Push vs. Pull: Implications of Protocol Design on Controlling Unwanted Traffic

Zhenhai Duan, Kartik Gopalan

Florida State University

Yingfei Dong

University of Hawaii

1
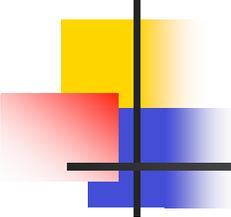
Objectives and Scope

- Controlling spam-like unwanted traffic
  - We target unsolicited asynchronous messages
  - These rely on content being read/heard by the receiver

- Two objectives
  1. Examine two protocol design choices
     - Sender-push vs. receiver-pull
  2. Study the feasibility of using receiver-pull for asynchronous message applications
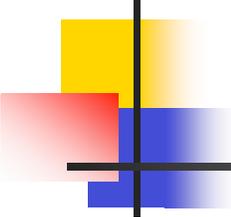
Outline of the Talk

- Message delivery models and their variants
  - Sender-Push (SP)
  - Receiver-Intent-based Sender-Push (RISP)
  - Receiver-Pull (RP)
  - Sender-Intent-based Receiver-Pull (SIRP)

- A simple receiver-pull-based email delivery system
  - The Differentiated Mail Transfer Protocol (DMTP)

- Summary
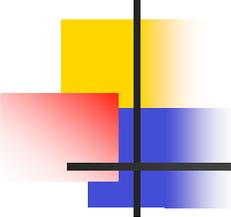
Sender-Push (SP)

Sender →*Content Push*→ Receiver

(a) Sender Push

- Examples: SMTP-based email, asynchronous voice messages

- Roles
  - S: Controls what content is delivered and when it is delivered
  - R: Passively receives the *entire message* before processing/discarding

- Responsibilities
  - S: Prepare and transmit message when ready
  - R: Has to wait, receive, process, store/discard each message.

- Accountability
  - Senders can vanish after pushing messages

Receiver-Intent-based Sender-Push (RISP)

(1) Intent to receive

Sender ← — — — → Receiver

(2) Content Push

(b) Receiver Intent Based Sender Push

- Examples:
  - Mailing lists, subscription-based stock/news ticker, instant messaging.

- Receiver can exercise minimal control over sender
  - Subscribe/unsubscribe
  - Whitelist/blacklist

- Basic problems for SP
  - Senders control what/when to send
  - Receiver must accept entire message.
  - Poor accountability

5

Receiver-Pull (RP)

Sender | Content Pull | → Receiver
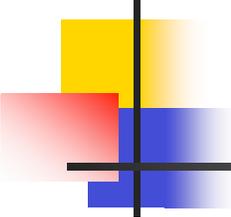
(c) Receiver Pull

- Examples: HTTP and FTP
- Roles
  - S: Stores the message and passively waits for retrieval
  - R: Controls if and when to retrieve the message

- Responsibilities
  - S: Prepare, store, manage the content and wait (stay online)
  - R: Retrieve the message when convenient

- Accountability
 - Sender's identity is visible for a larger window of time

Sender-intent-based Receiver Pull (SIRP)

(1) Intent to send

Sender → Receiver

(2) Content Pull

(d) Sender Intent Based Receiver Pull

- Example: Pager service

- Allow senders to express short intent to send a message
  - Content delivery is still controlled by receiver

- Primary advantages of RP
  - Receivers control delivery
  - Senders commit more resources
  - Senders can be held accountable
    - Senders cannot vanish before message is retrieved

- Disadvantage:
  - To some extent, intent notice may itself be considered as SPAM.
  - Definitely better than receiving the whole message.

