Sicurezza nel Sistema di EMail

Electronic Mail Security

Agenda:

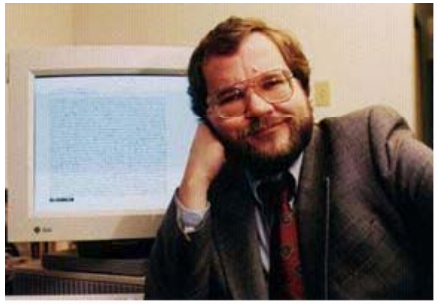
- **Introduction to PGP**
- **5 PGP Services**
- **Key Management**
- **Use of Trust**
- **Demo Of PGP In Use**

Pretty Good Privacy

- Definito nel 1991 da Phil Zimmermann
- Fornisce servizi di confidentiality e authentication per i protocolli di email e per le applicazioni di archiviazione di file

Phil Zimmermann

- Target of three year criminal investigation
- Gave software away to friend who put it on the Internet in 1991
- Intended to give individuals "the right to be let alone"
- US export restrictions violated – same class as munitions and nuclear weapons
- Government dropped the case in 1996



"PGP has spread like a prairie fire, fanned by countless people who fervently want their privacy restored in the information age"
- Phil Zimmermann, testifying before the US Senate, 1996

Pretty Good Privacy

- Seleziona i migliori algoritmi crittografici disponibili
- Li integra in un'applicazione general purpose
- Il codice sorgente e la documentazione sono liberamente accessibili su Internet
- Esiste un accordo commerciale con la compagnia Viacrypt per la versione low cost

Notazione

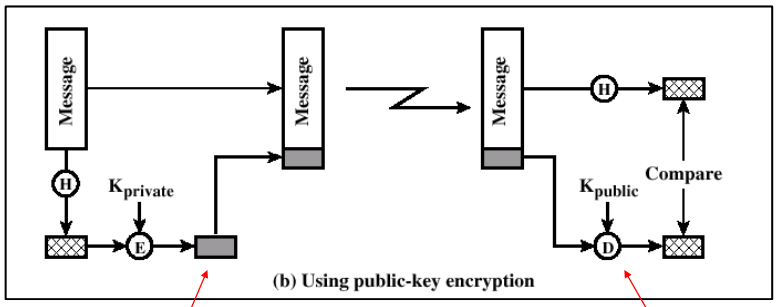
K_S = session key used in conventional encryption
KR_a = private key of user A, used in public key encryption
KU_a = public key of user A, used in public key encryption
EP = public-key encryption
DP = public-key decryption
EC = conventional encryption
DC = conventional decryption
H = hash function
|| = concatenation
Z = compression using ZIP algorithm
R64 = conversion to radix 64 ASCII format

Servizi PGP

authentication
confidentiality

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key, and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key, and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversion.
Segmentation	—	To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

Recall One Way Hash Function



Less computation since message does not have to be encrypted

Recall SHA-1 Secure Hash Function

- Developed by NIST in 1995
- Input is processed in 512-bit blocks
- Produces as output a 160-bit message digest
- *Every bit of the hash code is a function of every bit of the input*
- Very secure – so far!

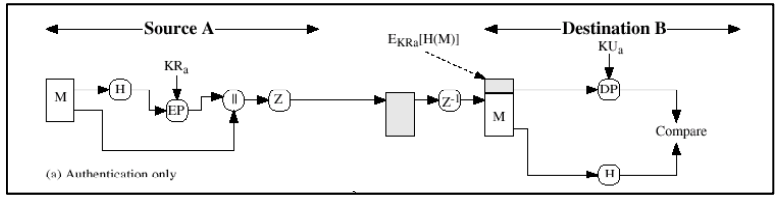
Autenticazione (1/2)

- Il Sender
  - crea il messaggio
  - genera il relativo hash code con SHA-1
  - l'hash code viene crittografato (RSA) con la propria chiave privata e il risultato viene posto in testa al messaggio

Autenticazione (2/2)

- Il Receiver
  - decritta mediante la chiave pubblica del sender e ottiene l'hash code
  - genera un nuovo hash code per il messaggio e lo confronta con quello ricevuto
  - Se il confronto ha esito positivo, allora il messaggio è autentico

PGP Cryptographic Functions



Recall Other Public Key Algorithms

- Digital Signature Standard (DSS) – makes use of SHA-1 and presents a new digital signature algorithm (DSA)
- Only used for digital signatures not encryption or key exchange

Summary of 5 PGP Services

authentication →

confidentiality →

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key, and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key, and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversion.
Segmentation	—	To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

