source available
  - – USS
  - – TSO
  - – Batch

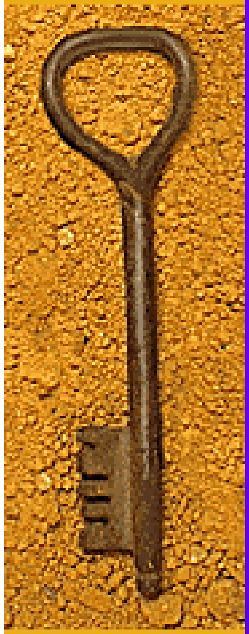
www.nichols.de/os390/pgp/



PGP on the S/390

- MegaCryption/MVS
  - Commercial Product
  - Implements a subset of OpenPGP
  - TSO/REXX
  - Batch

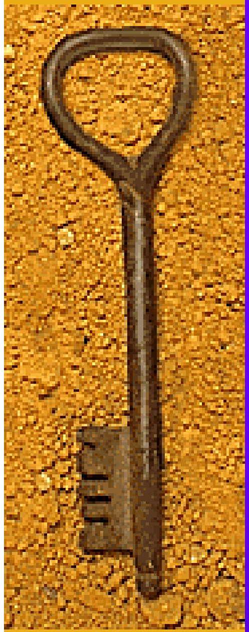
megacryption.hypermart.net



PGP on the S/390

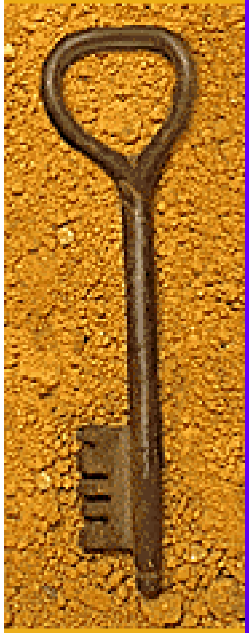
- Linux on S/390
  - Runs any version of PGP which will run on native Linux
  - GPG
  - OpenPGP

- It’s Linux!!!

