
# Protection Against Spam Using Pre-Challenges

Rodrigo Roman, Jianying Zhou, ***Javier Lopez***

June 1, 2005

# Table Of Contents


- Spam
- Anti-Spam schemes
- Pre-Challenge Scheme
- Discussions
- Conclusions

# Spam

# Spam - What is Spam?

E-Mail Spam = Junk Mail = Unsolicited Commercial E-mail (UCE)

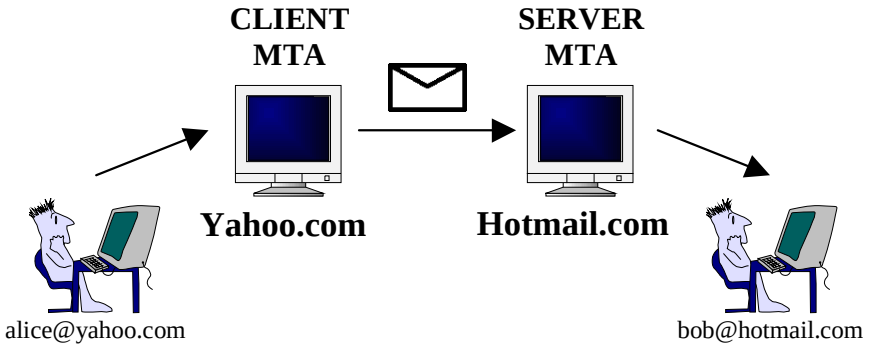
- Nigerian “419” Scam, Pyramid Schemes, Sexual “improvements”,...

Why Spam happens?

- E-Mail infrastructure is vulnerable!
- Based on a protocol made in 1982 (SMTP), with minor revisions

# Spam - SMTP Flaws

SMTP:




MTA = Mail server

E-mail = Σ ( source address, destination address, body, headers)

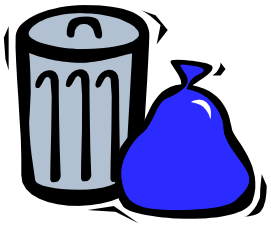
- …there is no authentication! Everything can be forged.
- Spammer can control an user machine and/or a Client MTA

# Spam - Economy

Is it profitable?

- It’s easy to send spam to millions of recipients
  - Just need to know how to reach them! (addresses)
  - Web agents that analyse Web Sites, side attacks over SMTP…
- Equipment? It’s almost free!
  - Software: a Mass Mailing program
  - Hardware:
    - Spammer using own servers (one or more machine(s) + Internet line). Need anonymizer (proxy)
    - Spammer controlling another MTA (e.g. relay server)

# Anti-Spam Schemes



# Actual Schemes

Purpose

- Avoiding Spam while maintaining actual SMTP protocol and E-mail infrastructure

Tools that can be used against spam

- “Homebrew” solutions (John NOSPAM Doe AT yahoo DOT com)
- “Received” headers
- Destination address
- Email Content
- Others: Micropayments, Challenge-Response

# Actual Schemes

"Received" headers

- They indicate the path of the email through Internet
- Malicious Client MTAs can be detected ("blacklisting")
- Problem: Individual spammers, banning "innocent" users/domains

Destination address

- A policy or password can be encoded in the address of the receiver
- Contains temporal policies (can be used until X), valid senders,...
- Problem: Scheme oriented for computers, not for humans

# Actual Schemes

Email contents - filtering

- Content can be analyzed using AI or statistical techniques
- Try to distinguish whether an email is spam or not
- Problem: Can lead to false positives and false negatives

Micropayment

- Client MTA must compute a function before sending any message
- Prevents evil MTAs from sending millions of emails
- Problems: Client devices with weak capability, reduce MTAs performance

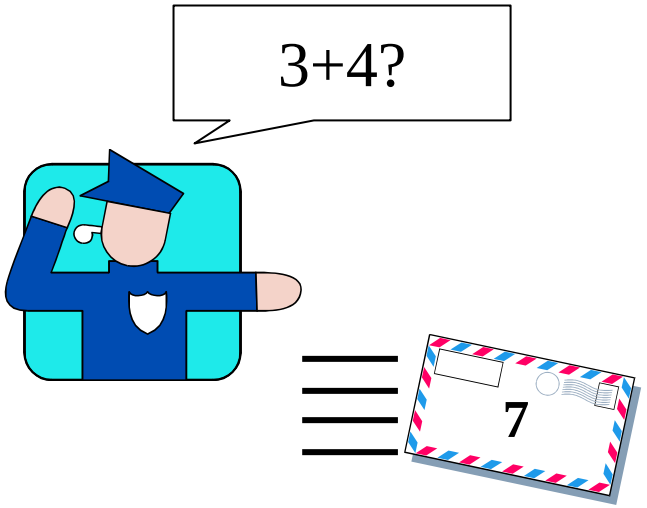
# Actual Schemes

Challenge - Response

- When receiving email from unknown user: send him/her a challenge
- Challenge can be simple (reply) or complicated (hard-AI problem)
- When correct response is received, emails of that user are allowed to enter
- Hybrid Solutions: Microsoft “Penny Black” Project (micropayments)
- Problem: Mailing lists, delay of service, possible DDoS

