DANE for SMTP

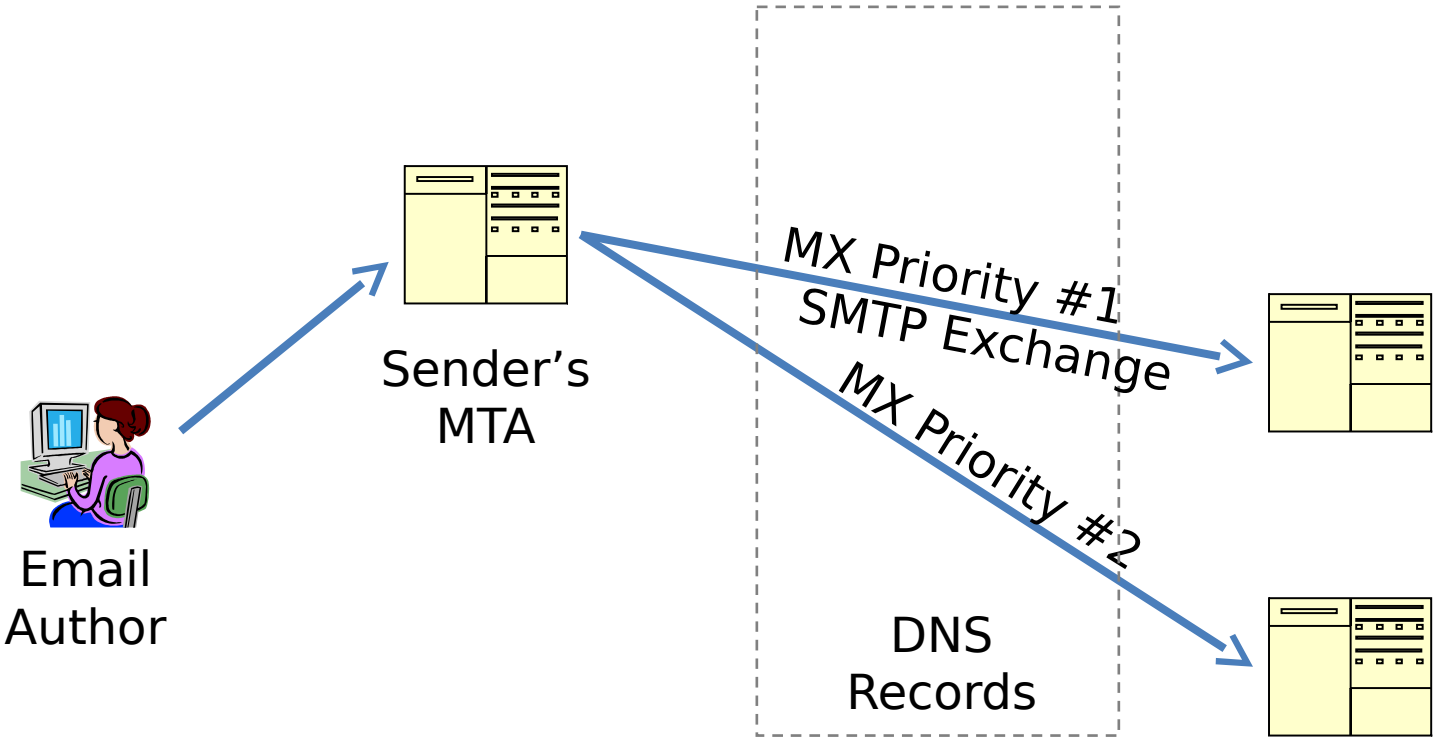
Viktor Dukhovni
&
Wes Hardaker

IETF 87, Berlin
July 2013

Addresses in SMTP

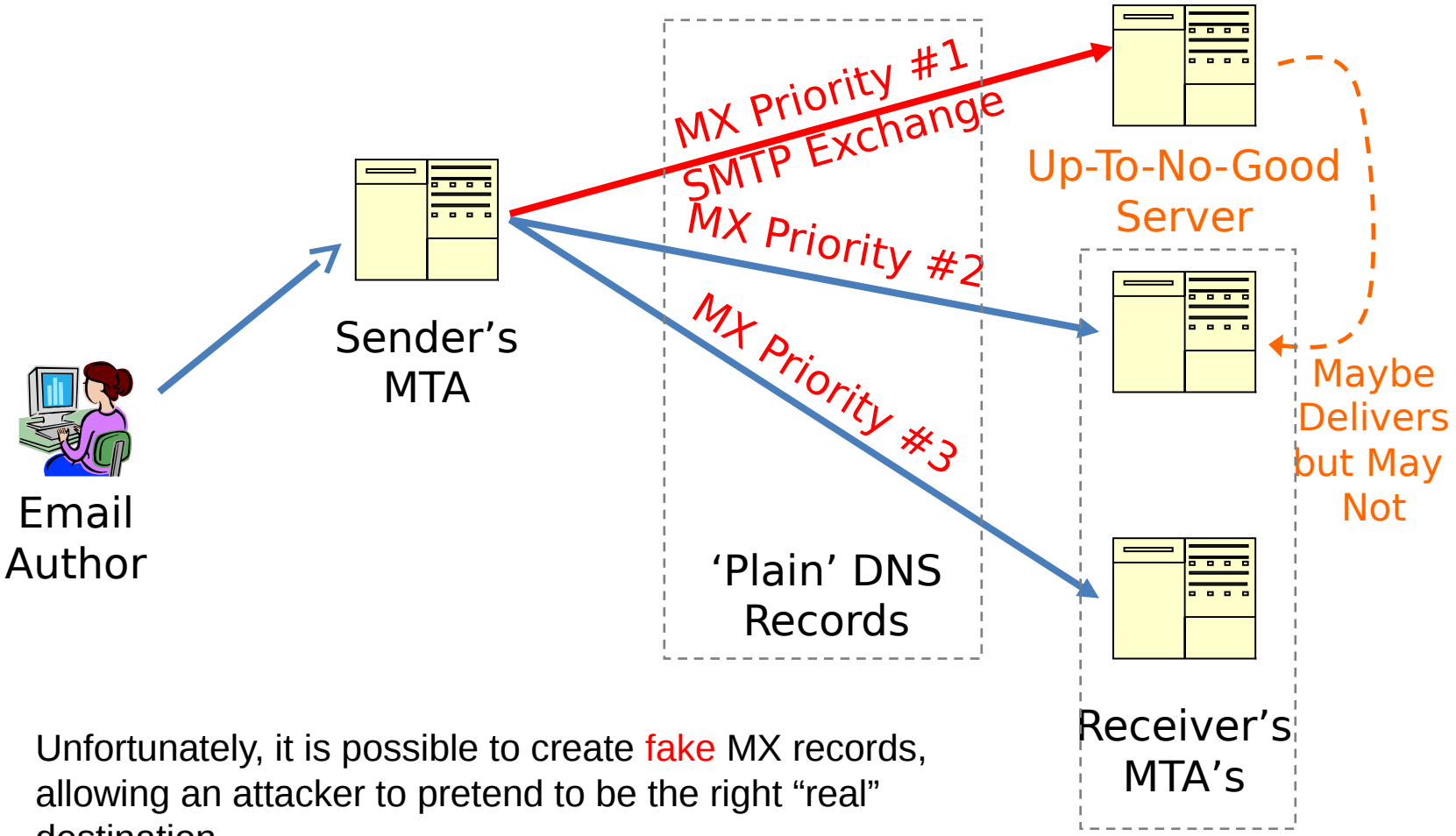
- <mailbox@example.com> is security agnostic:
  - SMTP with and without TLS runs over port 25
    - There is no URI *scheme* to designate “SMTP” vs. “SMTPS”
    - STARTTLS is used to signal TLS support
  - SMTP is multi-hop store & forward
    - TLS is **hop-by-hop**
    - SMTP addresses are NOT transport addresses
    - Typical minimum number of hops is 3
    - Some may be protected, some may not
  - MX RRs abstract hop destinations via DNS