Recall CAST-128

- 1997, Entrust Technologies
- RFC 2144
- Extensively reviewed
- Variable key length, 40-128 bits
- Used in PGP

Recall Conventional Encryption Algorithms

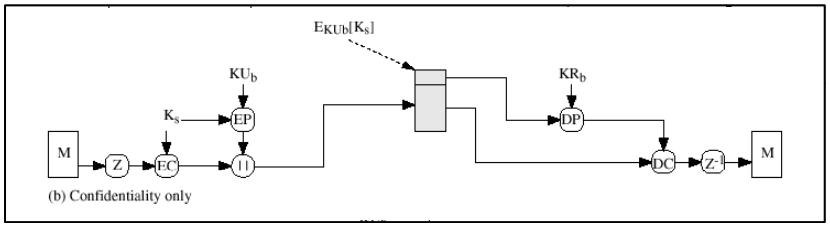
Algorithm	Key Size (bits)	Block Size (bits)	Number of Rounds	Applications
DES	56	64	16	SET, Kerberos
Triple DES	112 or 168	64	48	Financial key management, PGP, S/MIME
AES	128, 192, or 256	128	10, 12, or 14	Intended to replace DES and 3DES
IDEA	128	64	8	PGP
Blowfish	variable to 448	64	16	Various software packages
RC5	variable to 2048	64	variable to 255	Various software packages

We have choices in PGP for confidentiality!

Confidenzialità

- Il Sender crea un messaggio e un numero random di 128 bit, quale session key
- Il messaggio è crittografato usando CAST-128 con la session key
- La session key è crittografato con la public key del destinatario e il risultato viene posto in testa al messaggio
- Il destinatario decritta con la propria private key e ottiene la session key, usata poi per decrittare il messaggio

PGP Cryptographic Functions



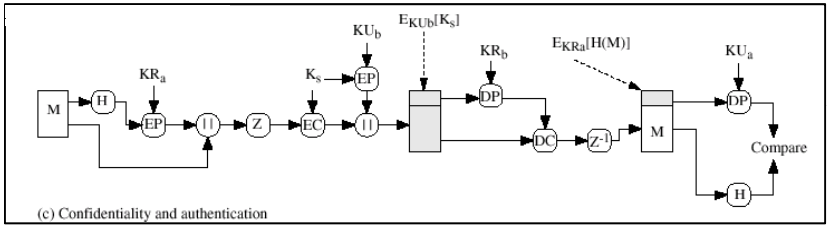
Ancora Confidenzialità

- Alternative per la crittografia convenzionale: RSA or Diffie-Hellman (ElGamal)
- Gli algoritmi convenzionali sono molto più veloci
- Ogni messaggio è un evento indipendente one-time con la sua specifica chiave
- $768 \leq$ key size ≤ 3072

Confidenzialità & Autenticazione

- Possono essere usati entrambi i servizi per lo stesso messaggio
- Prima viene generate la signature per il messaggio e viene posta in testa
- Il messaggio è poi crittografato con la session key
- La session key è infine crittografata con la public key del destinatario

PGP Cryptographic Functions



Summary of 5 PGP Services

authentication →

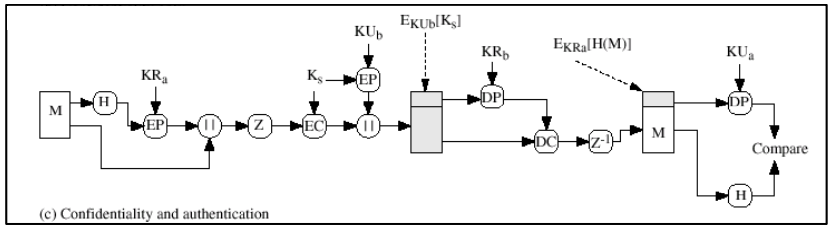
confidentiality →

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key, and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key, and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversion.
Segmentation	—	To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

Compressione

- PGP usa ZIP per comprimere il msg dopo aver applicato la signature, e prima della fase di crittografia
  - È meglio firmare un messaggio non compresso
  - Si aumenta la sicurezza crittografando il msg compresso

PGP Cryptographic Functions



Summary of 5 PGP Services

authentication →

confidentiality →

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key, and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key, and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversion.
Segmentation	—	To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