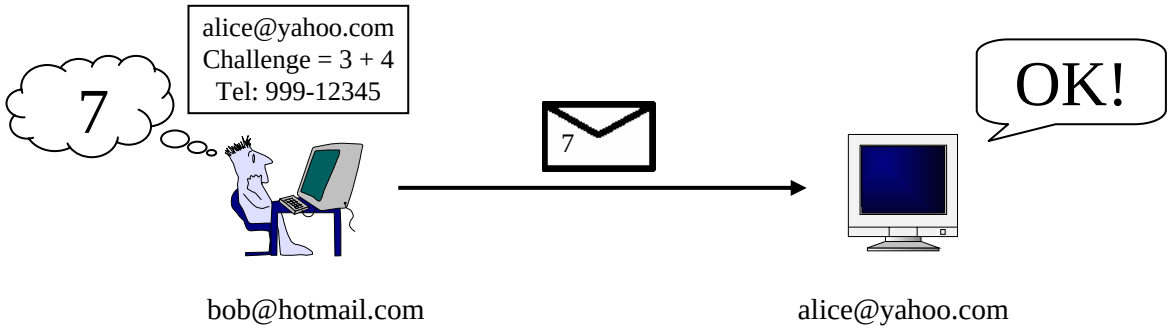
Evolution $\rightarrow$ Pre-Challenge Scheme

# Pre-Challenge Scheme



# Pre-Challenge - Core

- MAIN IDEA: Sender must retrieve receiver's email from somewhere. So... Also challenge (simultaneously)
- Sender solves challenge, send email. If correct, will be accepted
- Why? Check whether there is a machine behind sender's computer (Mass mailing programs)



# Pre-Challenge - Challenge

Who defines the challenge?

- Every human user defines his/her own challenge ("What is the name of my dog" in a website about my dog)

Where is the challenge stored?

- Next to its user's email address
- In a website, in a business card,…

alice@yahoo.com
Challenge = 3 + 4
Tel: 999-12345

How is the challenge?

- Range from a single word or mathematical operation to a hard-AI problem

# Pre-Challenge - How it Works

Data Structures

- Contains e-mail addresses of users

*White-List (safe-list)*

- Users already accepted - no challenge tests

*Reply-List*

- Users which the local user sent e-mail, and did not reply yet

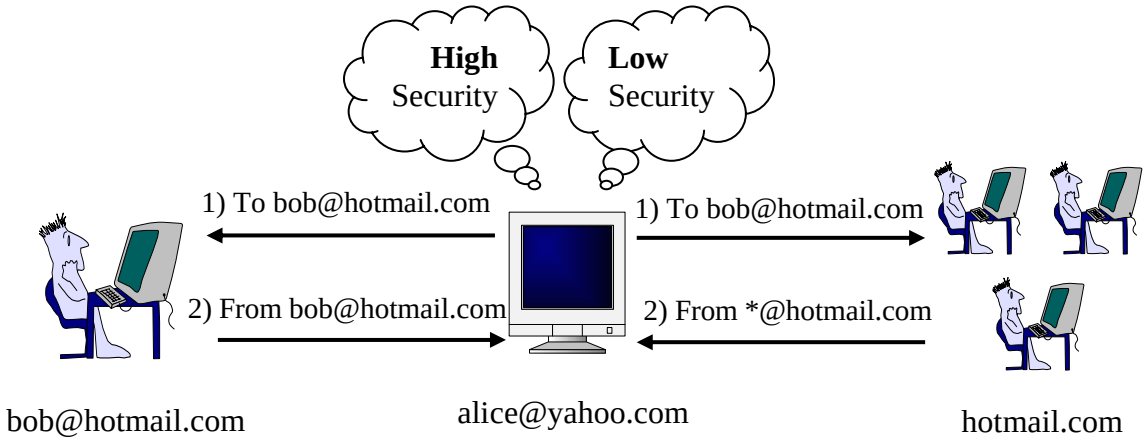
*Warning-List*

- Users that had been warned about how is the actual challenge

# Pre-Challenge - How it Works

Security Levels

- High Security: Reply-List is queried searching for a <user,domain> match
- Low Security: Reply-List is queried searching for a <*,domain> match

