CS-630: Cyber and Network Security

Lecture # 7: Email Security (Email Sender Authentication, PGP and S/MIME)

Prof. Dr. Sufian Hameed

Department of Computer Science

FAST-NUCES

Overview

- *Intro to Email and SMTP*
- *Sender Authentication Problem*
- *Email Sender Authentication Standards*
  - *Sender Policy Framework (SPF)*
  - *SenderID*
  - *Domain Key Identified Mail (DKIM)*
- *Improvements*
- *Email Security*
  - *S/MIME*
  - *PGP*

Electronic Mail

- Email or Electronic mail is the biggest service being used over the Internet today.
- **44 –** Years since the first email was sent, in 1971.
- The *de facto* standard for e-mail transmissions across the Internet is Simple Mail Transfer Protocol (SMTP).
- SMTP is defined in RFC 821 and it is a relatively simple, text-based protocol.
- Not entirely secure thus vulnerable to SPAM.

Email Statistics

In 2010

- **107 trillion –** The number of emails sent on the Internet.
- **294 billion –** Avg. number of email messages per day.
- **1.88 billion –** The number of email users worldwide.
- **480 million –** New email users since the year before.
- **89.1% –** The share of emails that were spam.
- **262 billion –** The number of spam emails per day (assuming 89% are spam).

***royal.pingdom.com**

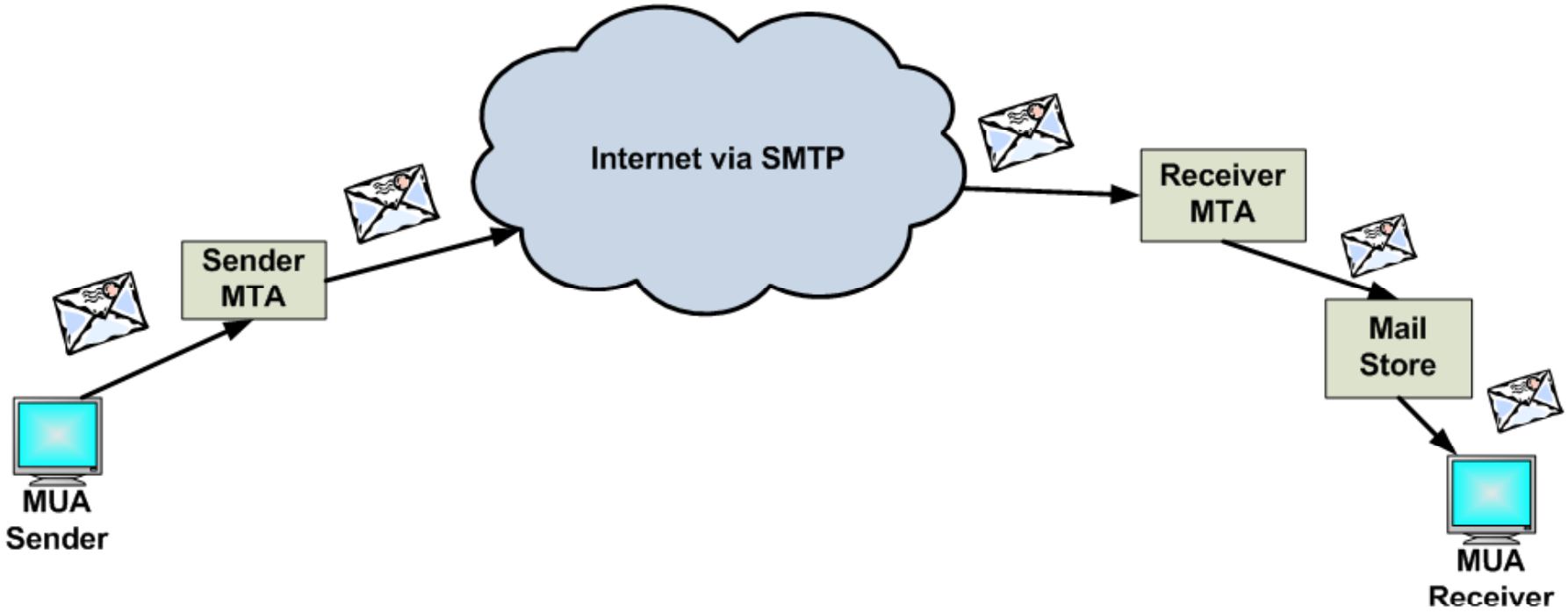
Email Statistics

In 2011

- **3.146 billion** – The number of email accounts worldwide.
- **25%** – Share of email accounts that are corporate.
- **27.6% –** Microsoft Outlook was the most popular email client.
- **19% –** Percentage of spam emails delivered to corporate email inboxes despite spam filters.
- **112 –** Number of emails sent and received per day by the average corporate user.
- **425 million –** Total number of Gmail users (largest email service in the world as of June 2012).
- **$44.25 –** The estimated return on $1 invested in email marketing in 2011.

***royal.pingdom.com**

Basic SMTP Architecture



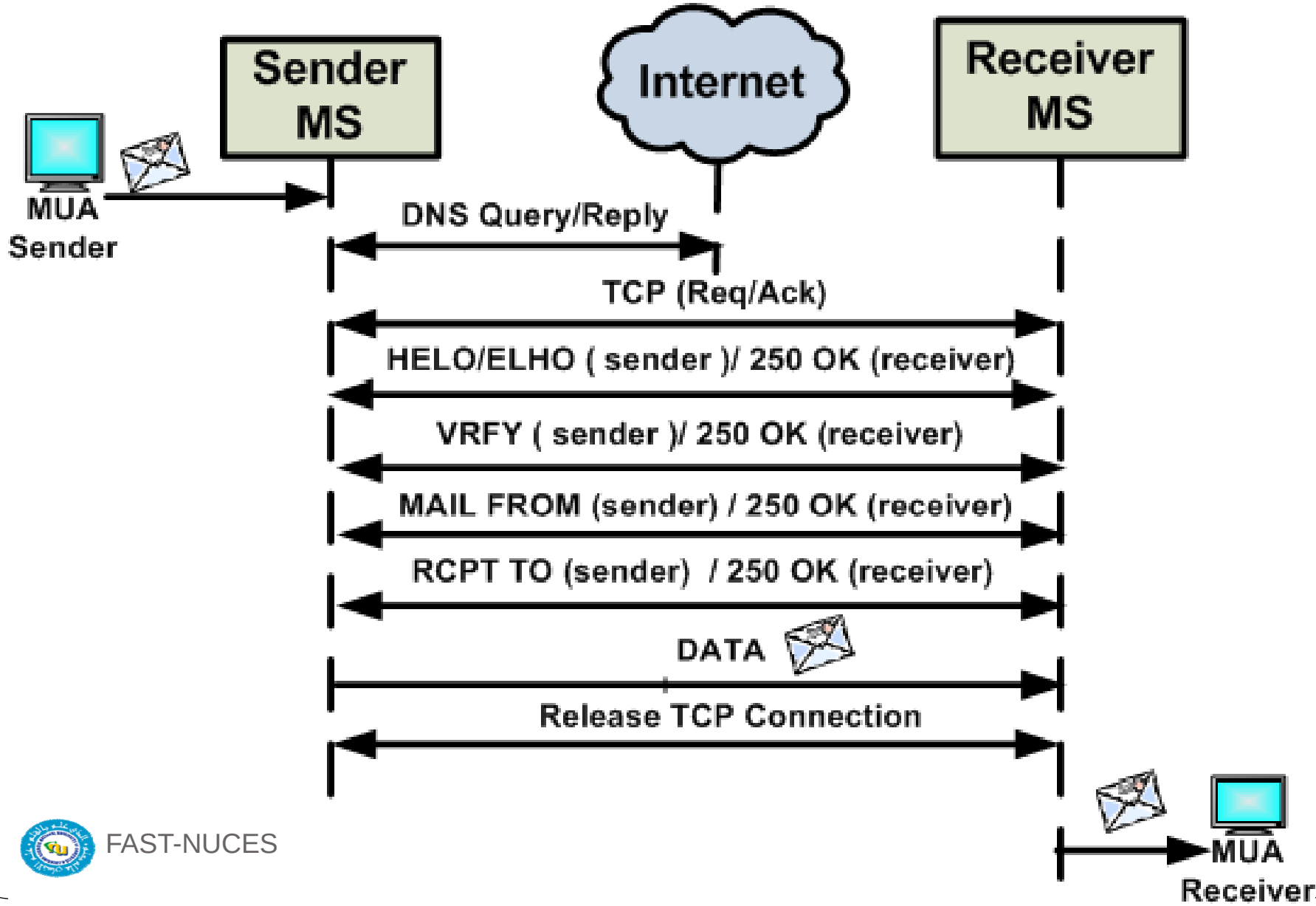
SMTP clients and servers have two main components

- **Mail User Agent (MUA)** – Prepares the message, encloses it in an envelope (ex. MS-Outlook, Thunderbird).
- **Mail Transfer Agent (MTA)** – Transfers the mail across the internet (ex. Sendmail, Exim).