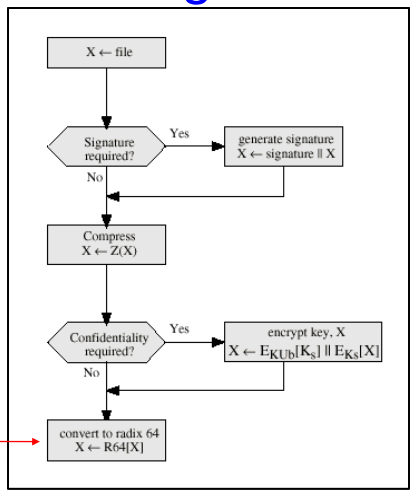
Compatibilità

- Il blocco o una sua parte consiste di uno stream di ottetti (di 8 bit)
- Molti sistemi di mail permettono solo codice ASCII
- PGP converte lo stream binario in stream di caratteri ASCII

Stream Of Printable ASCII Chars

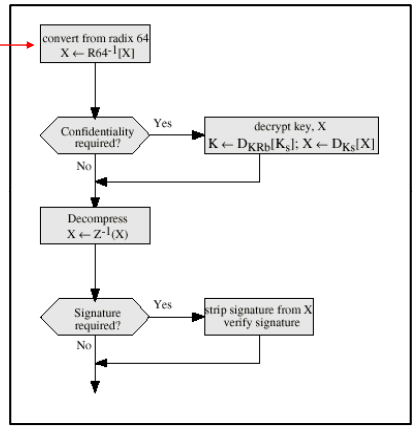
```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3i
mQBNAi23Dv0AAAECAMm6GNU3nqebKr3HW/fmrEhMlrFkwuZ6KHIYEat92nYfQIUj
lRLgj3TPHTRIMbswyTdaIJA7OvkSgxETLBCExX0ABRG0K0FuZHJlYXMgUmllZ2Vy
IDwxMDAxMTEuMzU0MEBjb21wdXNlcnZlLmNvbT4=
=8t7f
-----END PGP PUBLIC KEY BLOCK-----
```


Generic Transmission Diagram



Generic Reception Diagram

ASCII text to binary



Summary of 5 PGP Services

authentication

confidentiality

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key, and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key, and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversion.
Segmentation	—	To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

Segmentazione

- Ci sono restrizioni per la lunghezza massima dei messaggi email
- PGP suddivide automaticamente un msg che eccede la lunghezza max in segmenti più piccolo, e li invia separatamente
- PGP riassembla il blocco originale a destinazione

Summary of 5 PGP Services

- **Authentication**
- **Confidentiality**
- **Compression**
- **E-Mail Compatibility**
- **Segmentation**

PGP Cryptographic Keys

- Chiavi convenzionali one-time session
- **Public** Keys
- **Private** Keys

Requisiti per le Chiavi

- Serve un mezzo per generare chiavi di sessione non predicibili
- Deve essere permesso all'utente di avere più coppie di chiavi pubblica-privata
- Ogni entità PGP deve registrare un file con le proprie chiavi e le coppie pubblica-privata dei suoi corrispondenti

Session Key Generation

- Numeri casuali a 128 bit sono generate con il CAST-128
- L'input è costituito da uno stream random a 128 bit basato su un tasto pigiato dall'utente
- Produce una sequenza di chiavi di sessione impredicibile

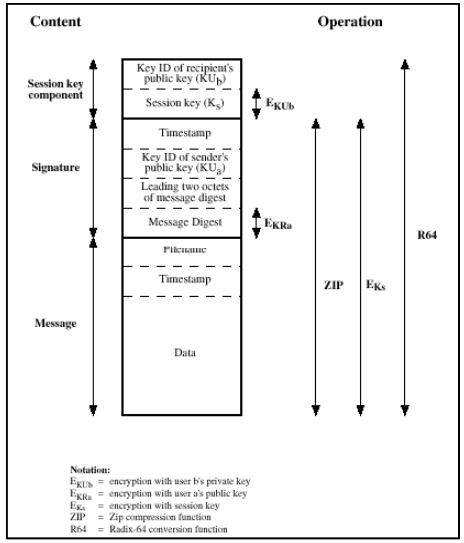
Identificatori di Chiavi

- Servono per identificare le chiavi
- PGP assegna un key ID a ogni chiave pubblica