Sending E-Mail Today



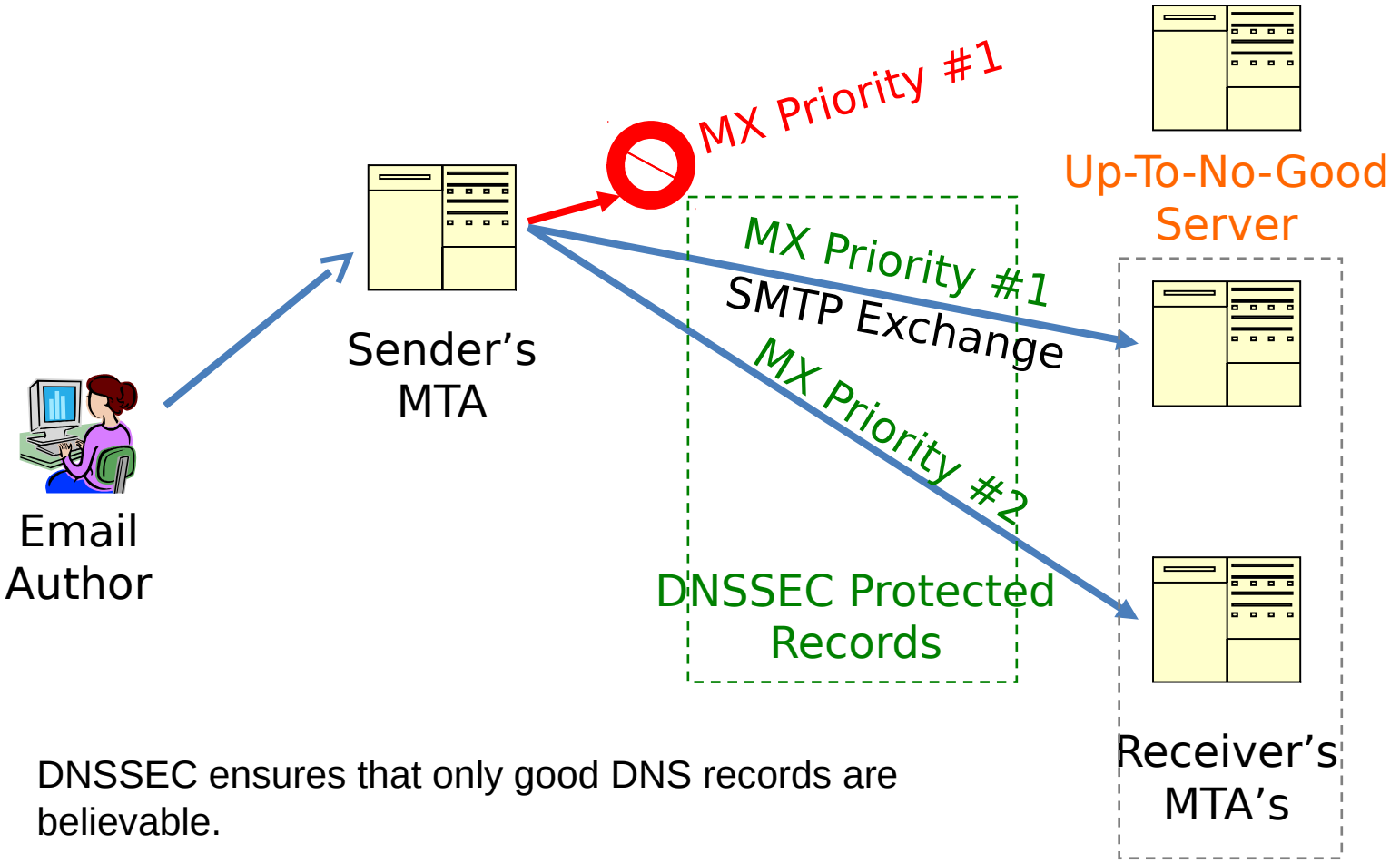
1. Sender transmits to their Mail Transfer Agent (MTA)
2. MTA uses the receiver's DNS "MX" records to find a destination MTA
3. The sender's MTA sends email to the receiver's MTA