Basic SMTP Commands

- HELO/ELHO: sender’s host domain name
- VRFY: name to be verified
- MAIL FROM: email address of sender
- RCPT TO: email of Indented recipient
- DATA: header and body of message
- QUIT:

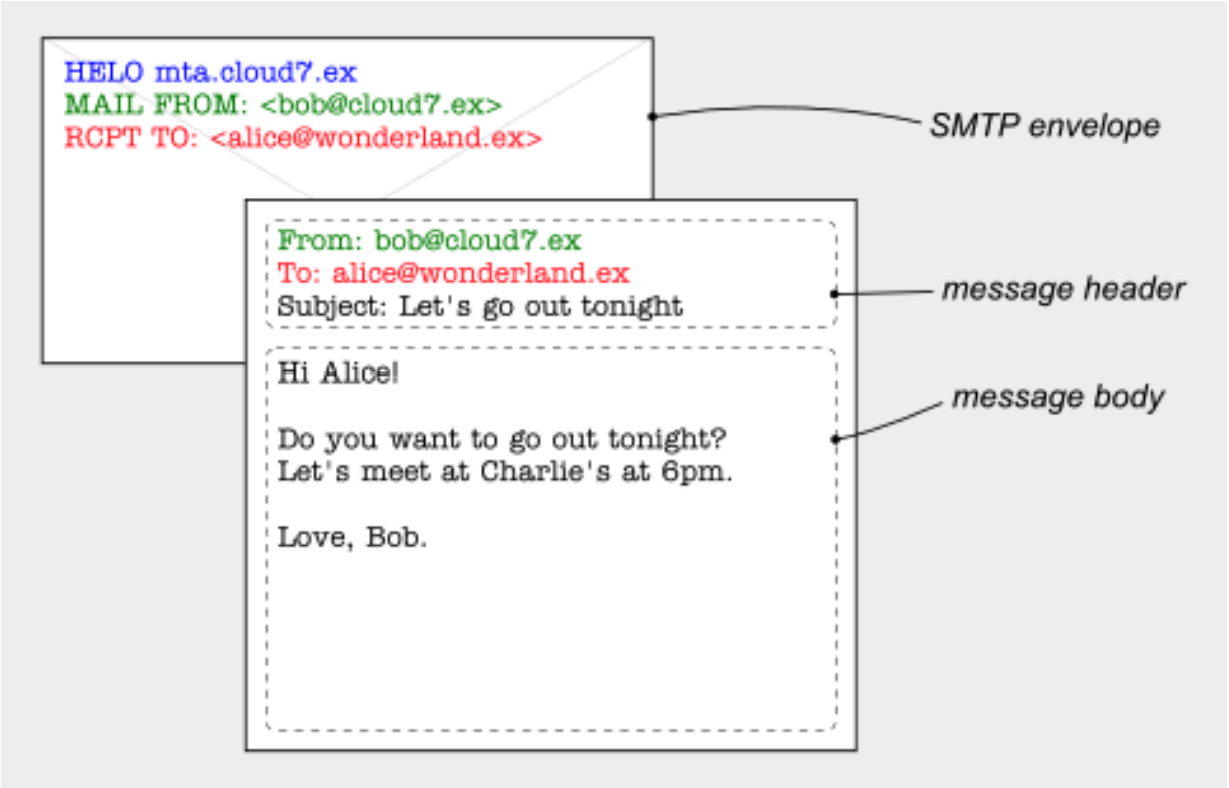
How SMTP Works



Format of an Email

Consists of two parts

- SMTP Envelope
- Data
  - Header
  - Body



Limitations of SMTP

- SMTP is continuously evolving, but when it was designed, in the early 1980s, there was no cause to consider security.
- The original SMTP specification did not include a facility for authentication of senders, making SMTP vulnerable to *From* address forgery.
- Modifying SMTP extensively, or replacing it completely, is not believed to be practical, due to the network effects of the huge installed base of SMTP.

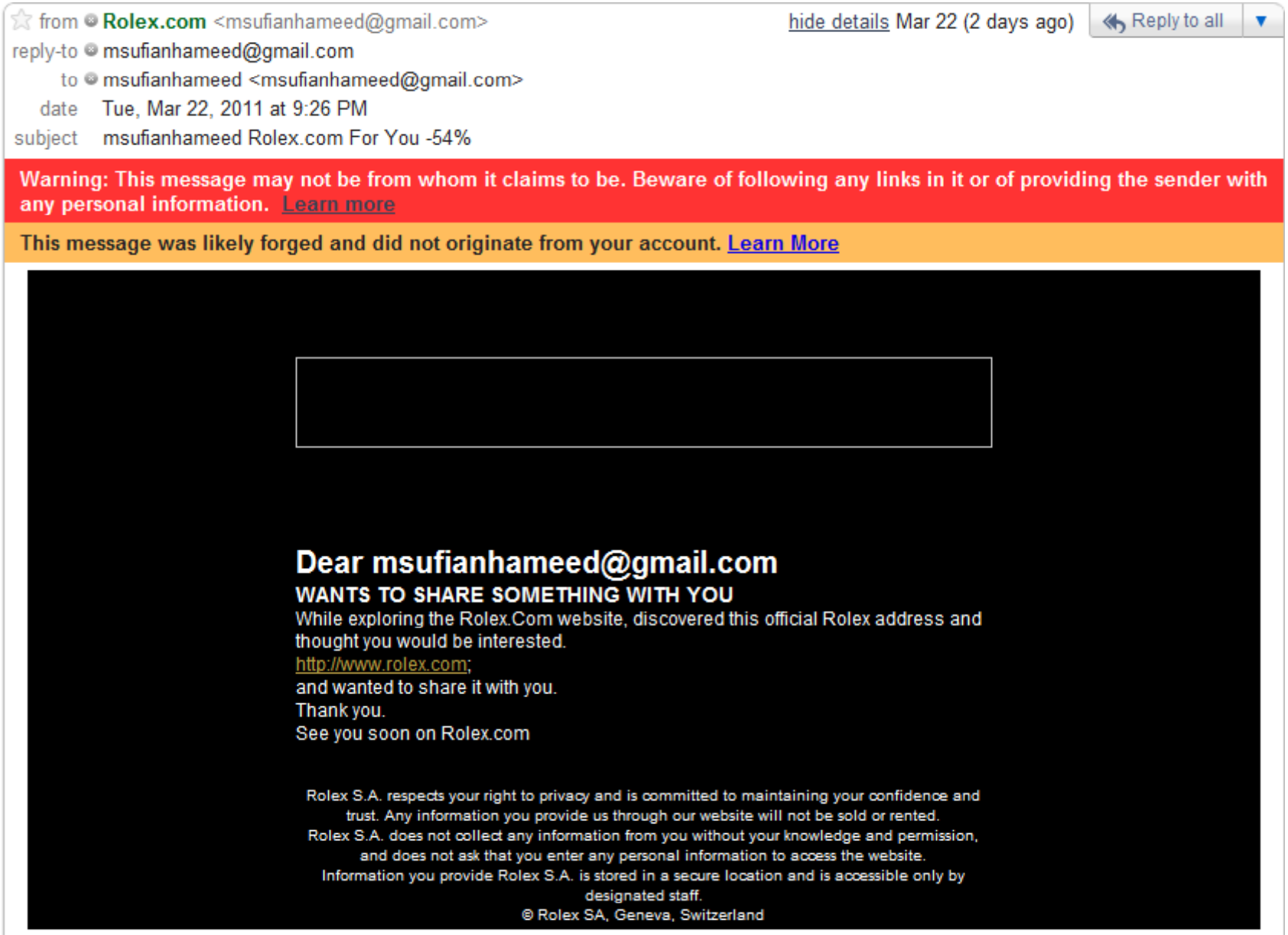
Email Spam

Spam

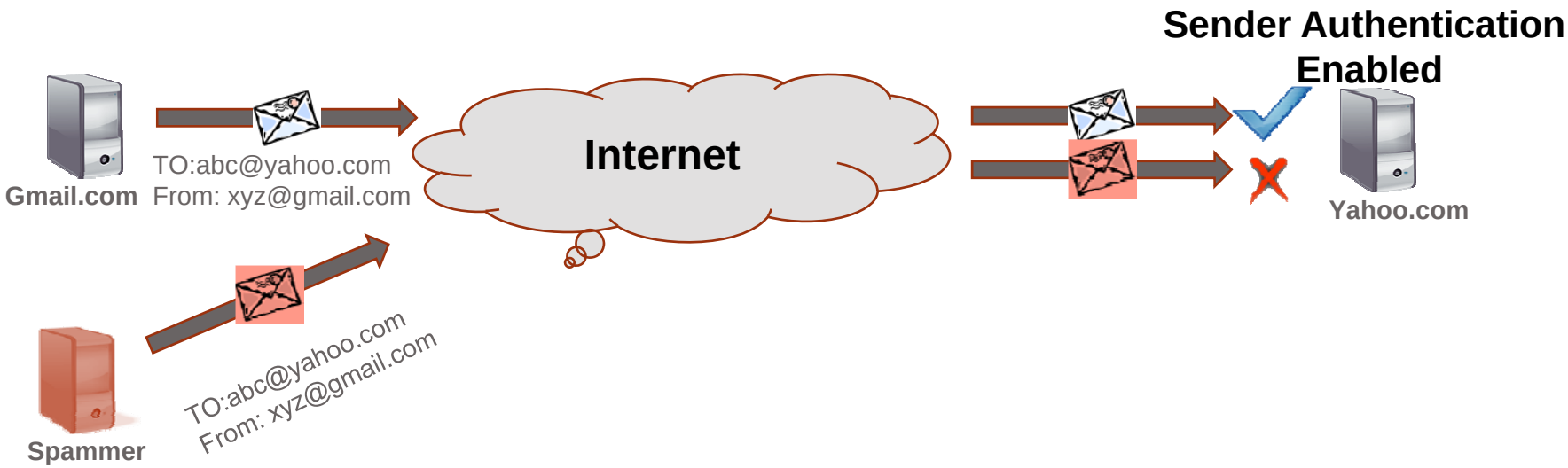
- Spam emails is still an open problem largely outnumbering legitimate ones.
- In 2010, 89% of the emails were spams (262 billion spam messages daily) [1].
- Projections show that spam will incur a cost of $338 billion by 2013 [2].

