Internet Engineering Task Force (IETF)
Request for Comments: 8314
Updates: 1939, 2595, 3501, 5068, 6186, 6409
Category: Standards Track
ISSN: 2070-1721

K. Moore
Windrock, Inc.
C. Newman
Oracle

January 2018

Cleartext Considered Obsolete: Use of Transport Layer Security (TLS) for Email Submission and Access

Abstract

This specification outlines current recommendations for the use of Transport Layer Security (TLS) to provide confidentiality of email traffic between a Mail User Agent (MUA) and a Mail Submission Server or Mail Access Server. This document updates RFCs 1939, 2595, 3501, 5068, 6186, and 6409.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8314.

Table of Contents

1. Introduction

Software that provides email service via the Internet Message Access Protocol (IMAP) [RFC3501], the Post Office Protocol (POP) [RFC1939], and/or Simple Mail Transfer Protocol (SMTP) Submission [RFC6409] usually has Transport Layer Security (TLS) [RFC5246] support but often does not use it in a way that maximizes end-user confidentiality. This specification describes current recommendations for the use of TLS in interactions between Mail User Agents (MUAs) and Mail Access Servers, and also between MUAs and Mail Submission Servers.

In brief, this memo now recommends that:

- TLS version 1.2 or greater be used for all traffic between MUAs and Mail Submission Servers, and also between MUAs and Mail Access Servers.

- MUAs and Mail Service Providers (MSPs) (a) discourage the use of cleartext protocols for mail access and mail submission and (b) deprecate the use of cleartext protocols for these purposes as soon as practicable.

- Connections to Mail Submission Servers and Mail Access Servers be made using "Implicit TLS" (as defined below), in preference to connecting to the "cleartext" port and negotiating TLS using the STARTTLS command or a similar command.

This memo does not address the use of TLS with SMTP for message relay (where Message Submission [RFC6409] does not apply). Improving the use of TLS with SMTP for message relay requires a different approach. One approach to address that topic is described in [RFC7672]; another is provided in [MTA-STS].

The recommendations in this memo do not replace the functionality of, and are not intended as a substitute for, end-to-end encryption of electronic mail.

1.1. How This Document Updates Previous RFCs

This document updates POP (RFC 1939), IMAP (RFC 3501), and Submission (RFC 6409, RFC 5068) in two ways:

1. By adding Implicit TLS ports as Standards Track ports for these protocols as described in Section 3.

2. By updating TLS best practices that apply to these protocols as described in Sections 4 and 5.

This document updates RFC 2595 by replacing Section 7 of RFC 2595 with the preference for Implicit TLS as described in Sections 1 and 3 of this document, as well as by updating TLS best practices that apply to the protocols in RFC 2595 as described in Sections 4 and 5 of this document.

This document updates RFC 6186 as described herein, in Section 5.1.

2. Conventions and Terminology Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The term "Implicit TLS" refers to the automatic negotiation of TLS whenever a TCP connection is made on a particular TCP port that is used exclusively by that server for TLS connections. The term "Implicit TLS" is intended to contrast with the use of STARTTLS and similar commands in POP, IMAP, SMTP Message Submission, and other protocols, that are used by the client and the server to explicitly negotiate TLS on an established cleartext TCP connection.

The term "Mail Access Server" refers to a server for POP, IMAP, and any other protocol used to access or modify received messages, or to access or modify a mail user’s account configuration.

The term "Mail Submission Server" refers to a server for the protocol specified in [RFC6409] (or one of its predecessors or successors) for submission of outgoing messages for delivery to recipients.

The term "Mail Service Provider" (or "MSP") refers to an operator of Mail Access Servers and/or Mail Submission Servers.

The term "Mail Account" refers to a user’s identity with an MSP, that user’s authentication credentials, any user email that is stored by the MSP, and any other per-user configuration information maintained by the MSP (for example, instructions for filtering spam). Most MUAs support the ability to access multiple Mail Accounts.

For each account that an MUA accesses on its user’s behalf, it must have the server names, ports, authentication credentials, and other configuration information specified by the user. This information, which is used by the MUA, is referred to as "Mail Account Configuration".

This specification expresses syntax using the Augmented Backus-Naur Form (ABNF) as described in [RFC5234], including the core rules provided in Appendix B of [RFC5234] and the rules provided in [RFC5322].