SIRP Email Architecture
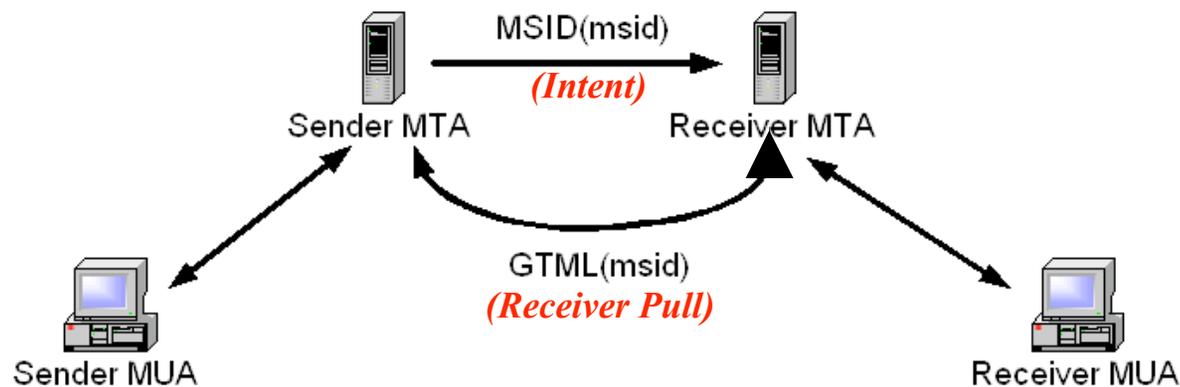
Issue: All messages, whether spam or legitimate, are affected by the two-step delivery
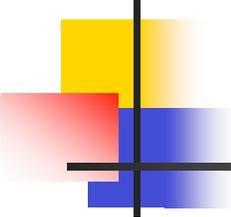
DMTP: Differentiated Mail Transfer Protocol

- Classify the senders
  - Allowed – regular contacts
  - Denied – well-known spammers
  - Unclassified – anyone not in allowed/denied

- Differentiate delivery of messages based on sender classification
  - Allowed: Directly accept the entire message
  - Denied: Directly decline the message *before* content is delivered.
  - Unclassified: Use the SIRP model to retrieve message

- Classification granularity at
  - MTA level and
  - (optionally) Email address level

DMTP Message Reception
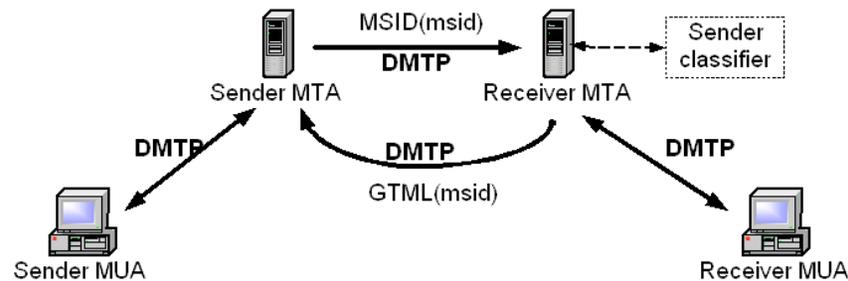

```
If (SMTA is Denied)
    return 550 (PE)
  close TCP session
else (SMTA is allowed)
  proceed using SMTP
else /* SMTA is unclassified */
  accept MSID
  (reject any DATA command)
  /* pull message later if and when user wants */
```
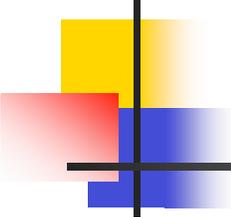
Example DMTP transactions

SMTA: open TCP connection
RMTA: Get IP address of SMTA

// Case 1: SMTA IP is Allowed
RMTA: 220

// Case 2: SMTA IP is Denied
RMTA: 550
RMTA: close TCP connection

// Case 3: SMTA IP is Unclassified
RMTA: 220
SMTA: EHLO domain.com
RMTA: 220 MSID
SMTA: MAIL FROM: <yyy> DMTP
RMTA: 220
SMTA: RCPT TO: <xxx>
RMTA: 220
SMTA: MSID <identifier>
RMTA: 220

// if DATA command is attempted
SMTA: DATA
RMTA: 550
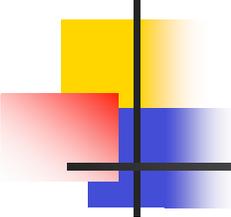
Other aspects

- *DMTP can be incrementally deployed*
  - *No need to change everyone from SMTP ➜ DMTP overnight*

- SIRP model is also applicable to mobile text messages, asynchronous voice message etc.

- Other references:
  - *Receiver-Driven Extensions to SMTP*, Internet Draft
  - *DiffMail: Controlling Spam Through Message Differentiation*, TR, FSU
  - *DiffMail Project webpage*: http://www.cs.fsu.edu/~duan/projects/diffmail/
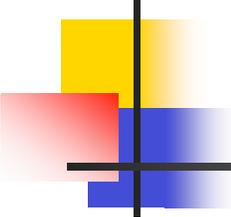
Summary

- We examined two message delivery models and their variants
    - Receiver-pull model preferred in controlling unwanted messages

- Presented application of receiver-pull to email delivery
    - Differentiated Mail Transfer protocol (DMTP)
    - Currently implementing DMTP in Sendmail.

    (code to be available soon)

- Thank you!