Why Spam works

- Use exploited MTAs (i.e., botnets) and with forged From: addresses.
- 88.2% of the total spam are sent by botnets using forged addresses (Symantec 2010).
- No or marginal infrastructural cost.
- No ownership, spammer remains anonymous, hence no action legal action possible.


from Rolex.com <msufianhameed@gmail.com>
hide details Mar 22 (2 days ago)
Reply to all
reply-to msufianhameed@gmail.com
to msufianhameed <msufianhameed@gmail.com>
date Tue, Mar 22, 2011 at 9:26 PM
subject msufianhameed Rolex.com For You -54%
Warning: This message may not be from whom it claims to be. Beware of following any links in it or of providing the sender with any personal information. Learn more
This message was likely forged and did not originate from your account. Learn More
Dear msufianhameed@gmail.com
WANTS TO SHARE SOMETHING WITH YOU
While exploring the Rolex.Com website, discovered this official Rolex address and thought you would be interested.
http://www.rolex.com;
and wanted to share it with you.
Thank you.
See you soon on Rolex.com
Rolex S.A. respects your right to privacy and is committed to maintaining your confidence and trust. Any information you provide us through our website will not be sold or rented.
Rolex S.A. does not collect any information from you without your knowledge and permission, and does not ask that you enter any personal information to access the website.
Information you provide Rolex S.A. is stored in a secure location and is accessible only by designated staff.
© Rolex SA, Geneva, Switzerland

Sender Authentication

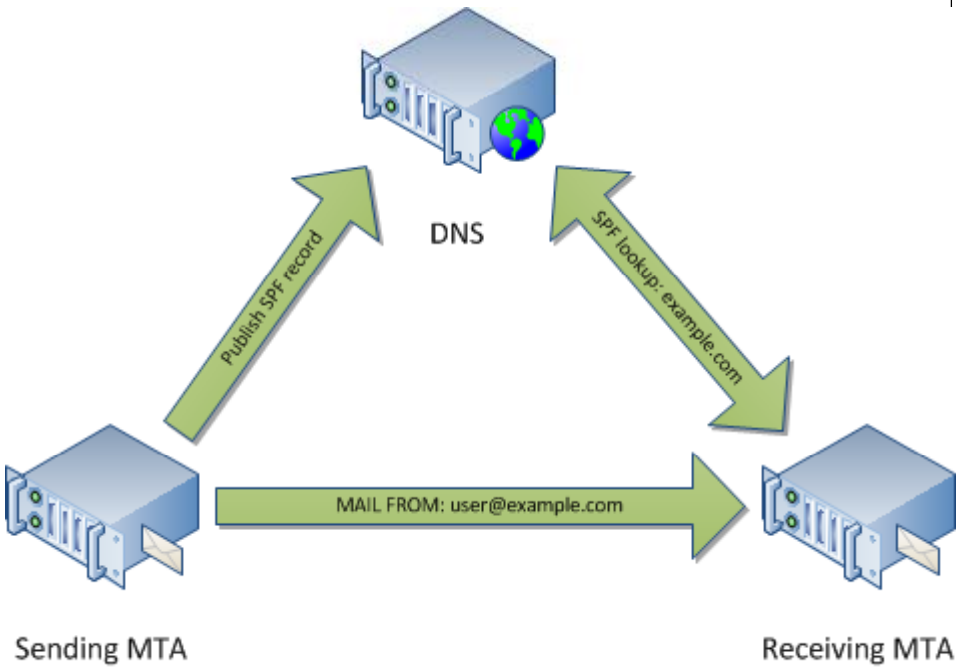


Industry Standards

- **AOL** -- Sender Policy Framework (SPF) --- started in mid 2005
- **Hotmail/MSN** – SenderID --- started in 2006
- **Yahoo** – Domain Key Identified Mail (DKIM) --- started in 2006

Sender Policy Framework (SPF)

- SPF is an IP-based sender authentication
- Operates on SMTP Envelope
- Allows domain admins to publish IP(s) for their valid servers as SPF record
- The receiving side can query the DNS to validate the senders' IP
- Most Adopted (60% of prominent domains)



Sender Policy Framework (SPF)

- Sample SPF
  - spf-a.hotmail.com text "v=spf1 ip4:209.240.192.0/19 ip4:65.52.0.0/14 ip4:131.107.0.0/16 ip4:157.54.0.0/15 ip4:157.56.0.0/14 ip4:157.60.0.0/16 ip4:167.220.0.0/16 ip4:204.79.135.0/24 ip4:204.79.188.0/24 ip4:204.79.252.0/24 ip4:207.46.0.0/16 ip4:199.2.137.0/24„
- SPF validation results
  - Pass – the host is authorized by the domain to send its e-mail.
  - Fail – the domain forbids the host to send its e-mail.
  - Neutral – the domain owner explicitly states that they cannot or do not want to assert whether the IP is authorized or not.
  - None – no SPF record found or no checkable sender's domain found.

Sender Policy Framework (SPF)

Problems:

- SPF is also easily adopted by Spammers.
- 20% of Spamming domains already adopted SPF.
- 5% of legitimate messages can potentially fail SPF test.
- Message forwarding is also a limitation.

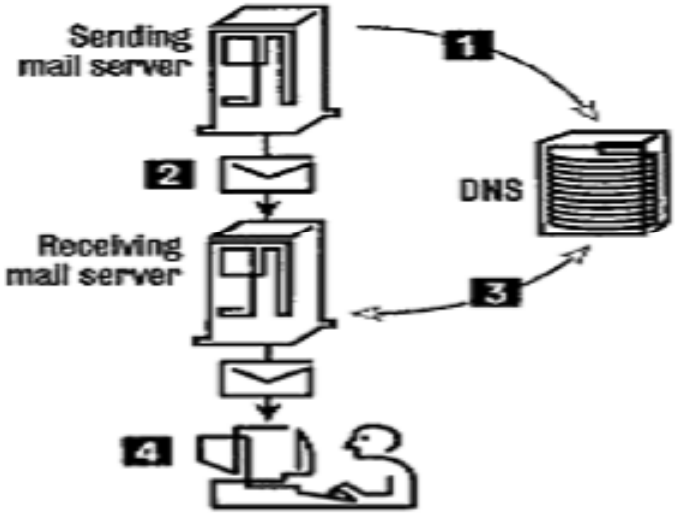
SenderID

- Heavily based on SPF, with only a few additions.
- Receiving MTA validates the Purported Responsible Address (PRA) i.e the From Address in the message header.
- Validation is not at the SMTP time
- Published record is almost identical to SPF, except
  - **v=spf1** is replaced by **spf2.0/mfrom, spf2.0/mfrom,pra or spf2.0/pra,mfrom** OR **spf2.0/pra**

Domain Key Identified Mail (DKIM)

- Header Based Approach (don't work at SMTP time)
- Defines a domain level digital signature authentication
- MTAs sign all the outbound mail (email header, body, etc)
- Use Public key cryptography
- DNS are used as key server technology (public keys are publish on the DNS). Receiver query the DNS for public key to verify the signature.

Domain Key Identified Mail (DKIM)



1. The sending domain publishes a public key in its DNS record.
2. The sending mail server digitally signs and sends the message.
3. The receiving mail server retrieves the public key from the sending domain's DNS record. It verifies the digital signature using the message content and the key.
4. The receiving mail server delivers the e-mail to the end user's mailbox.