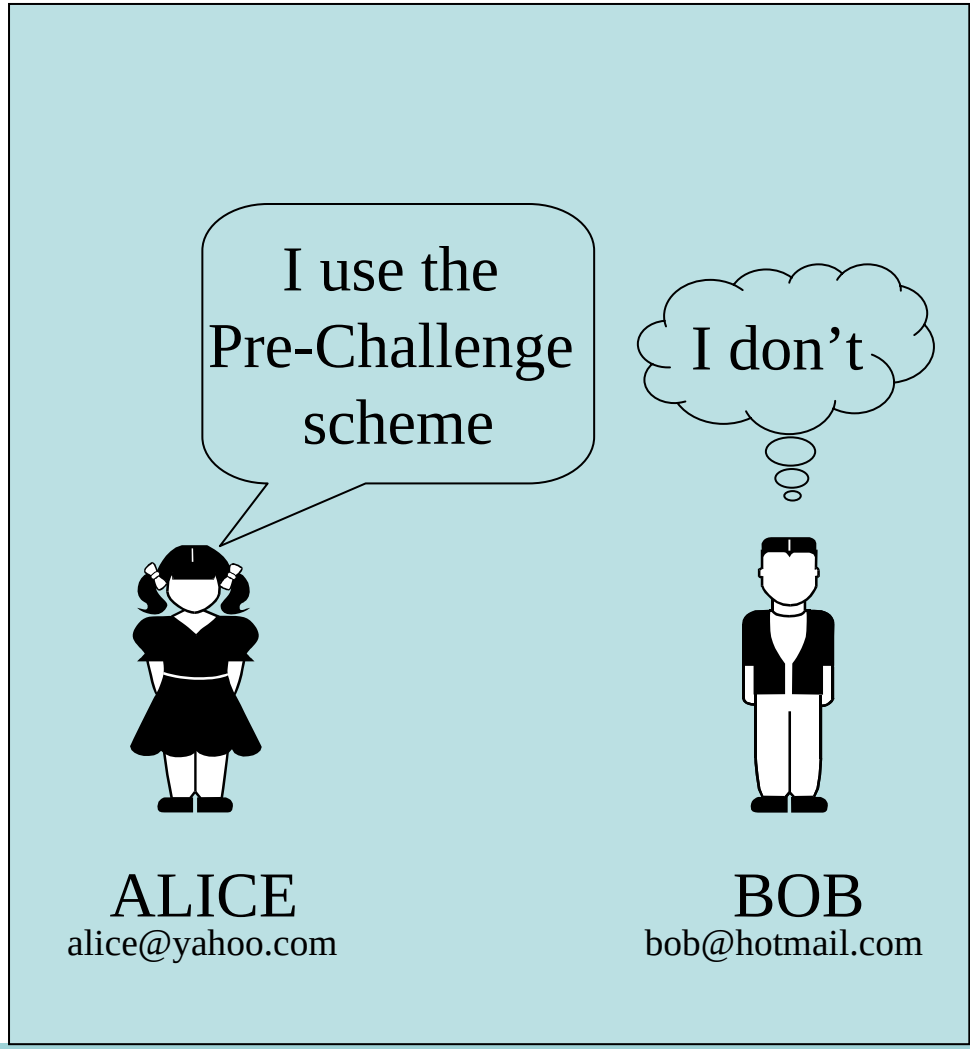


# Pre-Challenge - How it Works

Architecture - Meet the Actors

- **A**lice: User that uses the pre-challenge scheme
- **B**ob: User that does not use the Pre-Challenge scheme

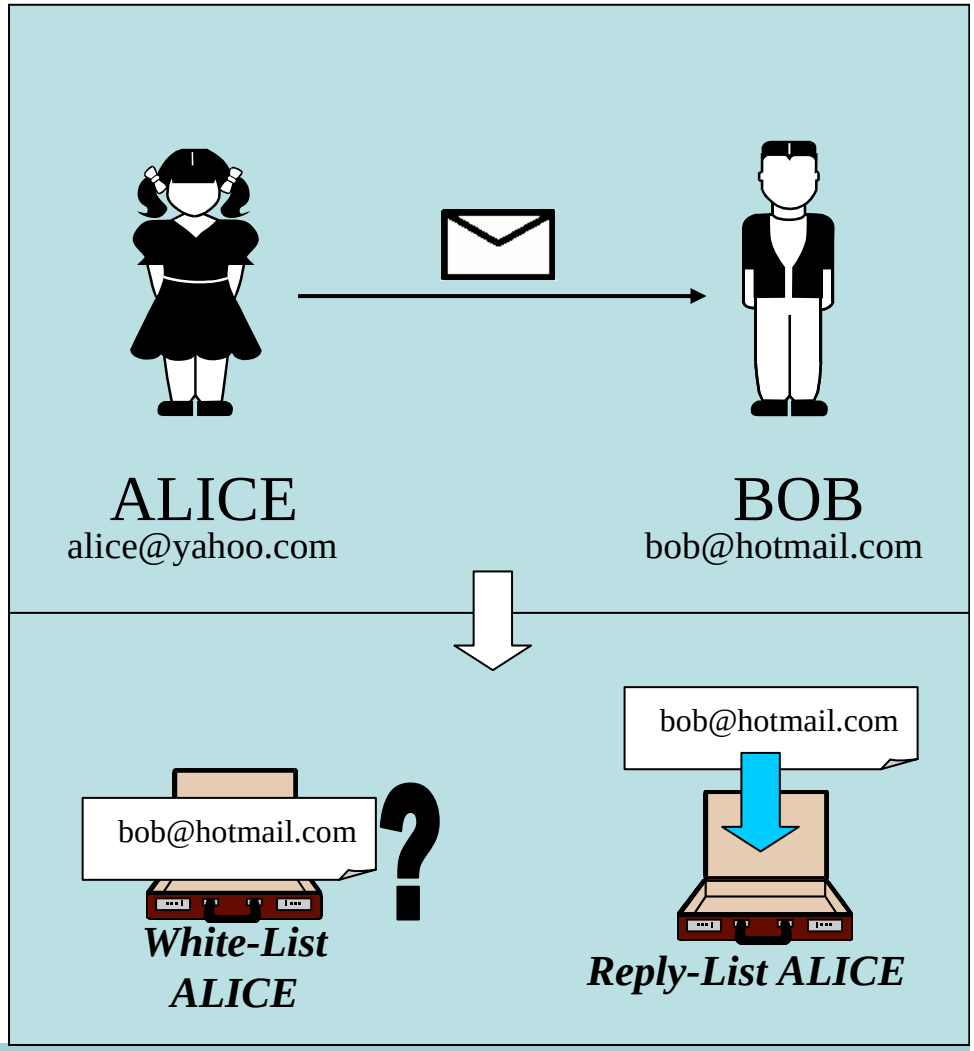
This is done for simplifying the explanation, both users can use the scheme simultaneously without problems