3. Implicit TLS

Previous standards for the use of email protocols with TLS used the STARTTLS mechanism: [RFC2595], [RFC3207], and [RFC3501]. With STARTTLS, the client establishes a cleartext application session and determines whether to issue a STARTTLS command based on server capabilities and client configuration. If the client issues a STARTTLS command, a TLS handshake follows that can upgrade the connection. Although this mechanism has been deployed, an alternate mechanism where TLS is negotiated immediately at connection start on a separate port (referred to in this document as "Implicit TLS") has been deployed more successfully. To encourage more widespread use of TLS and to also encourage greater consistency regarding how TLS is used, this specification now recommends the use of Implicit TLS for POP, IMAP, SMTP Submission, and all other protocols used between an MUA and an MSP.

3.1. Implicit TLS for POP

When a TCP connection is established for the "pop3s" service (default port 995), a TLS handshake begins immediately. Clients MUST implement the certificate validation mechanism described in [RFC7817]. Once the TLS session is established, POP3 [RFC1939] protocol messages are exchanged as TLS application data for the remainder of the TCP connection. After the server sends an +OK greeting, the server and client MUST enter the AUTHORIZATION state, even if a client certificate was supplied during the TLS handshake.

See Sections 5.5 and 4.2 for additional information on client certificate authentication. See Section 7.1 for port registration information.

3.2. Implicit TLS for IMAP

When a TCP connection is established for the "imaps" service (default port 993), a TLS handshake begins immediately. Clients MUST implement the certificate validation mechanism described in [RFC7817]. Once the TLS session is established, IMAP [RFC3501] protocol messages are exchanged as TLS application data for the remainder of the TCP connection. If a client certificate was provided during the TLS handshake that the server finds acceptable, the server MAY issue a PREAUTH greeting, in which case both the

server and the client enter the AUTHENTICATED state. If the server issues an OK greeting, then both the server and the client enter the NOT AUTHENTICATED state.

See Sections 5.5 and 4.2 for additional information on client certificate authentication. See Section 7.2 for port registration information.

3.3. Implicit TLS for SMTP Submission

When a TCP connection is established for the "submissions" service (default port 465), a TLS handshake begins immediately. Clients MUST implement the certificate validation mechanism described in [RFC7817]. Once the TLS session is established, Message Submission protocol data [RFC6409] is exchanged as TLS application data for the remainder of the TCP connection. (Note: The "submissions" service name is defined in Section 7.3 of this document and follows the usual convention that the name of a service layered on top of Implicit TLS consists of the name of the service as used without TLS, with an "s" appended.)

The STARTTLS mechanism on port 587 is relatively widely deployed due to the situation with port 465 (discussed in Section 7.3). This differs from IMAP and POP services where Implicit TLS is more widely deployed on servers than STARTTLS. It is desirable to migrate core protocols used by MUA software to Implicit TLS over time, for consistency as well as for the additional reasons discussed in Appendix A. However, to maximize the use of encryption for submission, it is desirable to support both mechanisms for Message Submission over TLS for a transition period of several years. As a result, clients and servers SHOULD implement both STARTTLS on port 587 and Implicit TLS on port 465 for this transition period. Note that there is no significant difference between the security properties of STARTTLS on port 587 and Implicit TLS on port 465 if the implementations are correct and if both the client and the server are configured to require successful negotiation of TLS prior to Message Submission.

Note that the "submissions" port provides access to a Message Submission Agent (MSA) as defined in [RFC6409], so requirements and recommendations for MSAs in that document, including the requirement to implement SMTP AUTH [RFC4954] and the requirements of Email Submission Operations [RFC5068], also apply to the submissions port.

See Sections 5.5 and 4.2 for additional information on client certificate authentication. See Section 7.3 for port registration information.

3.4. Implicit TLS Connection Closure for POP, IMAP, and SMTP Submission

When a client or server wishes to close the connection, it SHOULD initiate the exchange of TLS close alerts before TCP connection termination. The client MAY, after sending a TLS close alert, gracefully close the TCP connection (e.g., call the close() function on the TCP socket or otherwise issue a TCP CLOSE ([RFC793], Section 3.5)) without waiting for a TLS response from the server.