Problem #1: Fake MX Records



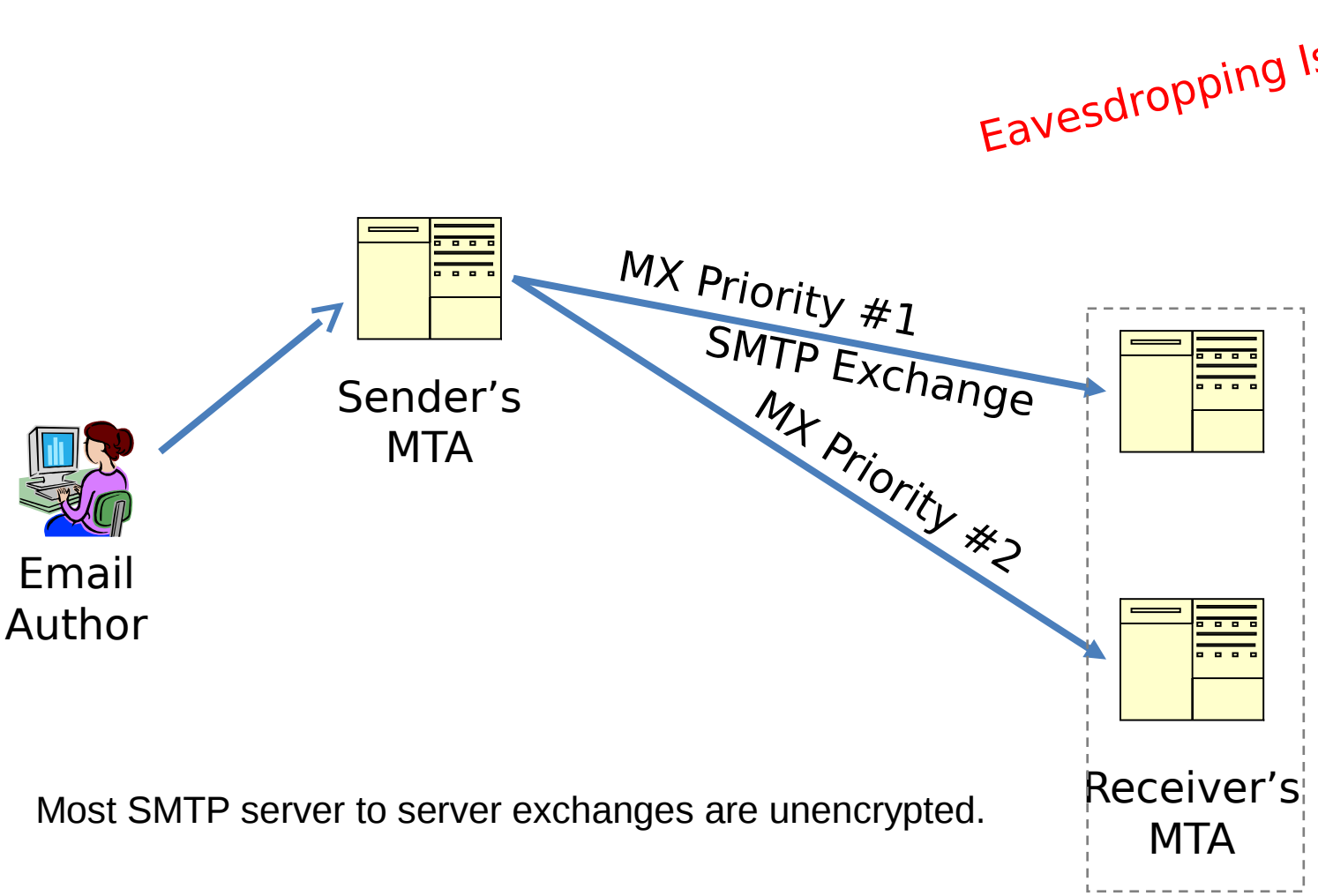
Unfortunately, it is possible to create fake MX records, allowing an attacker to pretend to be the right "real" destination.

Solution #1: DNSSEC



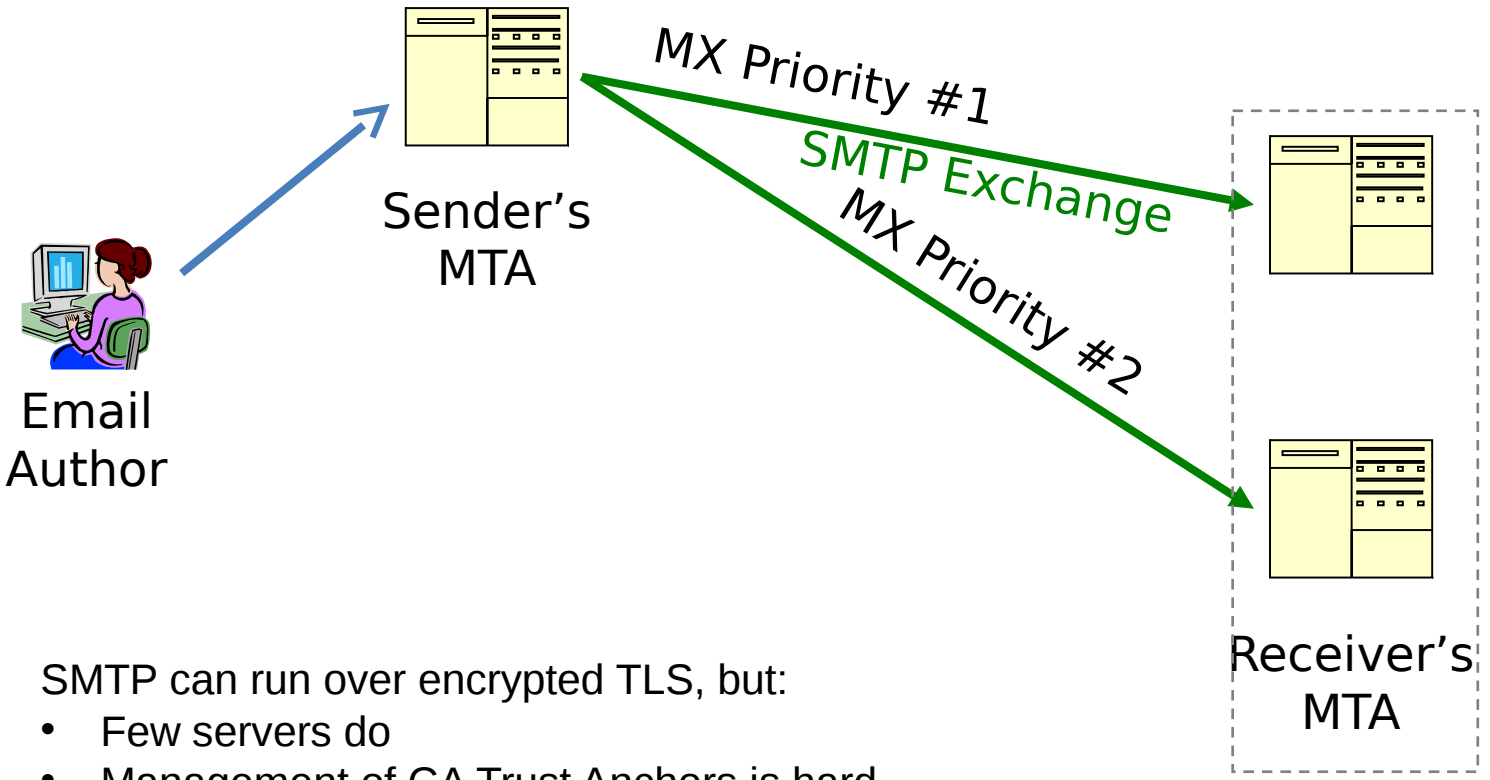
DNSSEC ensures that only good DNS records are believable.

Problem #2: Unprotected SMTP



Most SMTP server to server exchanges are unencrypted.