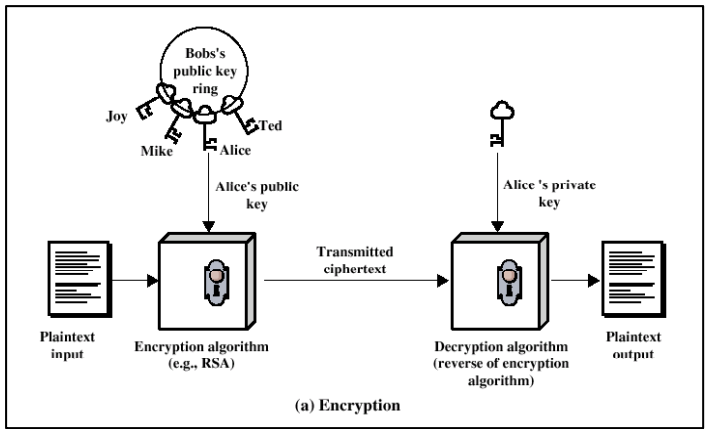
Messaggio Trasmesso

- In un messaggio ci sono 3 componenti principali
- Message component – I dati trasmessi più filename e timestamp
- Signature component – timestamp, message digest, I due ottetti principali del MD (checksum), Key ID della chiave pubblica del mittente
- Session key component – chiave di sessione più ID della chiave pubblica del destinatario usata nella session key

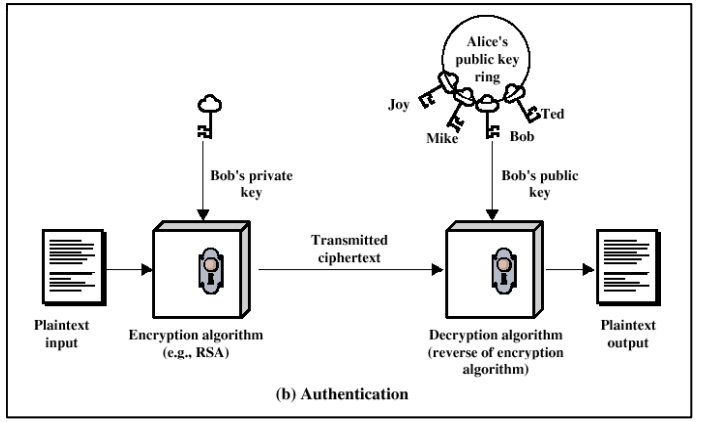
PGP Format



Recall Public Key Encryption



Recall Public Key Authentication



Key Rings

- PGP fornisce due strutture dati a ogni nodo:
  - Coppie delle chiavi pubblica e private
  - Chiavi pubbliche degli altri nodi
- Rispettivamente Private-Key Ring e Public-Key Ring
- I ring possono essere visti come tabelle
  - Ogni riga rappresenta una coppia pub/priv key

Key Ring Structure

Private Key Ring

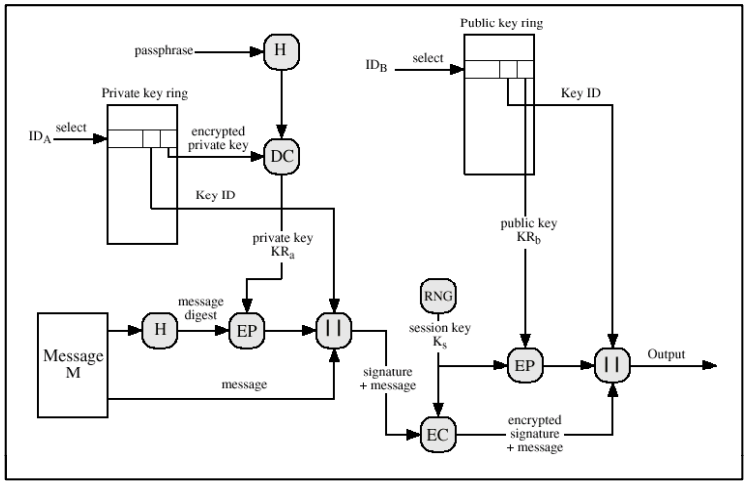
Timestamp	Key ID*	Public Key	Encrypted Private Key	User ID*
. . .	. . .	. . .	. . .	. . .
T_i	$KU_i \bmod 2^{64}$	KU_i	$E_{H(Pi)[KR_i]}$	User i
. . .	. . .	. . .	. . .	. . .