* **Via DKIM.org**

Domain Key Identified Mail (DKIM)

```
DKIM-Signature: a=rsa-sha1; q=dns;
   d=example.com;
   i=user@eng.example.com;
   s=jun2005.eng; c=relaxed/simple;
   t=1117574938; x=1118006938;
   h=from:to:subject:date;
   b=dzdVyOfAKCdLXdJOc9G2q8LoXSlEniSb
     av+yuU4zGeeruD00lszZVoG4ZHRNiYzR
```

Problems:

- Spam Transmission: signature can only be validated after the entire message content is received
- Prone to content munging (common problem with lists)
- Spammer can also adopt it to sign their messages

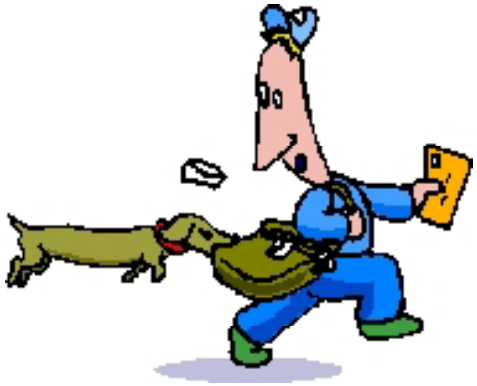
Example of SPF and DKIM

```
Delivered-To: msufianhameed@gmail.com
Received: by 10.14.128.1 with SMTP id e1csp211931eei;
        Mon, 19 Nov 2012 01:15:50 -0800 (PST)
Return-Path: <malik.jahan@gmail.com>
Received-SPF: pass (google.com: domain of malik.jahan@gmail.com designates 10.50.104.164 as permitted sender) client-ip=10.50.104.164
Authentication-Results: mr.google.com; spf=pass (google.com: domain of malik.jahan@gmail.com designates 10.50.104.164 as permitted sender)
dkim=pass header.i=malik.jahan@gmail.com
Received: from mr.google.com ([10.50.104.164])
        by 10.50.104.164 with SMTP id gf4mr614577igb.1.1353316549141 (num_hops = 1);
        Mon, 19 Nov 2012 01:15:49 -0800 (PST)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
        d=gmail.com; s=20120113;
        h=mime-version:sender:date:x-google-sender-auth:message-id:subject
         :from:to:content-type;
        bh=7JFOmWkykbYQNktE6QqMhmEu7imMjjTdOt9RW5y+nKk=;
        b=bfL9HOoa65tM/YrCovYDAn3Aw8BZRehNkM2Zs4brgd75jRXhb+2ehB0dozKnyyONzv
         uXHQAdTybxVkH17PNCjqsOUym2dtFrMoWxF1Sg2828iTOkSD1LXVGNwq9TufSMQkPzqa
         OJQiW1Lobj6h8ZkCDm5iu4+7ghLZGHxBCU3xFFmHzcFjSOW5BOe1CYZYGXJ+QcTgA93S
         7jwyf9/Gw8OT251IRZ24r866SFO24j4IOvuultZ6tboMCf3ikYZS+9a9yDOAd4dQcyTb
         1eHnJZrML8WbvrcuXpGjqXoREj7FuX7TOM2vvDrpuyXd3fJziOVak6GYU+4W4ecru3QG
```

Secure Email

- Email is one of the most widely used and regarded network services
- Currently message contents are not secure
  - may be inspected either in transit
  - or by suitably privileged users on destination system

Threats to E-mail



- Loss of confidentiality.
  - E-mails are sent in clear over open networks.
  - E-mails stored on potentially insecure clients and mail servers.
- Loss of integrity.
  - No integrity protection on e-mails; anybody be altered in transit or on mail server.

Threats to E-mail (2)

- Lack of data origin authentication.
  - Is this e-mail really from the person named in the From:field?
- Lack of non-repudiation.
  - Can I rely and act on the content? (integrity)
  - If so, can the sender later deny having sent it? Who is liable if I have acted?

Threats to E-mail (3)

- Lack of notification of receipt.
  - Has the intended recipient received my e-mail and acted on it?
  - A message locally marked as ‘sent’ may not have been delivered.

E-mail security

- Software for encrypting email messages has been widely available for more than 17 years, but the email-using public has failed to adopt secure messaging. This failure can be explained through a combination of:
  - technical,
  - community,
  - and usability factors

E-mail Security

- **Why Don’t People Use Email Security?**
  - I don’t because I don’t care.
  - I doubt any of my usual recipients would understand
  - the significance of the signature.
  - Never had the need to send these kinds of emails.
  - I don’t think it’s necessary to encrypt my email.
  - it’s just another step & something else I don’t have time

Why do I want secure email ?

- Protect sensitive data
- Prove authenticity to recipients
- Send attachments normally filtered
- Avoid the junk folder!

How does Secure Email works ?

- Secure email uses a set cryptographic tools to encapsulate a message into a specially formatted envelope.

Encryption

- Means of hiding a message through substitution or rearranging letters
- Requires a “key” to unlock the original message

Digital Signature

- A string of characters that uniquely identifies the signer of an electronic message.
- Recipients are able to
  - Verify message was from purported sender
  - Verify message was not modified in transit
- Sender cannot deny being originator of message

Pick your poison

- Most popular secure email standards
  - S/MIME
  - PGP (OpenPGP)
- How are these different?
  - Similar services
  - Different trust models

Hierarchical Trusts

- All users directly trust some central authority (CA) and the CA issue them a Digital Certificate
- Alice trusts Bob if Bob's "chain of trust" traces back to the central authority
- Example: driver's license
  - Issued by state authority to prove identity to others

Getting a Digital Certificate

- Must be issued by an authority
  - Organizational PKI
  - Third-party vendor
- Free personal certificates available
  - VeriSign
  - GeoTrust
  - Startcom
  - CACert
  - Comodo

http://www.sslshopper.com/email-certificates-smime-certificates.html

Web of Trust

- Incorporates user perception of trust
- Any user can be an authority to verify others
- Users can assign levels of trust
  - Not all authorities are equal
- “Alice and Bob think she is Carol, and that’s good enough for me.”

S/MIME (Secure/Multipurpose Internet Mail Extension)

- Originated from RSA Data Security Inc. in 1995.
- Further development by IETF S/MIME working group at: www.ietf.org/html.charters/smime-charter.html.
- Version 3.2 specified in RFC 5751.
- Allows flexible client-client security through encryption and signatures.
- Widely supported, e.g. in Microsoft Outlook, Thunderbird, Lotus Notes.

Understanding What S/MIME Does

- S/MIME provides two security services:
  - Digital signatures
  - Message encryption

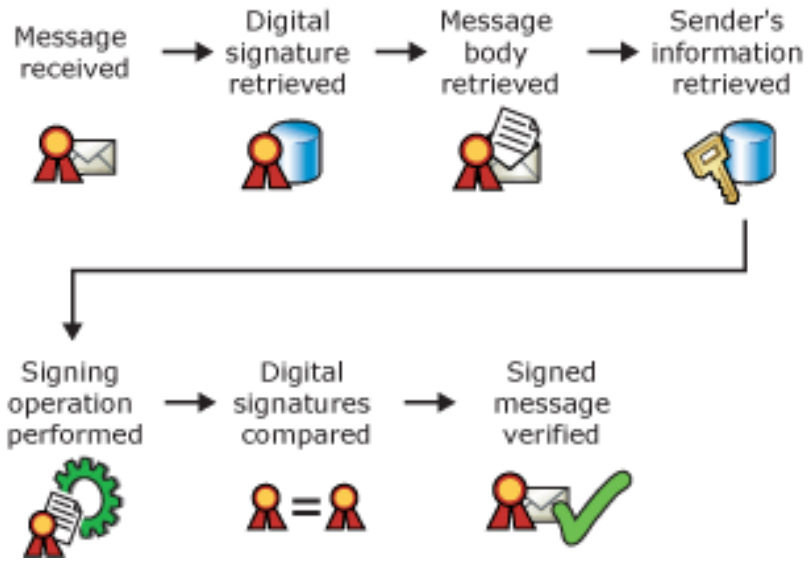
http://technet.microsoft.com/en-us/library/aa995740(v=exchg.65).aspx

S/MIME (Generating Digital Signature)

- Message is captured.
- Information uniquely identifying the sender is retrieved.
- Signing operation is performed on the message using the sender's unique information to produce a digital signature.
- Digital signature is appended to the message.
- Message is sent.

S/MIME (Verification of Digital Signature)

- Message is received.
- Digital signature is retrieved from the message.
- Message is retrieved.
- Information identifying the sender is retrieved.
- Signing operation is performed on the message.
- Digital signature included with the message is compared against the digital signature produced on receipt.
- If the digital signatures match, the message is valid.

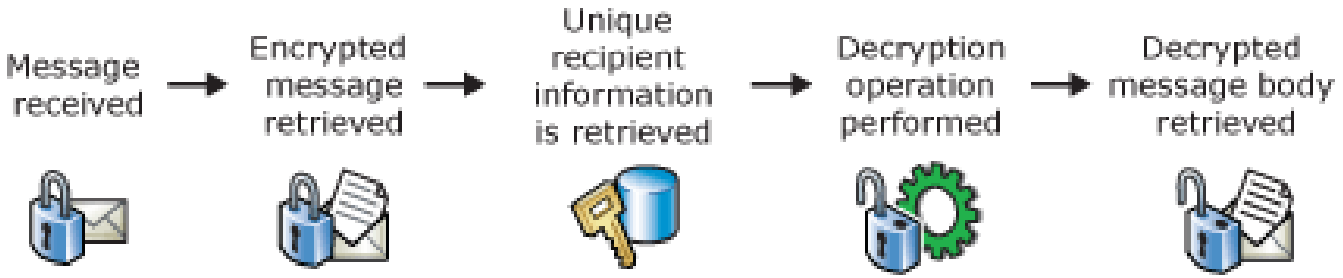


S/MIME (Performing an Encryption Operation)

- Message is captured.
- Information uniquely identifying the recipient is retrieved.
- Encryption operation is performed on the message using the recipient's information to produce an encrypted message.
- Encrypted message replaces the text in the message.
- Message is sent.