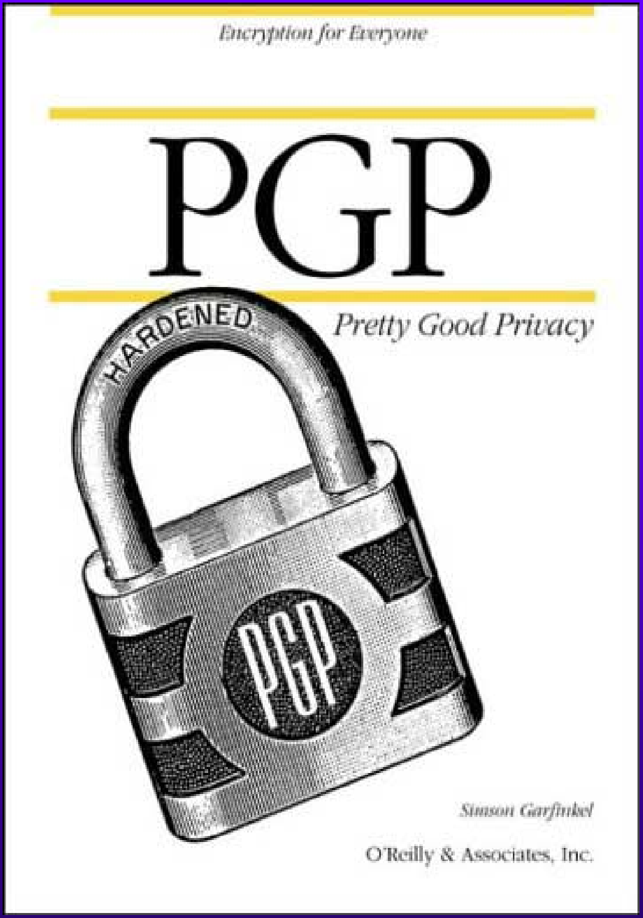


Versions of PGP

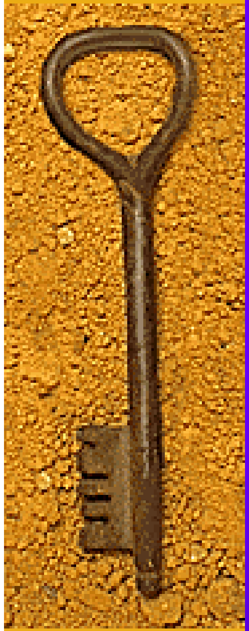
		Latest Version
Windows 9x/NT/2000		7.0.3
MacOS		7.0.3
Command Line	Windows	7.0.3
	UNIX	6.5.8
	MSDos	5.0i
	OS/2	5.0.i / 6.5.1i beta
	Amiga	5.0ib
	Atari	5.0i
Source		6.5.8
GnuPG		1.0.6
Download: www.pgpi.org/products/pgp/versions/freeware		



References

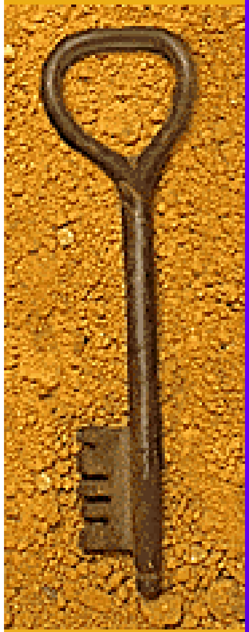


PGP
Simson Garfinkle
O’Reilly & Associates Inc.
ISBN 1-56592-098-8
1995



References

- Bacard, André, The Computer Privacy Handbook, Peachpit Press, 1995, ISBN 1-56609-171-3 (Out of Print)



References

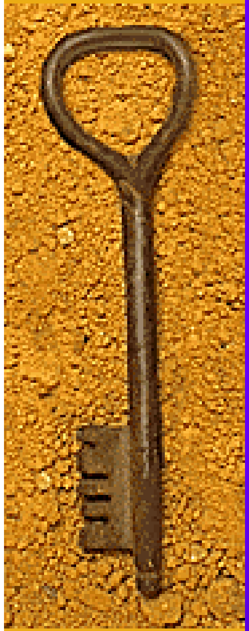
- PGP FAQ

  www.uk.pgp.net/pgpnet/pgp-faq/

- Non Technical PGP FAQ

  www.andrebacard.com/pgp.html

References



- OpenPGP en Français

  www.geocities.com/openpgp/

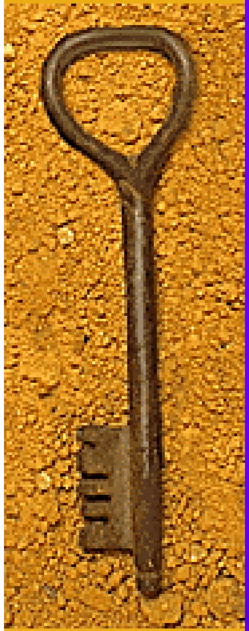
- Nichttechnische Einführung zu PGP

  www.iks-jena.de/mitarb/lutz/anon/pgp.html

- Documentacion acerca de PGP

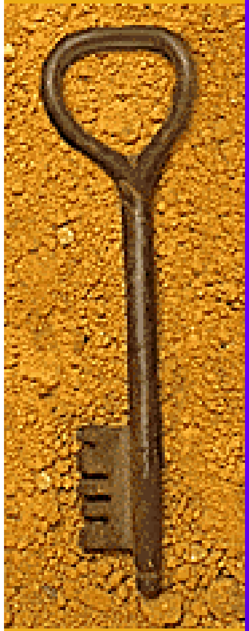
  www.rediris.es/pgp/doc/

- Use the Search Engines to find documentation in other languages



References

- Tom McCune’s PGP pages
  www.mccune.cc/PGP.htm
- Bernie Poole’s guide to PGP for windows
  www.pitt.edu/~poole/PGP.htm
- Francis Litterio’s Cryptography pages
  world.std.com/~franl/crypto/