# Pre-Challenge - How it Works

(1)

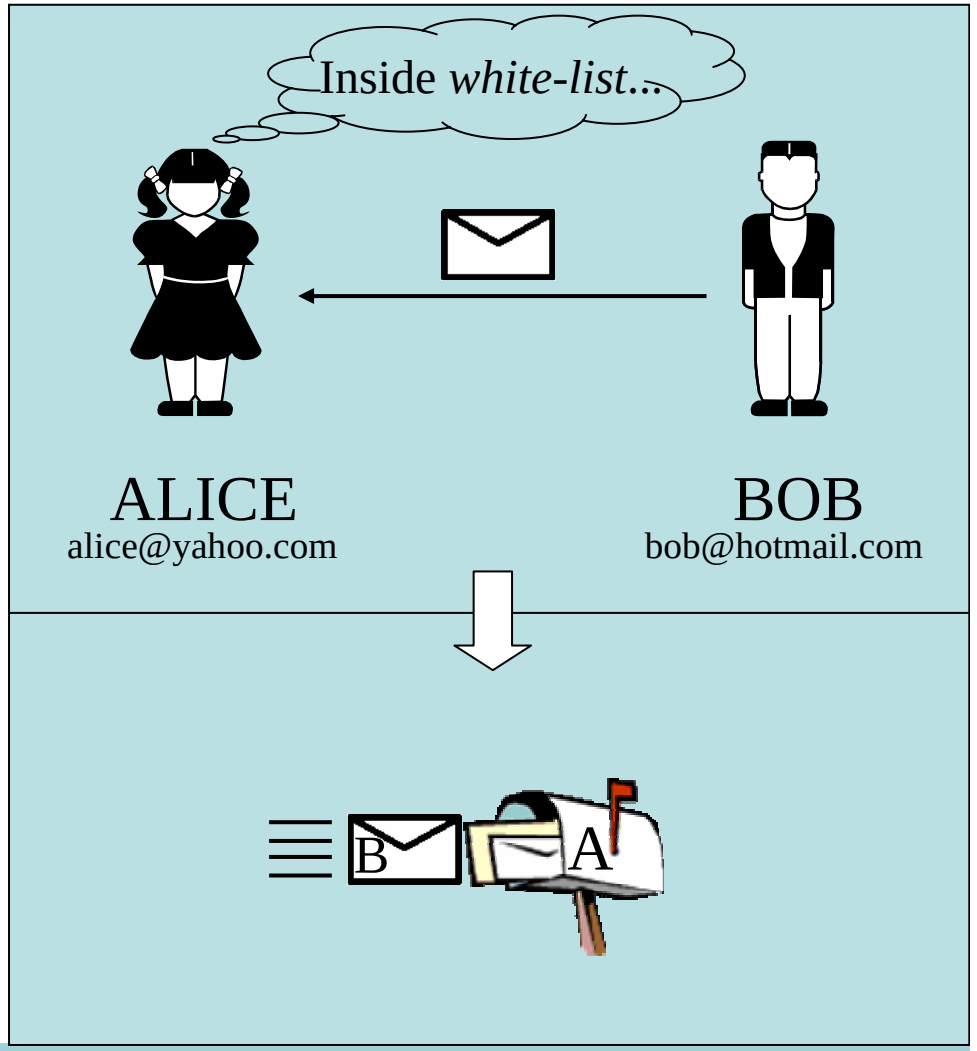
- When A sends an e-mail to B, B's e-mail address is added to *reply-list* if not in *white-list*



# Pre-Challenge - How it Works

(1’)

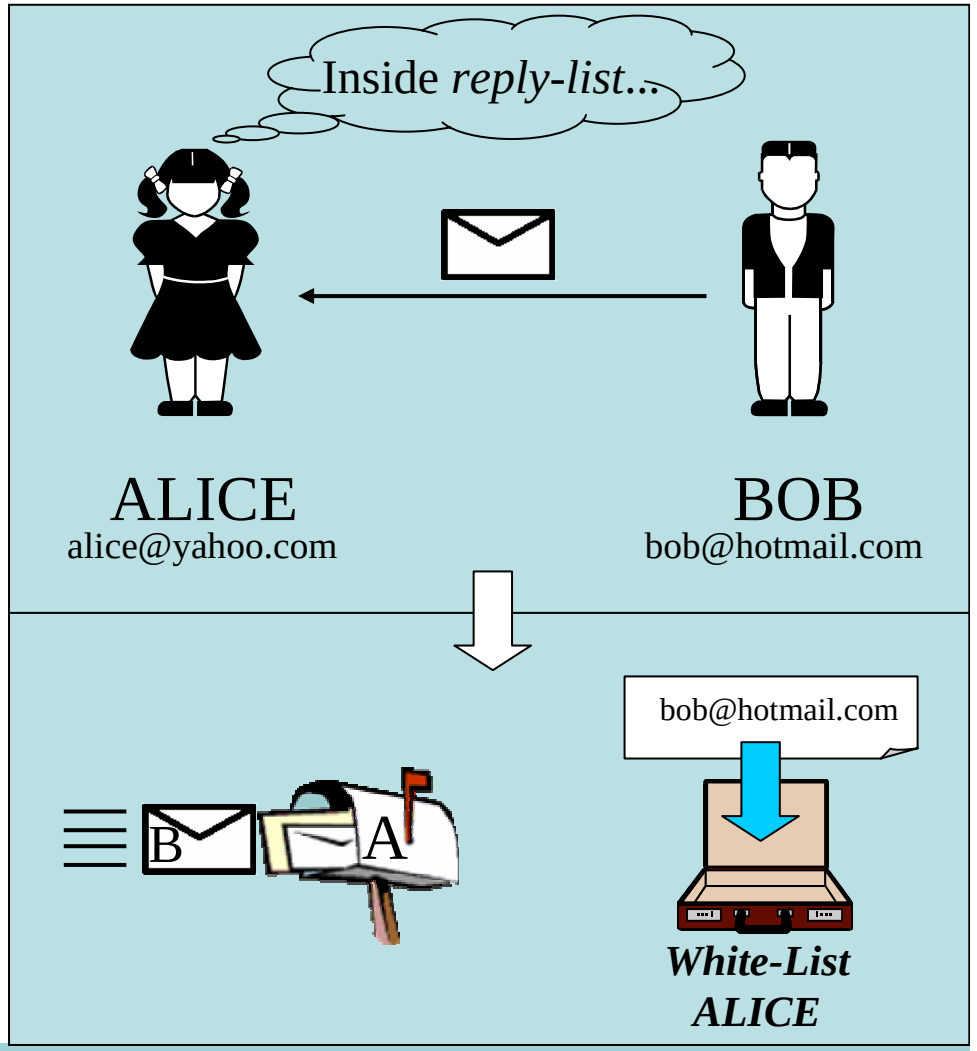
- When B sends an e-mail to A, A checks if B’s address is in *white-list*. If yes, mail reaches A’s mailbox



# Pre-Challenge - How it Works

(2’)

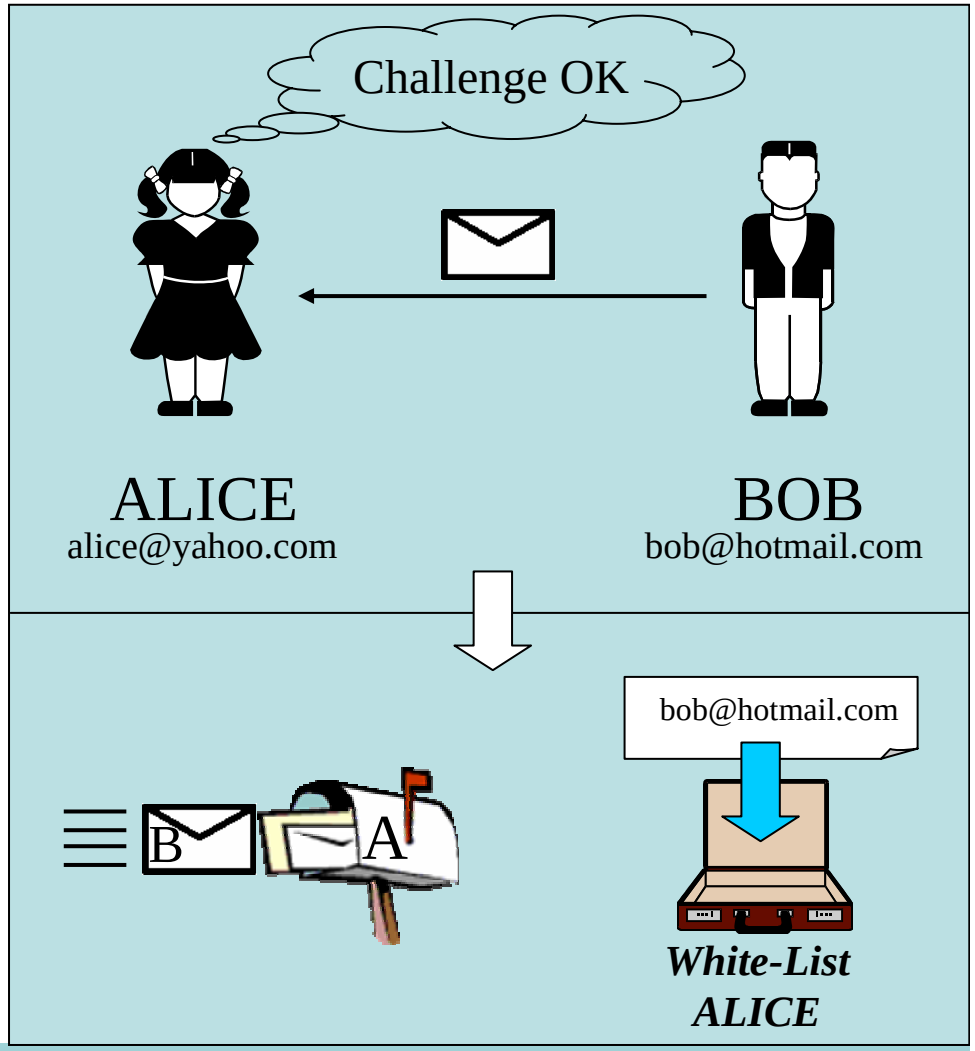
- If B is listed in the *reply-list*, the mail reaches A's mailbox and B is added to the *white-list*.
- In case of using a high security level, B is erased from the *reply-list* because A received the reply expected from B