4. Use of TLS by Mail Access Servers and Message Submission Servers

The following requirements and recommendations apply to Mail Access Servers and Mail Submission Servers, or, if indicated, to MSPs:

- MSPs that support POP, IMAP, and/or Message Submission MUST support TLS access for those protocol servers.

- Servers provided by MSPs other than POP, IMAP, and/or Message Submission SHOULD support TLS access and MUST support TLS access for those servers that support authentication via username and password.

- MSPs that support POP, IMAP, and/or Message Submission SHOULD provide and support instances of those services that use Implicit TLS. (See Section 3.)

- For compatibility with existing MUAs and existing MUA configurations, MSPs SHOULD also, in the near term, provide instances of these services that support STARTTLS. This will permit legacy MUAs to discover new availability of TLS capability on servers and may increase the use of TLS by such MUAs. However, servers SHOULD NOT advertise STARTTLS if the use of the STARTTLS command by a client is likely to fail (for example, if the server has no server certificate configured).

- MSPs SHOULD advertise their Mail Access Servers and Mail Submission Servers, using DNS SRV records according to [RFC6186]. (In addition to making correct configuration easier for MUAs, this provides a way by which MUAs can discover when an MSP begins to offer TLS-based services.) Servers supporting TLS SHOULD be advertised in preference to cleartext servers (if offered). In addition, servers using Implicit TLS SHOULD be advertised in preference to servers supporting STARTTLS (if offered). (See also Section 4.5.)

- MSPs SHOULD deprecate the use of cleartext Mail Access Servers and Mail Submission Servers as soon as practicable. (See Section 4.1.)

- MSPs currently supporting such use of cleartext SMTP (on port 25) as a means of Message Submission by their users (whether or not requiring authentication) SHOULD transition their users to using TLS (either Implicit TLS or STARTTLS) as soon as practicable.

- Mail Access Servers and Mail Submission Servers MUST support TLS 1.2 or later.

- All Mail Access Servers and Mail Submission Servers SHOULD implement the recommended TLS ciphersuites described in [RFC7525] or a future BCP or Standards Track revision of that document.

- As soon as practicable, MSPs currently supporting Secure Sockets Layer (SSL) 2.x, SSL 3.0, or TLS 1.0 SHOULD transition their users to TLS 1.1 or later and discontinue support for those earlier versions of SSL and TLS.

- Mail Submission Servers accepting mail using TLS SHOULD include in the Received field of the outgoing message the TLS ciphersuite of the session in which the mail was received. (See Section 4.3.)

- All Mail Access Servers and Mail Submission Servers implementing TLS SHOULD log TLS cipher information along with any connection or authentication logs that they maintain.

Additional considerations and details appear below.

4.1. Deprecation of Services Using Cleartext and TLS Versions Less Than 1.1

The specific means employed for deprecation of cleartext Mail Access Servers and Mail Submission Servers MAY vary from one MSP to the next in light of their user communities' needs and constraints. For example, an MSP MAY implement a gradual transition in which, over time, more and more users are forbidden to authenticate to cleartext instances of these servers, thus encouraging those users to migrate to Implicit TLS. Access to cleartext servers should eventually be either (a) disabled or (b) limited strictly for use by legacy systems that cannot be upgraded.

After a user's ability to authenticate to a server using cleartext is revoked, the server denying such access MUST NOT provide any indication over a cleartext channel of whether the user's authentication credentials were valid. An attempt to authenticate as such a user using either invalid credentials or valid credentials MUST both result in the same indication of access being denied.

Also, users previously authenticating with passwords sent as cleartext SHOULD be required to change those passwords when migrating to TLS, if the old passwords were likely to have been compromised. (For any large community of users using the public Internet to access mail without encryption, the compromise of at least some of those passwords should be assumed.)

Transition of users from SSL or TLS 1.0 to later versions of TLS MAY be accomplished by a means similar to that described above. There are multiple ways to accomplish this. One way is for the server to refuse a ClientHello message from any client sending a ClientHello.version field corresponding to any version of SSL or TLS 1.0. Another way is for the server to accept ClientHello messages from some client versions that it does not wish to support but later refuse to allow the user to authenticate. The latter method may provide a better indication to the user of the reason for the failure but (depending on the protocol and method of authentication used) may also risk exposure of the user's password over a channel that is known to not provide adequate confidentiality.