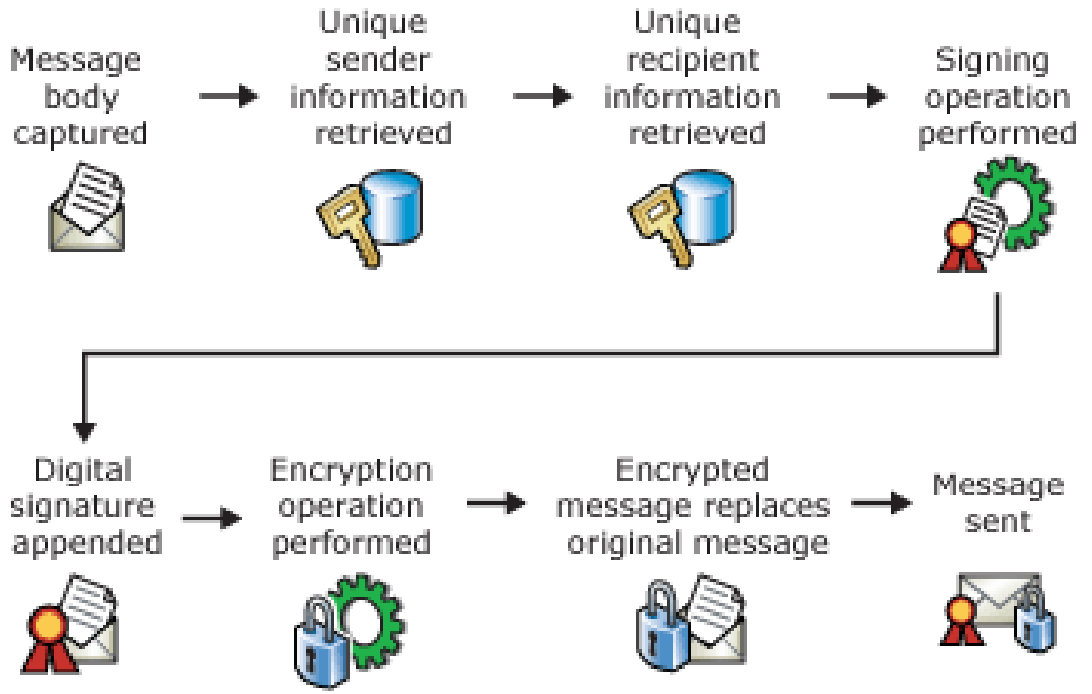
S/MIME (performing a Decryption Operation)

- Message is received.
- Encrypted message is retrieved.
- Information uniquely identifying the recipient is retrieved.
- Decryption operation is performed on the encrypted message using the recipient's unique information to produce an unencrypted message.
- Unencrypted message is returned to the recipient.



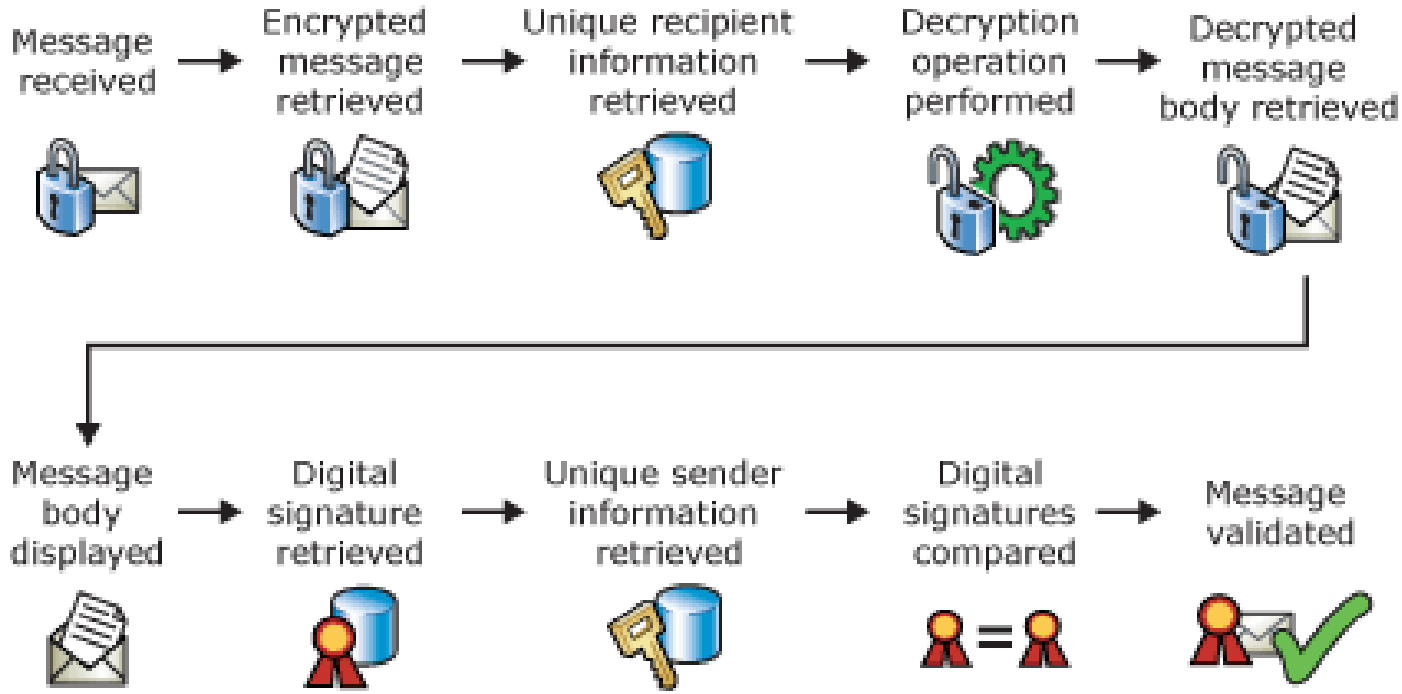
How Signatures and Encryption Work Together

- The following figure shows the sequence of signing and encrypting an e-mail message



How Signatures and Encryption Work Together

- The following figure shows the sequence of decrypting and verifying the digital signature.



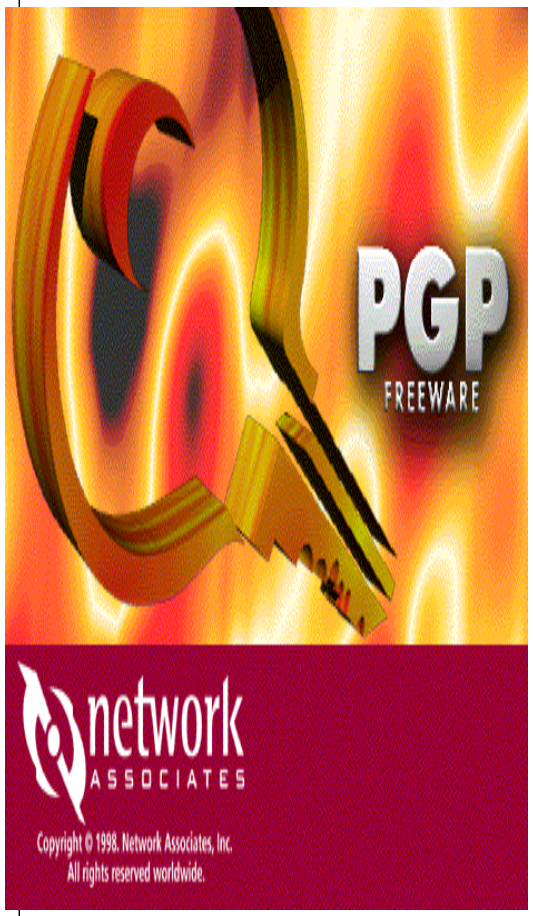
Key Wrapping and Content Encryption in S/MIME

- RSA is used as Key wrapping algorithm
- Content encryption is based on session keys.
  - AES-128 CBC added as MUST, AES-192 and AES-256 CBC are optional

PGP (Pretty Good Privacy)

- Open source, freely available software package for secure e-mail
- De facto standard for secure email
- Developed by Phil Zimmermann
- Selected best available crypto algos to use
- Runs on a variety of platforms like Unix, PC, Macintosh and other systems
- Originally free (now also have commercial versions available)

PGP (Pretty Good Privacy)



- PGP is an open-source freely available software package for e-mail security. It provides authentication; confidentiality; compression; e-mail compatibility; and segmentation and reassembly.

PGP (Pretty Good Privacy)

- **PGP Algorithms**
  - **Symmetric encryption:**
    - DES, 3DES, AES and others.
  - **Public key encryption of session keys:**
    - RSA or ElGamal.
  - **Hashing:**
    - SHA-1, MD-5 and others.
  - **Signature:**
    - RSA, DSS, ECDSA and others.

Key Management and Web of Trust

PGP use:

- **public keys** for encrypting session keys / verifying signatures.
- **private keys** for decrypting session keys / creating signatures.

- PGP adopts a trust model called the *web of trust.*
- No centralised authority
- Individuals sign one another's public keys, these "certificates" are stored along with keys in key rings.

Trust Level for Public Key

- PGP computes a *trust level* for each public key in key ring.
- Users interpret trust level for themselves.
- Trust levels for public keys dependent on:
  - Number of signatures on the key;
  - Trust level assigned to each of those signatures.
- Trust levels recomputed from time to time.

PGP Security

- There are many known attacks against PGP.
- Attacks against crypto-algorithms are not the main threat.
- An attacker may socially engineer himself into a web of trust, or some trustable person may change. Then he could falsify public keys. This breaks most of the security.
- PGP binaries can be corrupted when they are obtained.
- The PGP binaries can be modified in the computer.
- The passphrase can be obtained by a Trojan. Weak passphrases can be cracked.
- On multiuser system, access to the secret key can be obtained.

PGP Key Rings

- PGP Key Rings
  - PGP supports multiple public/private keys pairs per sender/recipient.
  - Keys stored locally in a *PGP Key Ring* – essentially a database of keys.
  - Private keys stored in encrypted form; decryption key determined by user-entered pass-phrase.

PGP Session Keys

- Need a session key for each message
  - Varying sizes: 56-bit DES, 128-bit CAST or IDEA, 168-bit Triple-DES, 128, 192, 256 bits AES
- Uses random inputs taken from

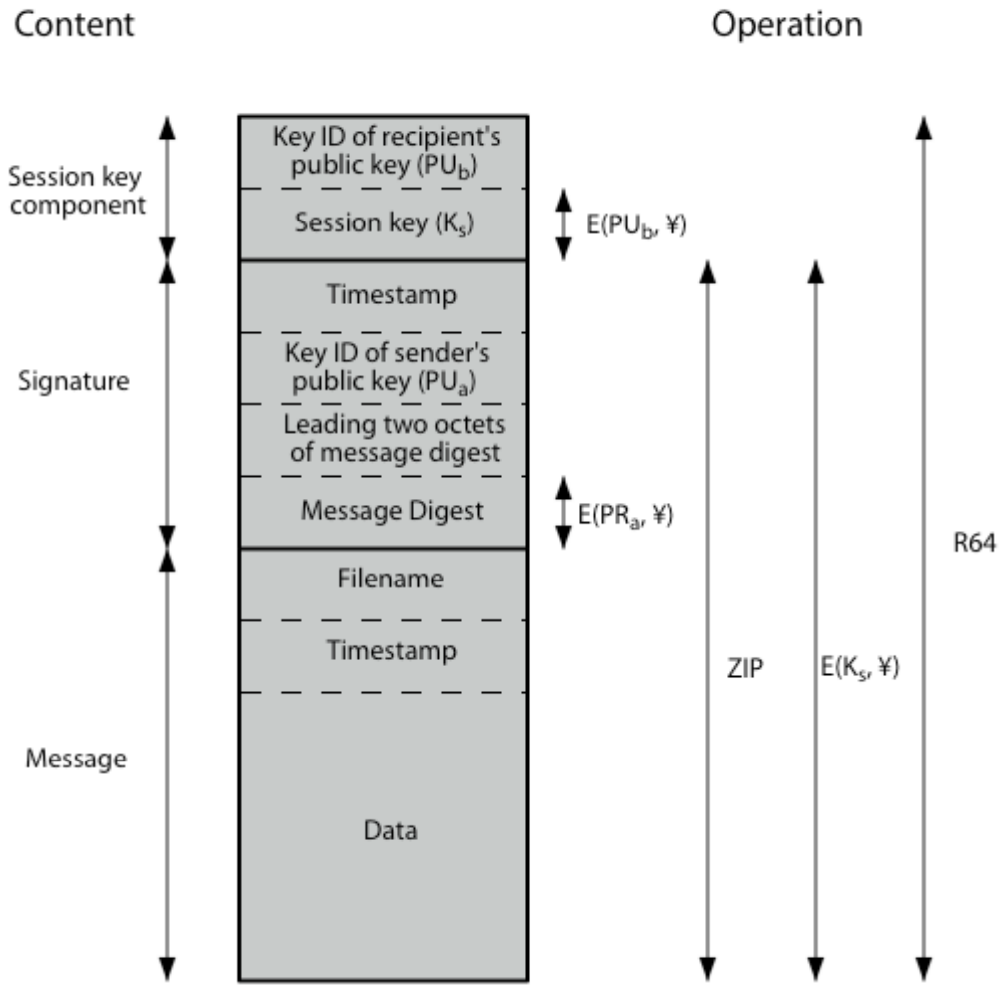
  -actual keys hit

  -keystroke timing of a user

PGP Public & Private Keys

- Since many public/private keys may be in use, need to identify which is actually used to encrypt session key in a message
  - Could send full public-key with every message
  - but this is inefficient
- Rather use a key identifier based on key
  - is least significant 64-bits of the key
  - will very likely be unique
- Also use key ID in signatures

PGP Message Format

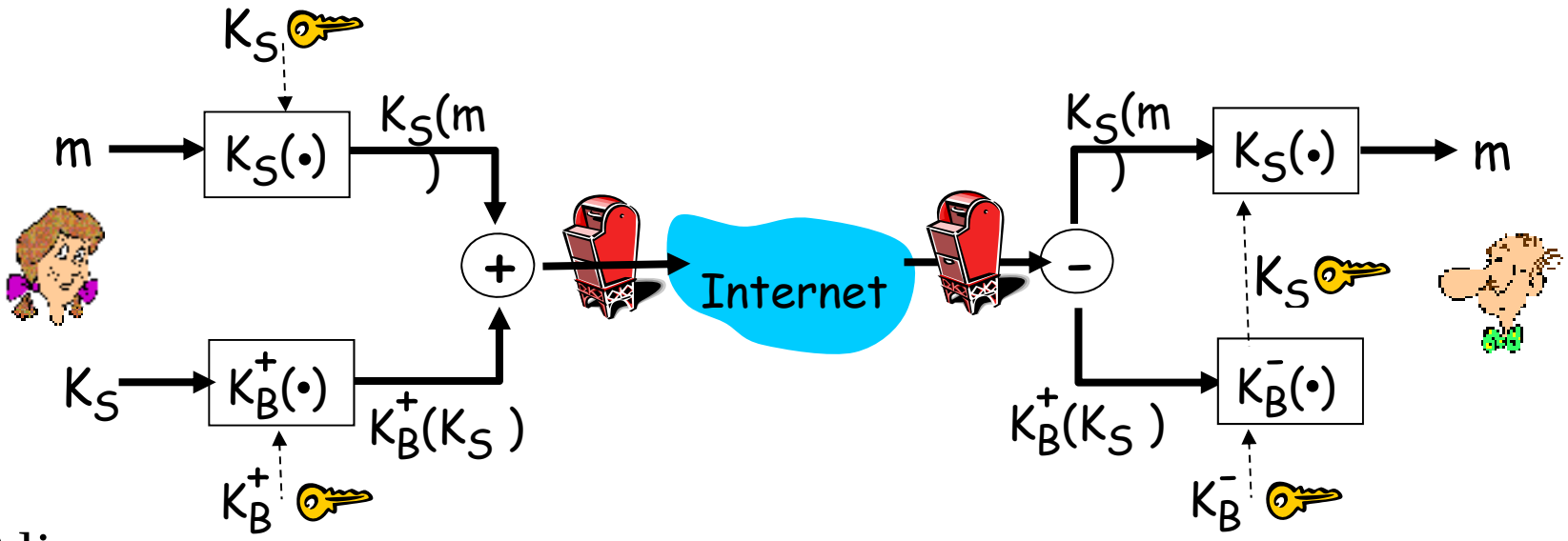


PGP services

- **Messages**
  - authentication
  - confidentiality
  - compression
  - e-mail compatibility
  - segmentation and reassembly
- **Key Management**
  - generation, distribution, and revocation of public/private keys
  - generation and transport of session keys and IVs

PGP Operation – Confidentiality

- Alice wants to send confidential e-mail, m, to Bob.