# Pre-Challenge - How it Works

(3’)

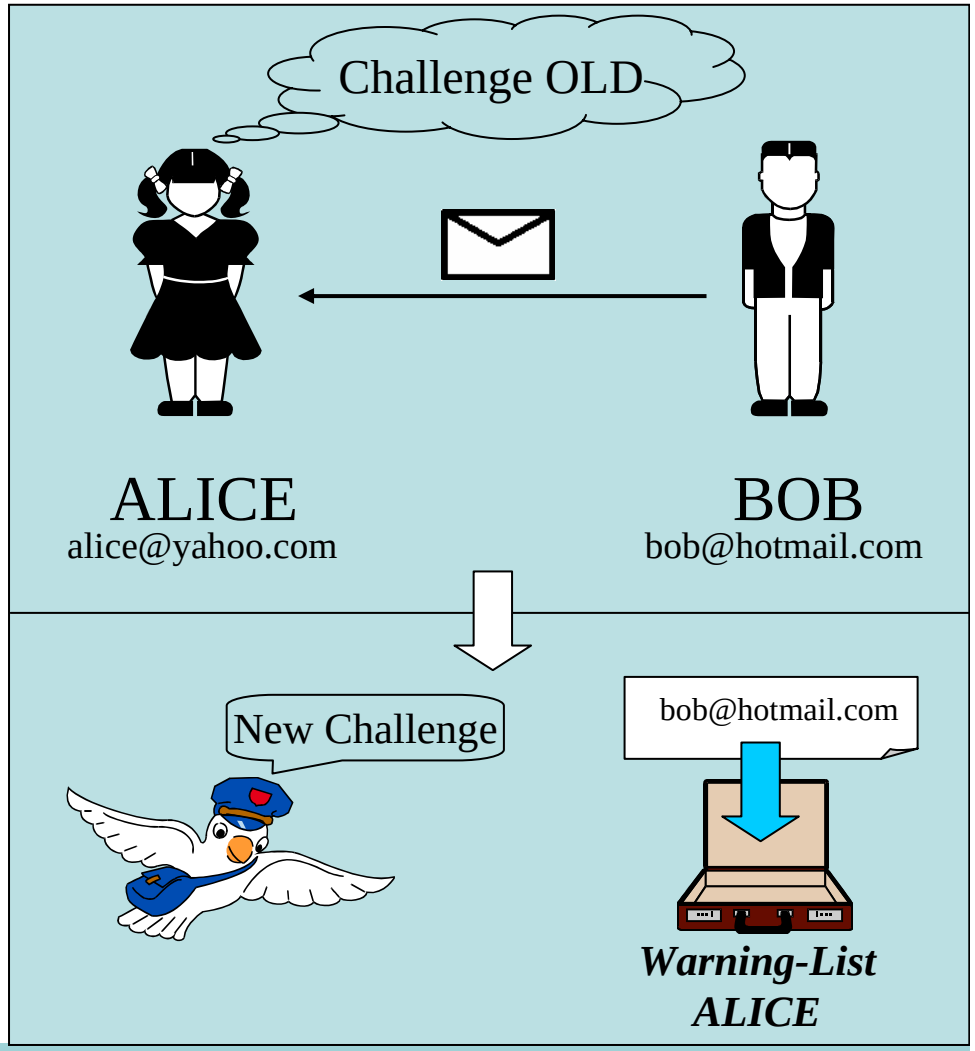
- If B is not listed in any list, the system checks whether the challenge of the email has been solved. If it is solved, the mail reaches A's mailbox and B is added to the *white-list*.
- Additionally, B receives a confirmation email.



# Pre-Challenge - How it Works

(4’)