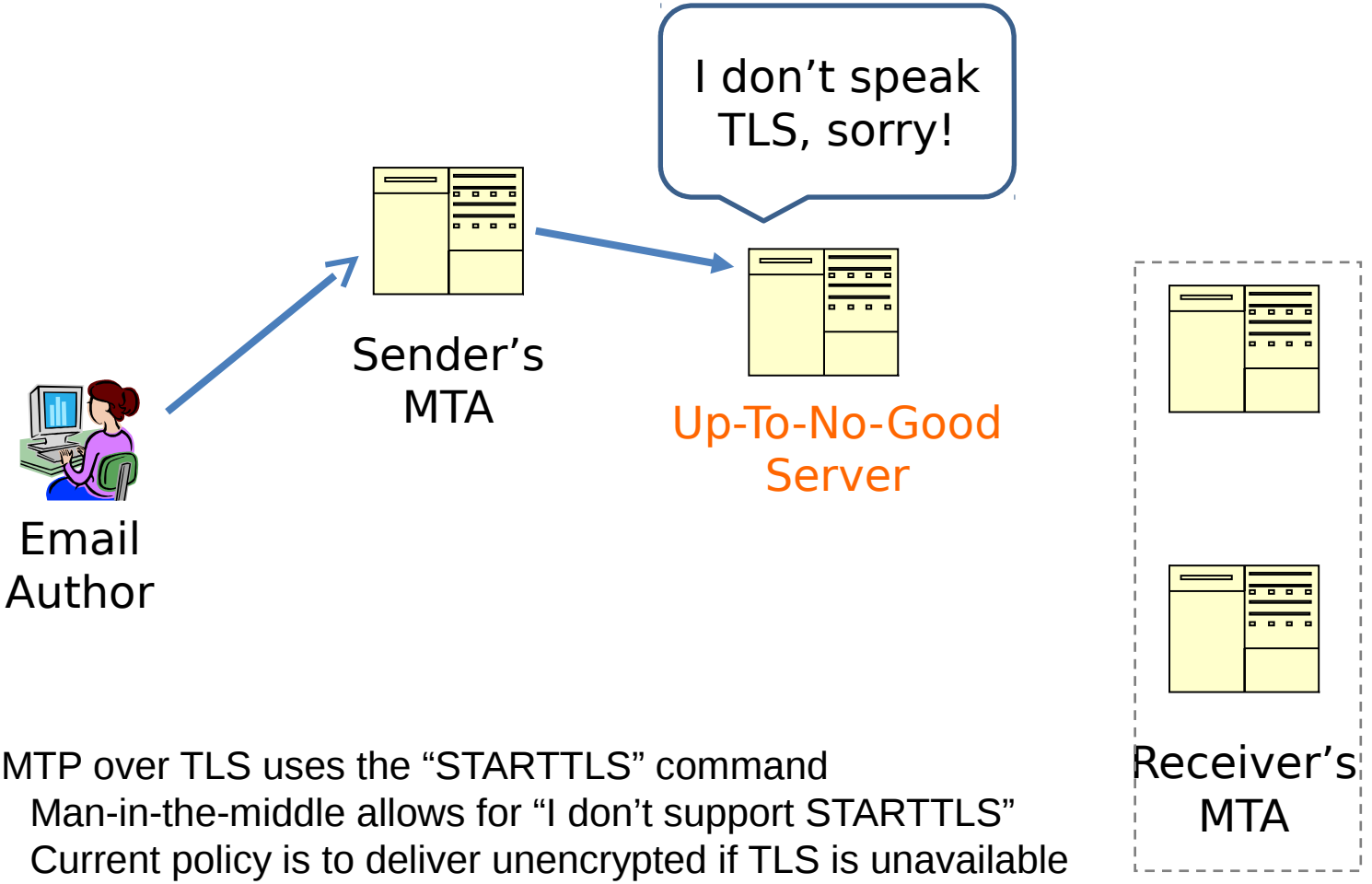
Solution #2: TLS-Protected SMTP



SMTP can run over encrypted TLS, but:

- Few servers do
- Management of CA Trust Anchors is hard
- MTA software doesn't distribute TAs

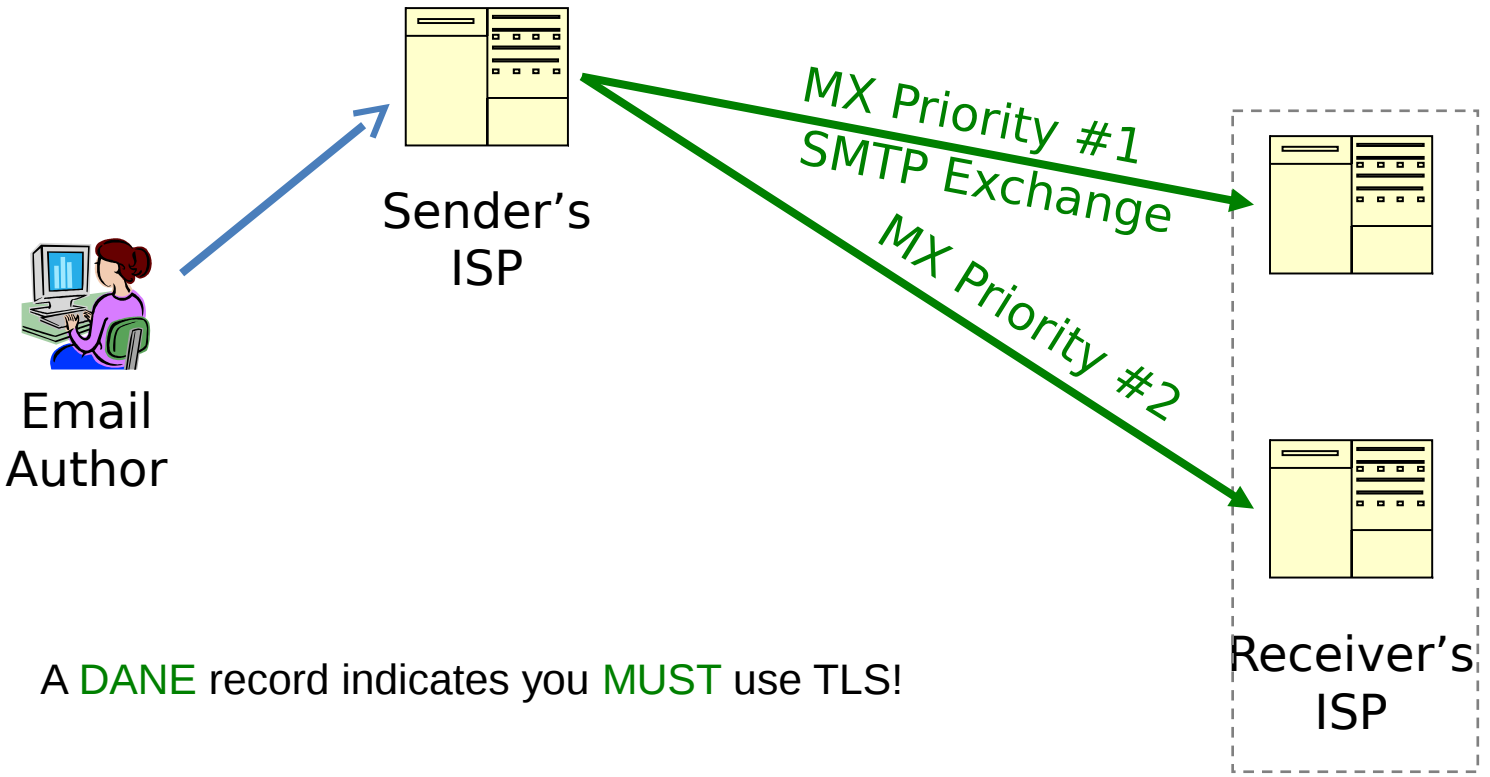
Problem #3: SMTP Man-in-the-Middle



SMTP over TLS uses the "STARTTLS" command

- Man-in-the-middle allows for "I don't support STARTTLS"
- Current policy is to deliver unencrypted if TLS is unavailable

Solution #3: SMTP over TLS with DANE



A DANE record indicates you MUST use TLS!

TLS and SMTP Summary

- MX/A... RRsets are insecure without DNSSEC
- Sender does not know when or how to use TLS
  - Except via administrative policy
- There is no user to click “OK”
  - Security must “just work”
  - With no “MUST use” signal, fallback to no-TLS
- STARTTLS allows for MITM downgrade attack

DANE and SMTP

- With DNSSEC and DANE we can:
  - Harden MX lookup via DNSSEC
  - Provide downgrade-resistant TLS support
  - Publish authentication public key digests (or keys)
  - Incremental adoption, without bilateral coordination!
    - It turns on automatically when both sides support it

DANE and SMTP

- SMTP TLS security depends on DNSSEC
  - If DNSSEC is broken, all bets are off
    - CAs and TLS alone fail to secure the transport
  - Usage 0 has same DNSSEC exposure as usage 2
  - Usage 1 has same DNSSEC exposure as usage 3
- Some MTAs (Exim and Postfix) have stated:
  - they may map 0 → 2 and 1 → 3
  - Will have empty CA lists by default

SMTP Referral Choices

- Host SMTP yourself
  - Good: MX to your own internal name
  - Ehhh: No MX: CNAME to your mail host (legal)
- Outsource SMTP service
  - Good: MX exchange name to their name
  - Ehhh: No MX: Use CNAMEs
    (legal but discouraged)
  - Bad: Copy their A and TLSA records to your zone
    (and point MX to your copies)
- Don't do this anywhere (illegal):
  - Ugly: MX records point to a CNAME

SMTP hosting example

- Easy example - MX to the outsourced name:
  - In client.com's zone:

```
client.com.                    IN MX 1 mx1.provider.com.
```