It is RECOMMENDED that new users be required to use TLS version 1.1 or greater from the start. However, an MSP may find it necessary to make exceptions to accommodate some legacy systems that support only earlier versions of TLS or only cleartext.

4.2. Mail Server Use of Client Certificate Authentication

Mail Submission Servers and Mail Access Servers MAY implement client certificate authentication on the Implicit TLS port. Such servers MUST NOT request a client certificate during the TLS handshake unless the server is configured to accept some client certificates as sufficient for authentication and the server has the ability to determine a mail server authorization identity matching such certificates. How to make this determination is presently implementation specific.

If the server accepts the client's certificate as sufficient for authorization, it MUST enable the Simple Authentication and Security Layer (SASL) EXTERNAL mechanism [RFC4422]. An IMAPS server MAY issue a PREAUTH greeting instead of enabling SASL EXTERNAL.

4.3. Recording TLS Ciphersuite in "Received" Header Field

The ESMTPS transmission type [RFC3848] provides trace information that can indicate that TLS was used when transferring mail. However, TLS usage by itself is not a guarantee of confidentiality or security. The TLS ciphersuite provides additional information about the level of security made available for a connection. This section

defines a new SMTP "tls" Received header additional-registered-clause that is used to record the TLS ciphersuite that was negotiated for the connection. This clause SHOULD be included whenever a Submission server generates a Received header field for a message received via TLS. The value included in this additional clause SHOULD be the registered ciphersuite name (e.g., TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256) included in the "TLS Cipher Suite Registry". In the event that the implementation does not know the name of the ciphersuite (a situation that should be remedied promptly), a four-digit hexadecimal ciphersuite identifier MAY be used. In addition, the Diffie-Hellman group name associated with the ciphersuite MAY be included (when applicable and known) following the ciphersuite name. The ABNF for the field follows:

```
tls-cipher-clause  =  CFWS "tls" FWS tls-cipher
                      [ CFWS tls-dh-group-clause ]

tls-cipher         =  tls-cipher-name / tls-cipher-hex

tls-cipher-name    =  ALPHA *(ALPHA / DIGIT / "_")
; as registered in the IANA "TLS Cipher Suite Registry"
; <https://www.iana.org/assignments/tls-parameters>

tls-cipher-hex     =  "0x" 4HEXDIG

tls-dh-group-clause = "group" FWS dh-group
; not to be used except immediately after tls-cipher

dh-group           = ALPHA *(ALPHA / DIGIT / "_" / "-")
; as registered in the IANA "TLS Supported Groups Registry"
; <https://www.iana.org/assignments/tls-parameters>
```

4.4. TLS Server Certificate Requirements

MSPs MUST maintain valid server certificates for all servers. See [RFC7817] for the recommendations and requirements necessary to achieve this.

If a protocol server provides service for more than one mail domain, it MAY use a separate IP address for each domain and/or a server certificate that advertises multiple domains. This will generally be necessary unless and until it is acceptable to impose the constraint that the server and all clients support the Server Name Indication (SNI) extension to TLS [RFC6066]. Mail servers supporting the SNI need to support the post-SRV hostname to interoperate with MUAs that have not implemented [RFC6186]. For more discussion of this problem, see Section 5.1 of [RFC7817].

4.5. Recommended DNS Records for Mail Protocol Servers

This section discusses not only the DNS records that are recommended but also implications of DNS records for server configuration and TLS server certificates.

4.5.1. MX Records

It is recommended that MSPs advertise MX records for the handling of inbound mail (instead of relying entirely on A or AAAA records) and that those MX records be signed using DNSSEC [RFC4033]. This is mentioned here only for completeness, as the handling of inbound mail is out of scope for this document.

4.5.2. SRV Records

MSPs SHOULD advertise SRV records to aid MUAs in determining the proper configuration of servers, per the instructions in [RFC6186].

MSPs SHOULD advertise servers that support Implicit TLS in preference to servers that support cleartext and/or STARTTLS operation.

4.5.3. DNSSEC

All DNS records advertised by an MSP as a means of aiding clients in communicating with the MSP’s servers SHOULD be signed using DNSSEC if and when the parent DNS zone supports doing so.