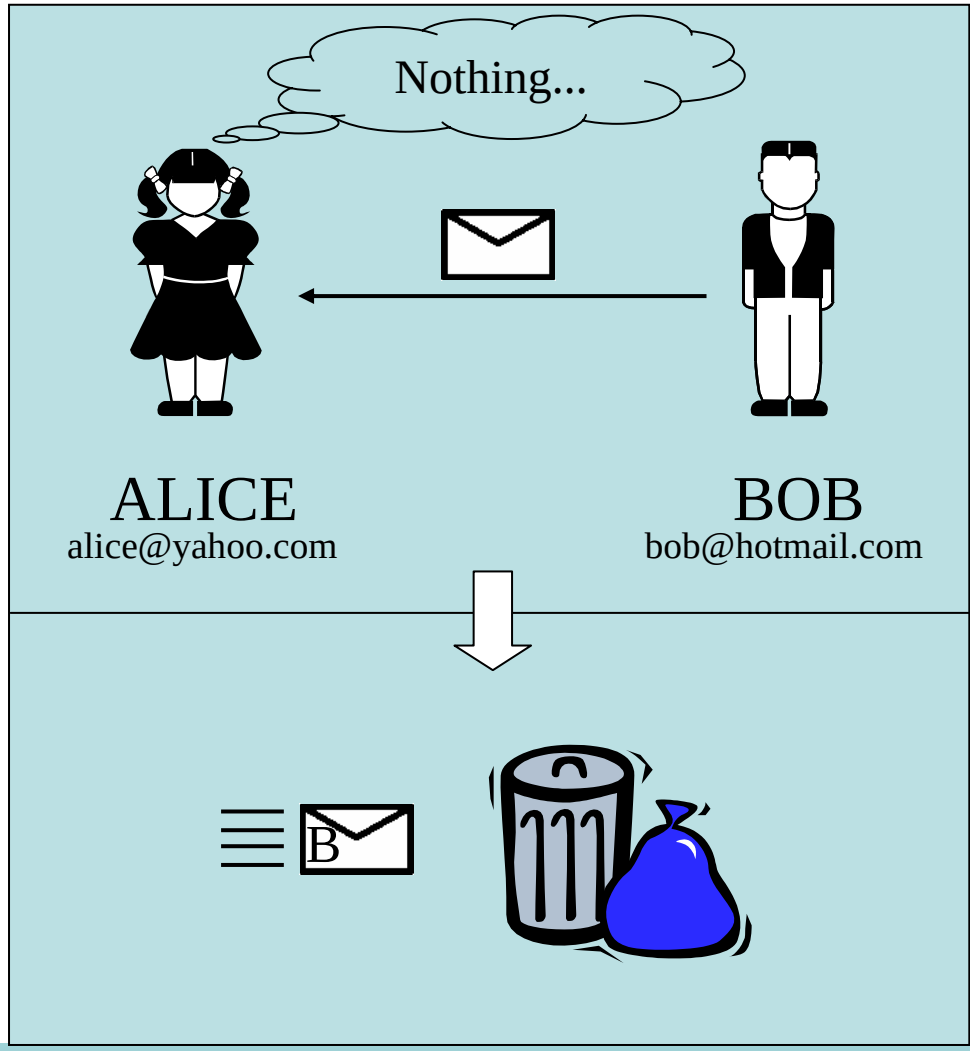
- If it is not solved but the message has a solution to an old challenge, the system checks if B is listed in the *warning-list*. If that is the case, the mail is discarded.
- Otherwise, B's address is added to the *warning-list* and B gets a reply containing information about the new challenge.



# Pre-Challenge - How it Works

(5’)

- If it is not solved and has no solution, the email is discarded without any reply to B indicating this fact



# Pre-Challenge - Spam Scenarios

First Scenario: Spammer harvest email, not pre-challenge

- Spam cannot achieve recipient's email - discarded without notice

Second Scenario: Spammer harvest email and pre-challenge

- Spammer must solve the challenge. In normal situations, only a human mind can do this.
- Spammer can achieve a single mailbox… but for being profitable he/she must achieve *millions* of mailboxes!
- Spammer can interchange challenge solutions (CDs!), or hire cheap labor - costly!
    - Users can change their challenge anytime - trashes inversion

# Discussions

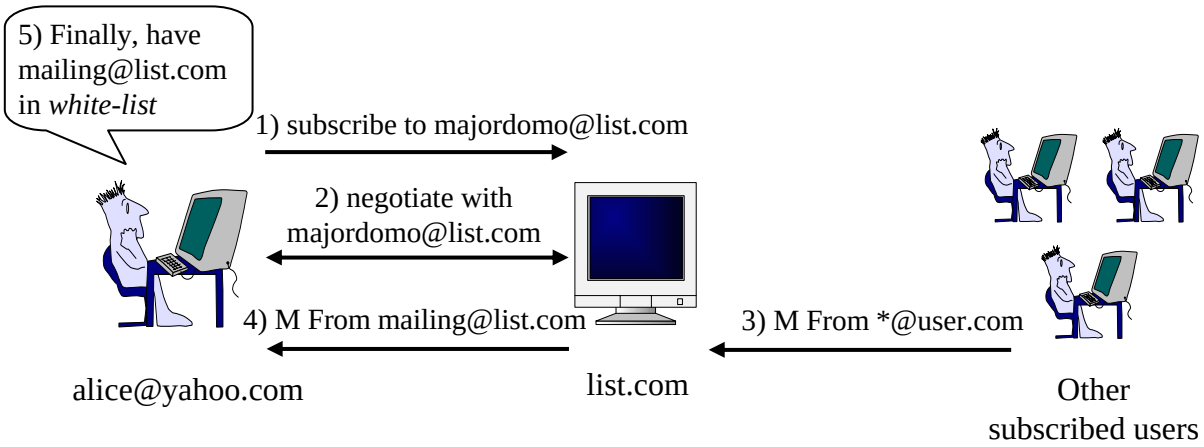
# Discussions - Mailing Lists

Problems

- Subscribing a mailing list means solving challenges managed by machines. Process not standard.

And Solutions

- We use Low security until ending the subscription



# Discussions - Availability

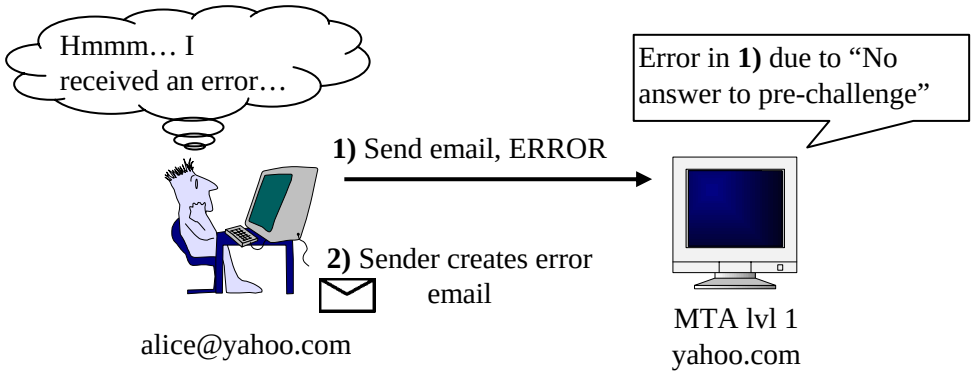
Where the challenge is available? (next to the user email address)