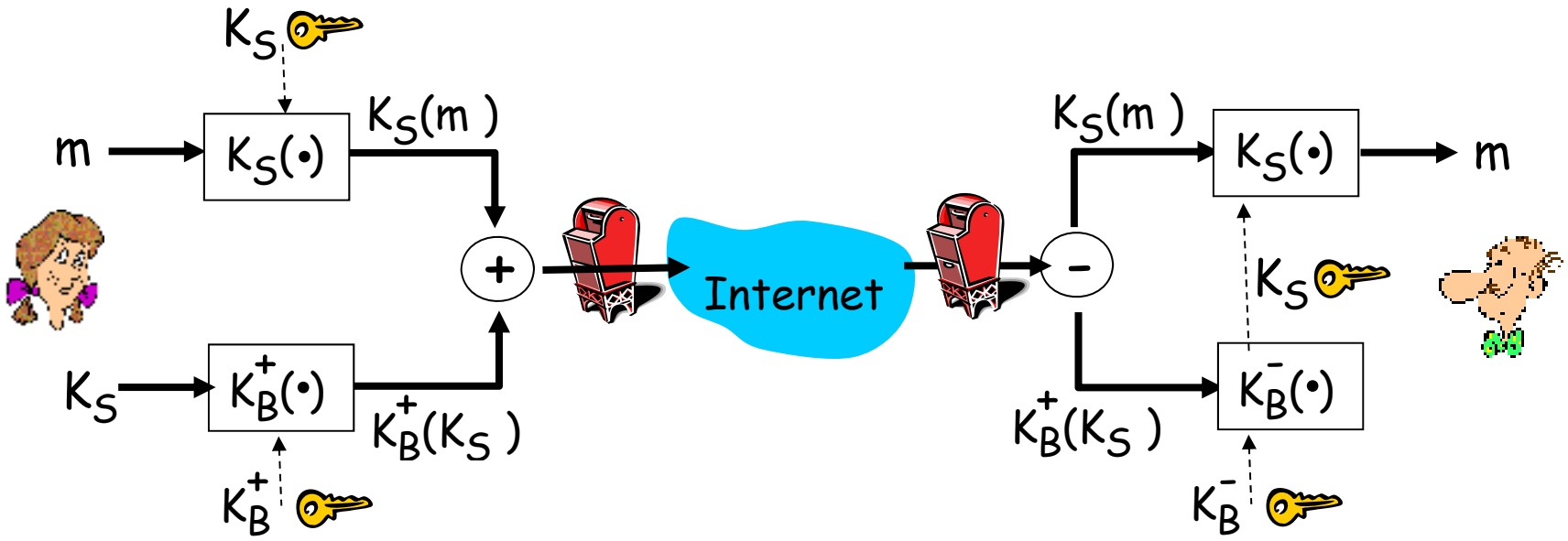


Alice:

- generates random *symmetric* private key, K_S.
- encrypts message with K_S (for efficiency)
- also encrypts K_S with Bob's public key.
- sends both $K_S(m)$ and $K_B(K_S)$ to Bob.

PGP Operation – Confidentiality

- Alice wants to send confidential e-mail, m, to Bob.

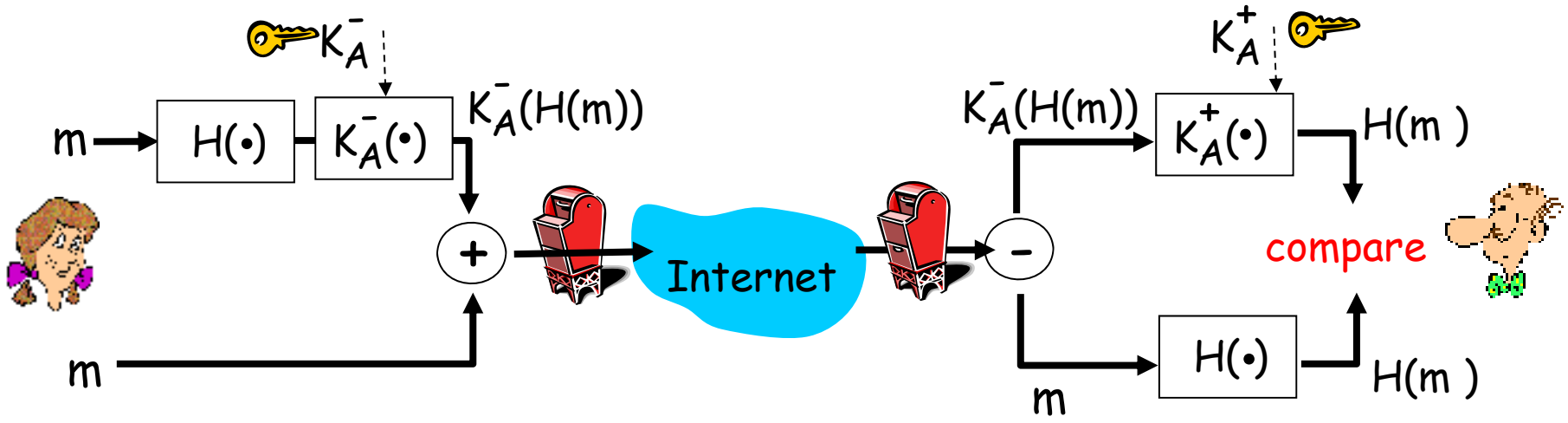


Bob:

- uses his private key to decrypt and recover K_S
- uses K_S to decrypt $K_S(m)$ to recover m

PGP Operation – Authentication

• Alice wants to provide sender authentication message integrity.



- Alice digitally signs message.
- sends both message (in the clear) and digital signature.

PGP Operation – Confidentiality & Authentication

- can use both services on the same message
  - create signature & attach it to the message
  - encrypt both message & signature
  - attach RSA/ElGamal encrypted session key

  This sequence is preferred because
  --one can store the plaintext message/file and its signature
  --no need to decrypt the message/file again and again

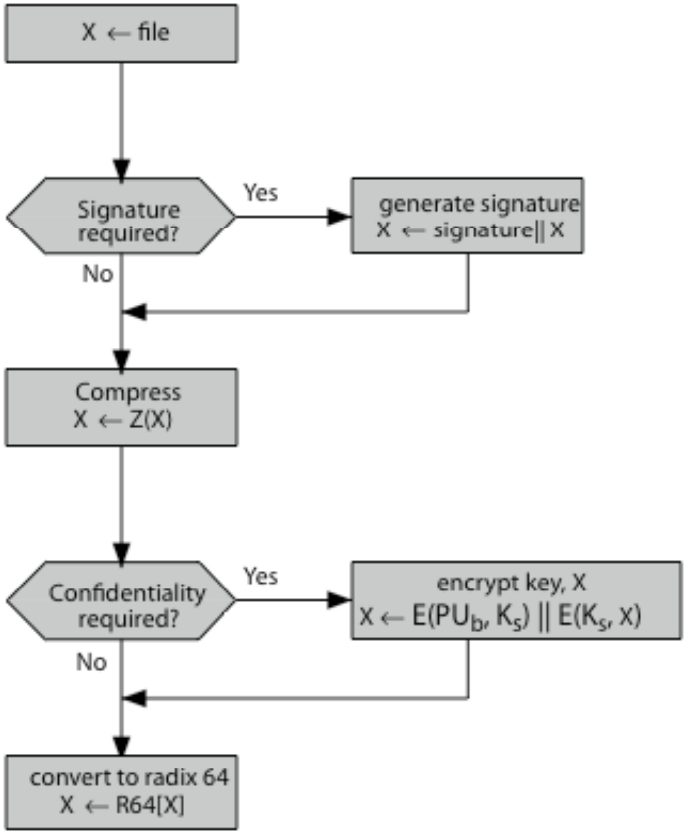
PGP Operation – Compression

- PGP compresses messages to save space for e-mail transmission and storage
- by default PGP compresses message after signing but before encrypting
  - so can store uncompressed message & signature for later verification
  - Encryption after compression strengthens security (because compression has less redundancy)
- uses ZIP compression algorithm

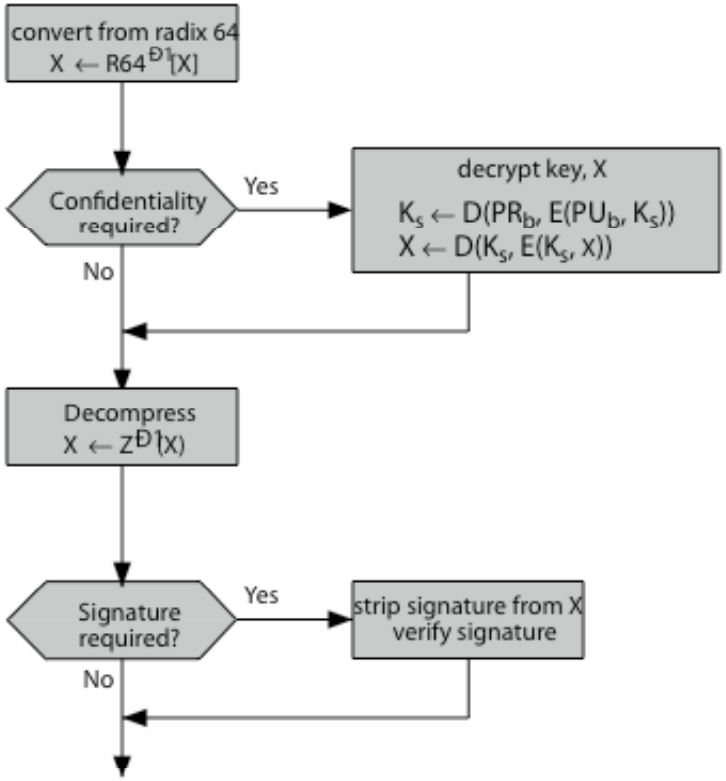
PGP Operation – Email Compatibility

- PGP will have binary data (8-bit octets) to send (encrypted message, etc)
- However email was designed only for text
- Hence PGP must encode raw binary data into printable ASCII characters
- Uses radix-64 algorithm
  - maps 3 bytes to 4 printable chars
  - also appends a CRC
- PGP also segments messages if too big (maximum length 50,000 octets)