  - In provider.com's zone:

```
mx1.provider.com.              IN A ….
_25._tcp.mx1.provider.com.     IN TLSA …
```

- SNI:
  - uses mx1.provider.com.
  - Only a single certificate is needed
    (for “mx1.provider.com”)

TLSA records and MX records

- MX exchange name → TLSA base domain
  - This is the TLS transport end-point
  - Certificate peername:
    - SHOULD be the TLSA base name
    - MAY be the domain of the email address or configured transport domain
  - The provider publishes the TLSA record for their keys
    - The client simply points and doesn't publish data
  - SNI not essential to support multiple client domains
- Operational guidance:
  - Use MX records this way!

DANE SMTP Model Summary

- Opportunistic and downgrade-resistant
- No interactive user: reliability must be paramount
- Trusts DNSSEC
  - Recommends type 3 and then 2
  - Type 0 and 1 usage are undefined and SHOULD NOT be used
- Certificate chains MUST include the TA

What To Do With This Work?

- Merge content with draft-ietf-dane-smtp?
- Publish different components separately?

Extra Slides

PKIX and SMTP

- Handy for bilateral secure-channels
  - Manually configured TLS expectation policies and keys
  - Explicit TLS requirement not dependent on competence of remote DNS operator
  - Explicit sender choice of (agreed upon) CA(s)
  - No bleeding edge code, tested TLS PKI.
  - However fragile when peer switches MX providers, CAs, etc. without notice.
  - Only viable for peer sites willing to coordinate infrastructure changes with sender!

SMTP Hosting With CNAMEs

- RFC 5321 Section 5.1:
  - If a CNAME record is found, the resulting name is processed as if it were the initial name.
- Many domains are MX-hosted by outside providers.
  - Almost always via MX RRs
  - CNAMEs are an edge case:
    - “bob@some-cname.example.com” is rare
    - “mail.example.com IN MX 1 cname.example.com” is illegal
  - Transport mappings, however, use CNAMEs
    - My server directly maps example.com to smtp.example.net

SMTP hosting via MX CNAME (illegal)

- MTAs may support RFC non-conformant CNAMEs in MX hostnames.
- Example, MX host a CNAME in provider's zone:
  - In client.com's zone:

```
client.com.              IN MX    1 mx1.provider.com
```

  - In provider.com's zone:

```
mx1.provider.com.    IN CNAME realmx.provider.com.
realmx.provider.com.   IN A     192.0.2.1
_25._tcp.realmx.provider.com. IN TLSA …
```

- Works with TLSA RR at either MX exchange name
  - Or CNAME target if MTA and DNS operator agree.

SMTP hosting via MX CNAME (illegal)

- Example, MX host a CNAME in client's zone:
  - In client.com's zone:

```
client.com.                 IN MX    1 mx.client.com.
mx.client.com               IN CNAME mx.provider.com.
_25._tcp.mx.client.com.     IN TLSA …                  ; (case I)
```

  - In provider.com's zone:

```
mx.provider.com.            IN A 192.0.2.1
_25._tcp.mx.provider.com.   IN TLSA …                  ; (case II)
```

- Case I (looking for a TLSA before CNAME expansion) problematic:
  - Provider must use a per-client certificate to match each client's MX base domain.
  - TLSA record must be copied/tracked from provider's server(s).
- Case II (looking for a TLSA record after CNAME expansion):
  - works just fine if MTA chases CNAMEs on MX records.
- Proposal: MTAs must chase CNAMEs to determine TLSA base name.

DNS and SMTP

- DNS trust unavoidable
  - Only place to store hop-by-hop security requirements
  - TLS Peername checks must trust the DNS
    - (they're pulling the peername **from** the MX record)
  - Without DNSSEC:
    - Spoofable MX records
    - Downgrade vulnerable TLS
    - No authentication

SMTP hosting and IP copying

- Harder example - MX points to an internal name:
  - In client.example.com's zone:

```
client.com.          IN MX 1 intmx.client.com.
; data copied from provider's records:
intmx.client.com.  IN  A 192.0.2.1 ; provider's address
_25._tcp.intmx.client.com. IN TLSA …
```

- Requires client copy outsource data
  - Bad practice for A records
  - Bad practice for TLSA records
  - SNI scaling issues with large # of certificates
  - Operational Guidance: Don't do this