Public Key Ring

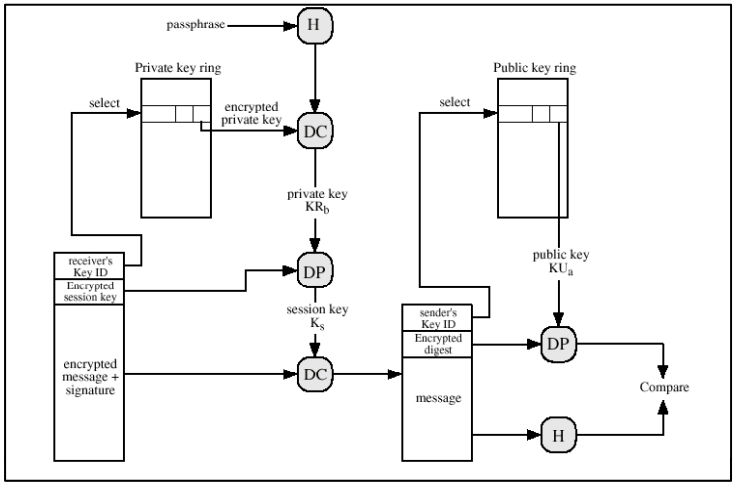
Timestamp	Key ID*	Public Key	Owner Trust	User ID*	Key Legitimacy	Signature(s)	Signature Trust(s)
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
T_i	$KU_i \bmod 2^{64}$	KU_i	$trust\_flag_i$	User i	$trust\_flag_i$		
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

* = field used to index table

PGP Message Generation



PGP Message Reception



Public Key Management

- Physically get the key from *B*
- Verify a key by telephone
- Obtain *B's* public key from a mutually trusted individual *D*
- Obtain *B's* public key from a trusted certifying authority

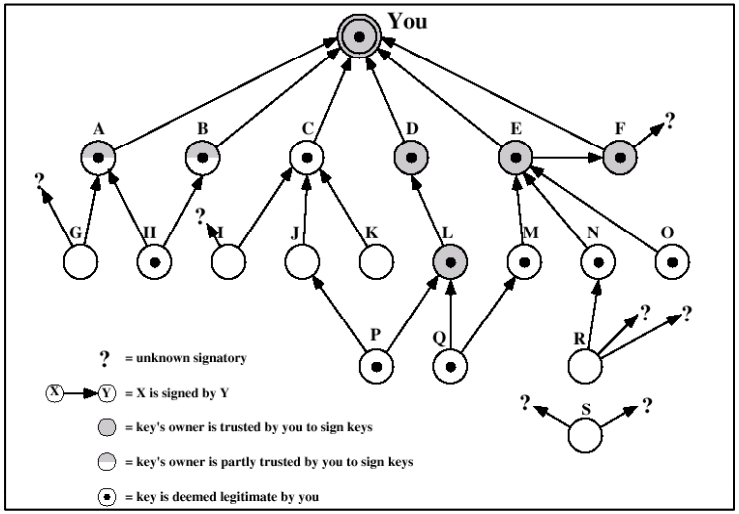
Uso dei Trust

- Associato a ogni chiave pubblica c'è:
  - un campo di key legitimacy - indica che PGP ritiene che quella sia una public key valida
  - Un campo di Signature trust – livello di affidabilità di chi firma assegnato dagli altri utenti
  - Campo di Owner trust – grado di affidabilità di questa public key
- Sono tutti contenuti in una struttura indicate come trust flag byte

Trust Flag Byte Contents

(a) Trust Assigned to Public-Key Owner (appears after key packet; user-defined)	(b) Trust Assigned to Public Key/User ID Pair (appears after User ID packet; computed by PGP)	(c) Trust Assigned to Signature (appears after signature packet; cached copy of OWNERTRUST for this signator)
OWNERTRUST Field —undefined trust —unknown user —usually not trusted to sign other keys —usually trusted to sign other keys —always trusted to sign other keys —this key is present in secret key ring (ultimate trust) BUCKSTOP bit —set if this key appears in secret key ring	KEYLEGIT Field —unknown or undefined trust —key ownership not trusted —marginal trust in key ownership —complete trust in key ownership WARNONLY bit —set if user wants only to be warned when key that is not fully validated is used for encryption	SIGTRUST Field —undefined trust —unknown user —usually not trusted to sign other keys —usually trusted to sign other keys —always trusted to sign other keys —this key is present in secret key ring (ultimate trust) CONTIG bit —set if signature leads up a contiguous trusted certification path back to the ultimately trusted keyring owner

PGP Trust Model Example



Revoca di Public Keys

- Un utente potrebbe voler revocare la sua public key
  - A esempio, perché sospetta sia compromessa o usata per troppo tempo o perché si è persa la private key
- Produce un certificate di revoca firmato opportunamente

Download PGP

- http://www.pgpi.org/download/gnupg/Windows version is: GnuPG 1.2.2
- http://enigmail.mozdev.org/download.html Enigmail download

Generating Keys

- Type: `gpg -gen-key`
- You should end up with something like this: