SECTION 03

# Encrypted is not the same as authenticated

SES encrypts the email but does not verify the recipient. Self-signed and expired certificates all delivered, regardless of configuration.

# SES delivered every certificate failure scenario

TLS is supposed to do two things: encrypt the email, and verify that the receiving server is who it claims to be. Verification relies on the server's certificate. When that certificate is invalid, the encrypted email could be heading to an impostor instead of the intended recipient.

We tested how certificates can fail, under both SES configurations:

- A certificate the receiver signed themselves instead of having it issued by a trusted authority
- A certificate that has expired

**0 of 4**

certificate failures were blocked by Amazon SES.

| RECEIVER CERTIFICATE STATE | SES DEFAULT | SES "REQUIRE TLS" |
|---|---|---|
| Self-signed | ⚠ Delivered | ⚠ Delivered |
| Expired | ⚠ Delivered | ⚠ Delivered |

# An encrypted session to an unverified recipient is not a secure delivery

The "Require TLS" configuration did not change the outcome in any of the four certificate tests. Combined with the version-downgrade finding, "Require TLS" performs exactly one function.

Each `Received` header recorded a fully encrypted session, with `TLSv1.3` and the strong `TLS_AES_256_GCM_SHA384` algorithm appearing on every successful delivery. The connection was encrypted. The recipient on the other end of it was never verified.

The practical consequence is active man-in-the-middle exposure.

Certificate transparency logs, hostname validation, and expired certificate rejection are all standard Paubox practice for handling sensitive data over email.

**1 of 4**

**failure modes "Require TLS" actually blocks.** Only the no-TLS case bounces. TLS 1.0, TLS 1.1, and invalid certificates all still pass through.

PAUBOX

# Secure every email you send and receive

Paubox combines healthcare compliance with AI to protect organizations from the most sophisticated attacks.

Talk to an expert