PGP Operation – Summary

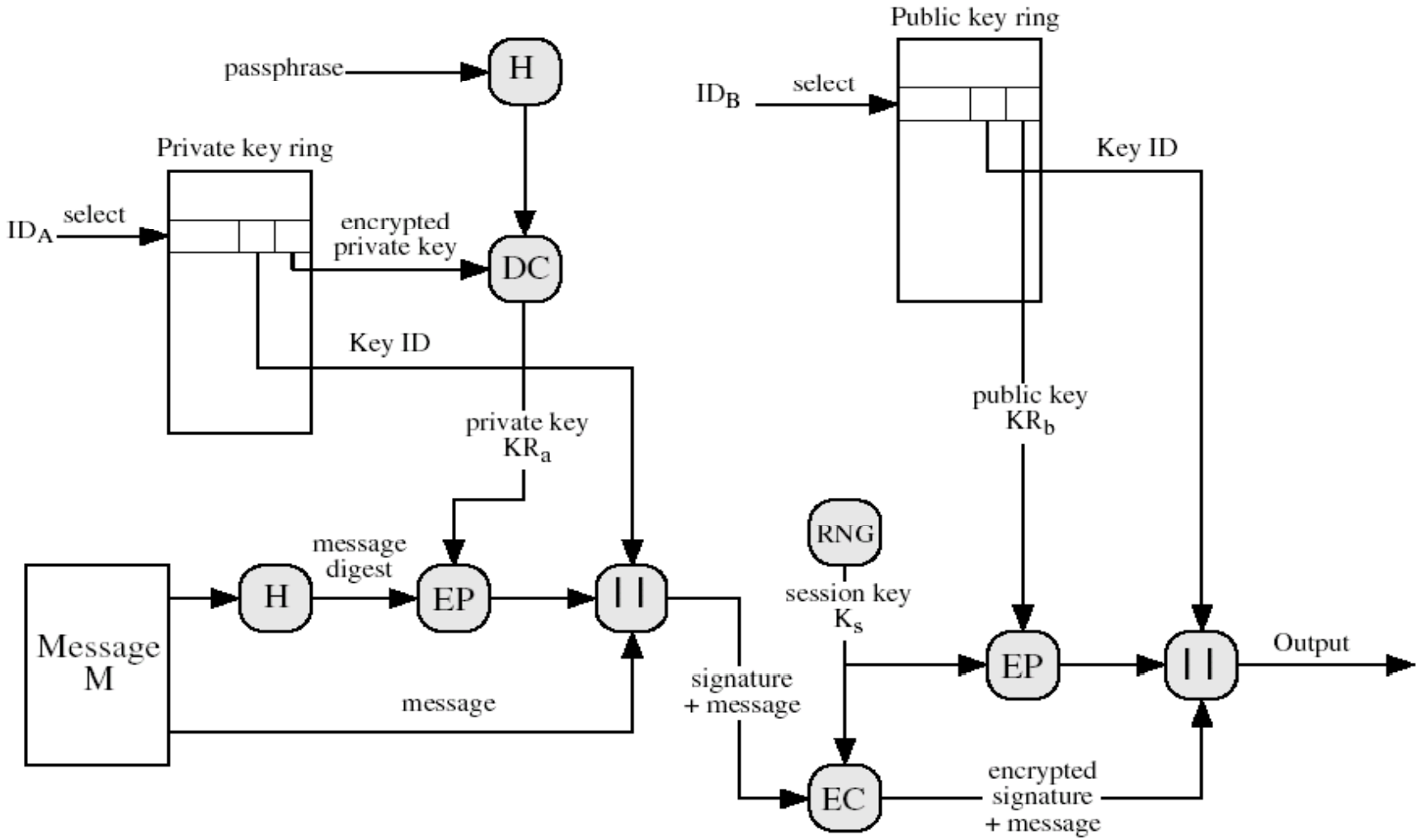


(a) Generic Transmission Diagram (from A)



(b) Generic Reception Diagram (to B)

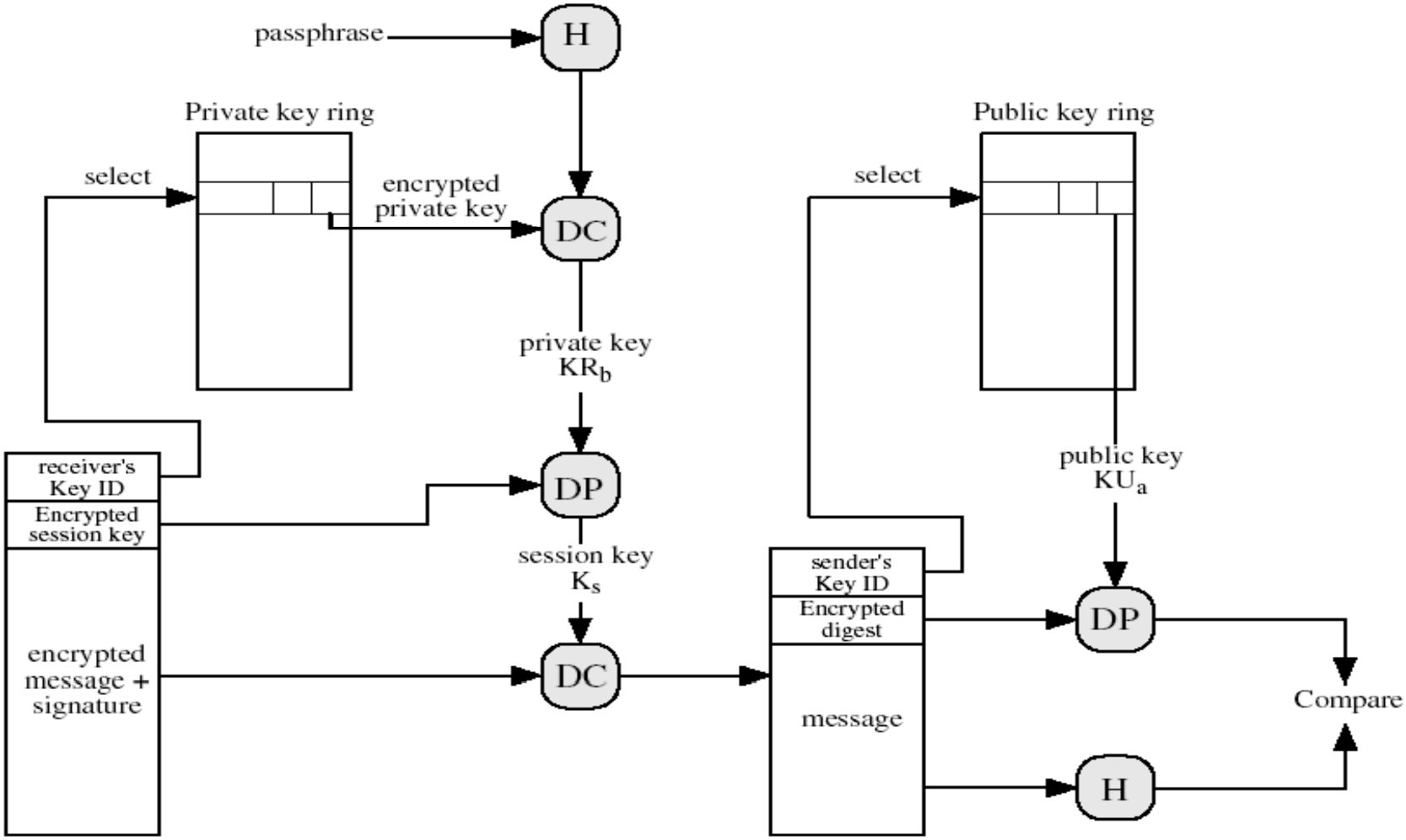
PGP Message Generation



PGP Message Generation

- The sending PGP entity performs the following steps:
  - Signs the message:
    - PGP gets sender's private key from key ring using its user id as an index.
    - PGP prompts user for passphrase to decrypt private key.
    - PGP constructs the signature component of the message.
  - Encrypts the message:
    - PGP generates a session key and encrypts the message.
    - PGP retrieves the receiver public key from the key ring using its user id as an index.
    - PGP constructs session component of message

PGP Message Reception



PGP Message Reception

- The receiving PGP entity performs the following steps:
  - Decrypting the message:
    - PGP get private key from private-key ring using Key ID field in session key component of message as an index.
    - PGP prompts user for passphrase to decrypt private key.
    - PGP recovers the session key and decrypts the message.
  - Authenticating the message:
    - PGP retrieves the sender's public key from the public-key ring using the Key ID field in the signature key component as index.
    - PGP recovers the transmitted message digest.
    - PGP computes the message for the received message and compares it to the transmitted version for authentication.

Resources

- http://www.pgpi.org/doc/faq/
- www.gnupg.org
- William Stallings," **Cryptography and Network Security Principles and Practices",** Fourth Edition " Prentice Hall , 2005
- GITA" **Encryption Technologies",** Standard P800-S850 V2.0, April 5, 2004
- Sieuwert van Otterloo" **A security analysis of Pretty Good Privacy",** September 7, 2001
- Amr el-kadi" what is computer security"2005

Acknowledgements

Material in this lecture are taken from the slides prepared by:

- Hussain Awad and Dr. Lo'ai Tawalbeh
- Lawrie Brown (University of Kentucky)
- M. Singhal (University of Kentucky)
- Addam Schroll (Purdue University)
- http://royal.pingdom.com/2011/01/19/email-spam-statistics/
- http://www.redcondor.com/company/