4.5.4. TLSA Records

MSPs SHOULD advertise TLSA records to provide an additional trust anchor for public keys used in TLS server certificates. However, TLSA records MUST NOT be advertised unless they are signed using DNSSEC.

4.6. Changes to Internet-Facing Servers

When an MSP changes the Internet-facing Mail Access Servers and Mail Submission Servers, including SMTP-based spam/virus filters, it is generally necessary to support the same and/or a newer version of TLS than the one previously used.

5. Use of TLS by Mail User Agents

The following requirements and recommendations apply to MUAs:

- MUAs SHOULD be capable of using DNS SRV records to discover Mail Access Servers and Mail Submission Servers that are advertised by an MSP for an account being configured. Other means of discovering server configuration information (e.g., a database maintained by the MUA vendor) MAY also be supported. (See Section 5.1 for more information.)

- MUAs SHOULD be configurable to require a minimum level of confidentiality for any particular Mail Account and refuse to exchange information via any service associated with that Mail Account if the session does not provide that minimum level of confidentiality. (See Section 5.2.)

- MUAs MUST NOT treat a session as meeting a minimum level of confidentiality if the server’s TLS certificate cannot be validated. (See Section 5.3.)

- MUAs MAY impose other minimum confidentiality requirements in the future, e.g., in order to discourage the use of TLS versions or cryptographic algorithms in which weaknesses have been discovered.

- MUAs SHOULD provide a prominent indication of the level of confidentiality associated with an account configuration that is appropriate for the user interface (for example, a "lock" icon or changed background color for a visual interface, or some sort of audible indication for an audio user interface), at appropriate times and/or locations, in order to inform the user of the confidentiality of the communications associated with that account. For example, this might be done whenever (a) the user is prompted for authentication credentials, (b) the user is composing mail that will be sent to a particular submission server, (c) a list of accounts is displayed (particularly if the user can select from that list to read mail), or (d) the user is asking to view or update any configuration data that will be stored on a remote server. If, however, an MUA provides such an indication, it MUST NOT indicate confidentiality for any connection that does not at least use TLS 1.1 with certificate verification and also meet the minimum confidentiality requirements associated with that account.

- MUAs MUST implement TLS 1.2 [RFC5246] or later. Earlier TLS and SSL versions MAY also be supported, so long as the MUA requires at least TLS 1.1 [RFC4346] when accessing accounts that are configured to impose minimum confidentiality requirements.

- All MUAs SHOULD implement the recommended TLS ciphersuites described in [RFC7525] or a future BCP or Standards Track revision of that document.

- MUAs that are configured to not require minimum confidentiality for one or more accounts SHOULD detect when TLS becomes available on those accounts (using [RFC6186] or other means) and offer to upgrade the account to require TLS.

Additional considerations and details appear below.

5.1. Use of SRV Records in Establishing Configuration

This document updates [RFC6186] by changing the preference rules and adding a new SRV service label _submissions._tcp to refer to Message Submission with Implicit TLS.

User-configurable MUAs SHOULD support the use of [RFC6186] for account setup. However, when using configuration information obtained via this method, MUAs SHOULD ignore advertised services that do not satisfy minimum confidentiality requirements, unless the user has explicitly requested reduced confidentiality. This will have the effect of causing the MUA to default to ignoring advertised configurations that do not support TLS, even when those advertised configurations have a higher priority than other advertised configurations.

When using configuration information per [RFC6186], MUAs SHOULD NOT automatically establish new configurations that do not require TLS for all servers, unless there are no advertised configurations using TLS. If such a configuration is chosen, prior to attempting to authenticate to the server or use the server for Message Submission, the MUA SHOULD warn the user that traffic to that server will not be encrypted and that it will therefore likely be intercepted by unauthorized parties. The specific wording is to be determined by the implementation, but it should adequately capture the sense of risk, given the widespread incidence of mass surveillance of email traffic.

Similarly, an MUA MUST NOT attempt to "test" a particular Mail Account configuration by submitting the user’s authentication credentials to a server, unless a TLS session meeting minimum confidentiality levels has been established with that server. If minimum confidentiality requirements have not been satisfied, the MUA must explicitly warn that the user’s password may be exposed to attackers before testing the new configuration.