Other References

- Federation of American Scientists Intelligence Resource Program

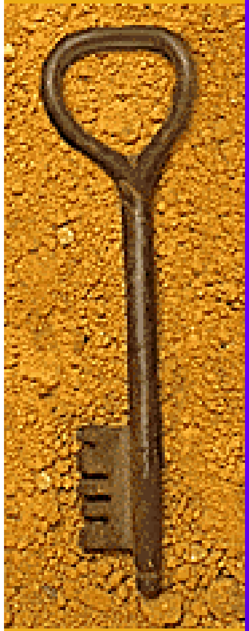
  fas.org/irp/

- Draft report of the temporary committee on the ECHELON interception system of the European Parliament

  fas.org/irp/program/process/europarl_draft.pdf

- RSA Labs Frequently Asked Questions

  www.rsasecurity.com/rsalabs/faq/

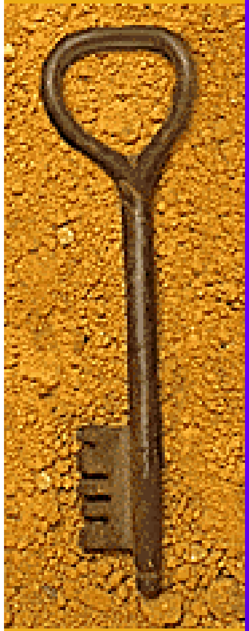


Other References

- One place shopping for information about PGP

http://Cryptography.org/getpgp.html

- Contains Many, Many links to PGP sites and information

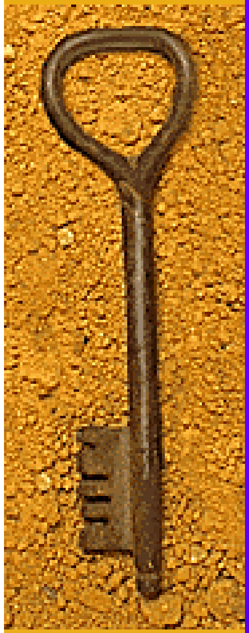


My Presentations

- Current copies of this and my other presentations can be found on my website

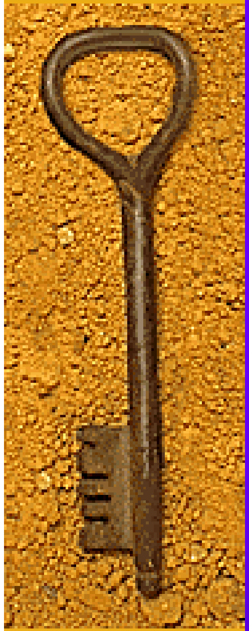
http://www.harold.pritchett.org

Follow the link to “my presentations”



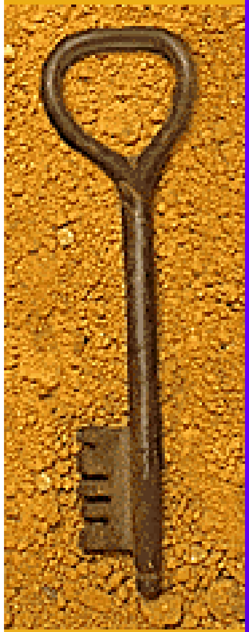
Key Signing Party

- Where
  - This room on Friday at 11:00
- Who
  - Anyone who has a key and wants it signed
- What should I do before hand
  - Create a key and send it to the moderator, before noon on Thursday



Key Signing Party

- ♦ How do I send my key to the moderator?
- ♦ Create the key, and export your public key as an ASCII file
- ♦ Mail the file to harold@uga.edu. Use the subject “pgp key signing party”
- ♦ Do this before noon on Thursday



Key Signing Party

- What should I bring with me
  - A picture ID (Driver's license, Passport, etc.)
  - A printout of my key's four parameters
    - Key type
    - Key length
    - Key ID
    - Key Signature
- More information at:

  http://www.arches.uga.edu/~harold/keysign.html

Questions?

Session 5516