- If not stored along with the e-mail address (e.g. URI pointing to the challenge), or in static place (e.g. business card), problems!
- Maybe the challenge is
  - Not accessible
  - Outdated
- Recommendation: Store challenge and pointer (e.g. URI) to the actual challenge in the same place
  - Thanks to the *Warning-List* feature, there is no problems
- Unsolved problem: challenge can be impossible to solve for a disabled user without help

# Discussions - Accesibility

Problem

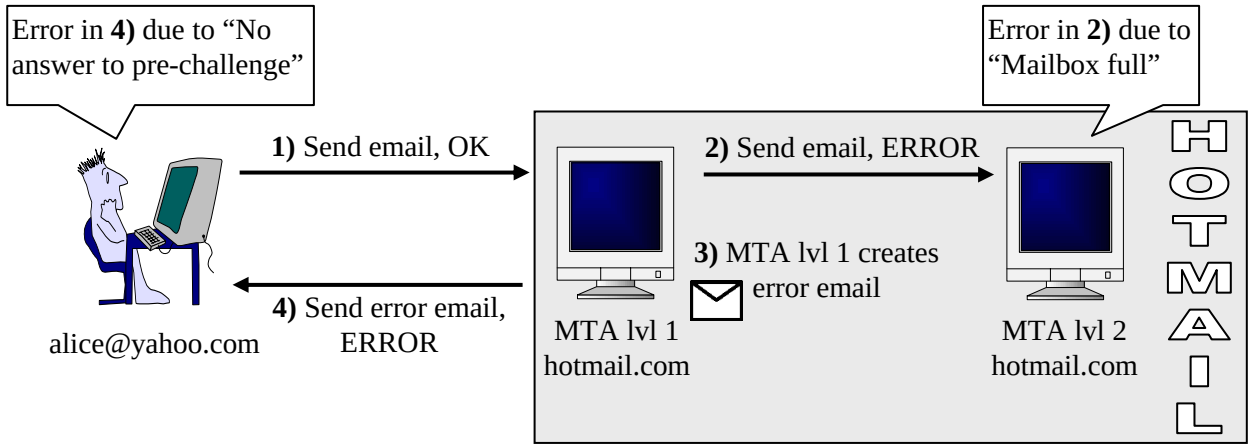
- Pre-Challenge trashes “wrong” e-mails without notice. People may think that the receiver got his/her e-mail and ignored it!
- More a social problem than a design problem
- Solution? : Use SMTP mechanisms for notifying failures



# Discussions - Error Messages

Problem

- An email delivery can fail: “Invalid recipient”, “Mailbox full”, “Invalid Pre-Challenge” $\Rightarrow$ sender creates an error message
- Error message created by computer! No pre-challenge, email trashed in some cases!



# Discussions - Error Messages

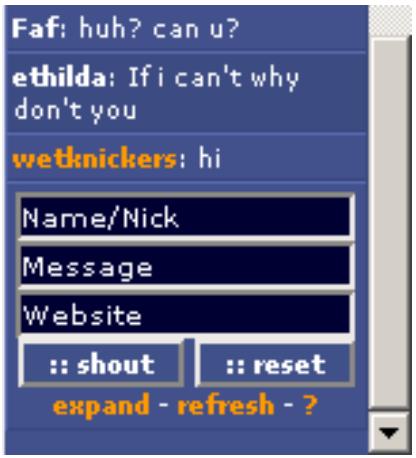
Solution

- Error messages can be detected, and have attached the mail that caused the error: includes recipient address and ID of the email
- Query the *reply-list* when error received - use address and ID (the error message is the “reply” to a email we sent).

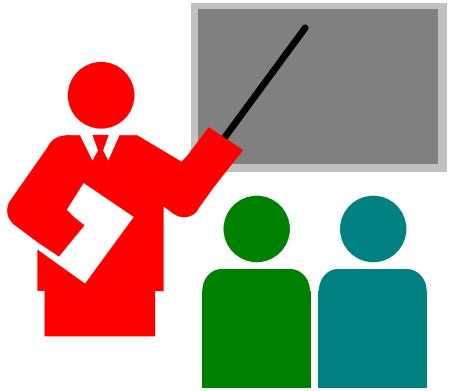
# Discussions - New Applications

Pre-Challenge can be used for other purposes!

- Example: IM Systems (MSN, AOL Messenger, ICQ, …)
- Some features are prone to receive spam! (ICQ World-Wide Pager, Shoutboxes)
- Use Pre-Challenge scheme for avoiding machine-based Spam

# Conclusions



# Conclusions

Benefits

- Standalone solution (no need to change other side)
- Does not create inconvenience to normal users
- Manages mailing list messages and error messages
- There is no delay on receiving e-mails
- There is no possibility of a DDoS
- Avoids email harvesting problems
- Also can be applied for other services!

Reaches a good balance between security against spam and convenience to normal users


Thanks for your
attention