When establishing a new configuration for connecting to an IMAP, POP, or SMTP submission server, based on SRV records, an MUA SHOULD verify that either (a) the SRV records are signed using DNSSEC or (b) the target Fully Qualified Domain Name (FQDN) of the SRV record matches the original server FQDN for which the SRV queries were made. If the target FQDN is not in the queried domain, the MUA SHOULD verify with the user that the SRV target FQDN is suitable for use, before executing any connections to the host. (See Section 6 of [RFC6186].)

An MUA MUST NOT consult SRV records to determine which servers to use on every connection attempt, unless those SRV records are signed by DNSSEC and have a valid signature. However, an MUA MAY consult SRV records from time to time to determine if an MSP's server configuration has changed and alert the user if it appears that this has happened. This can also serve as a means to encourage users to upgrade their configurations to require TLS if and when their MSPs support it.

5.2. Minimum Confidentiality Level

MUAs SHOULD, by default, require a minimum level of confidentiality for services accessed by each account. For MUAs supporting the ability to access multiple Mail Accounts, this requirement SHOULD be configurable on a per-account basis.

The default minimum expected level of confidentiality for all new accounts MUST require successful validation of the server's certificate and SHOULD require negotiation of TLS version 1.1 or greater. (Future revisions to this specification may raise these requirements or impose additional requirements to address newly discovered weaknesses in protocols or cryptographic algorithms.)

MUAs MAY permit the user to disable this minimum confidentiality requirement during initial account configuration or when subsequently editing an account configuration but MUST warn users that such a configuration will not assure privacy for either passwords or messages.

An MUA that is configured to require a minimum level of confidentiality for a Mail Account MUST NOT attempt to perform any operation other than capability discovery, or STARTTLS for servers not using Implicit TLS, unless the minimum level of confidentiality is provided by that connection.

MUAs SHOULD NOT allow users to easily access or send mail via a connection, or authenticate to any service using a password, if that account is configured to impose minimum confidentiality requirements and that connection does not meet all of those requirements. An

example of "easy access" would be to display a dialog informing the user that the security requirements of the account were not met by the connection but allowing the user to "click through" to send mail or access the service anyway. Experience indicates that users presented with such an option often "click through" without understanding the risks that they're accepting by doing so. Furthermore, users who frequently find the need to "click through" to use an insecure connection may become conditioned to do so as a matter of habit, before considering whether the risks are reasonable in each specific instance.

An MUA that is not configured to require a minimum level of confidentiality for a Mail Account SHOULD still attempt to connect to the services associated with that account using the most secure means available, e.g., by using Implicit TLS or STARTTLS.

5.3. Certificate Validation

MUAs MUST validate TLS server certificates according to [RFC7817] and PKIX [RFC5280].

MUAs MAY also support DNS-Based Authentication of Named Entities (DANE) [RFC6698] as a means of validating server certificates in order to meet minimum confidentiality requirements.

MUAs MAY support the use of certificate pinning but MUST NOT consider a connection in which the server's authenticity relies on certificate pinning as providing the minimum level of confidentiality. (See Section 5.4.)

5.4. Certificate Pinning

During account setup, the MUA will identify servers that provide account services such as mail access and mail submission (Section 5.1 describes one way to do this). The certificates for these servers are verified using the rules described in [RFC7817] and PKIX [RFC5280]. In the event that the certificate does not validate due to an expired certificate, a lack of an appropriate chain of trust, or a lack of an identifier match, the MUA MAY offer to create a persistent binding between that certificate and the saved hostname for the server, for use when accessing that account's servers. This is called "certificate pinning".

(Note: This use of the term "certificate pinning" means something subtly different than HTTP Public Key Pinning as described in [RFC7469]. The dual use of the same term is confusing, but unfortunately both uses are well established.)

Certificate pinning is only appropriate during Mail Account setup and MUST NOT be offered as an option in response to a failed certificate validation for an existing Mail Account. An MUA that allows certificate pinning MUST NOT allow a certificate pinned for one account to validate connections for other accounts. An MUA that allows certificate pinning MUST also allow a user to undo the pinning, i.e., to revoke trust in a certificate that has previously been pinned.

A pinned certificate is subject to a man-in-the-middle attack at account setup time and typically lacks a mechanism to automatically revoke or securely refresh the certificate. Note also that a man-in-the-middle attack at account setup time will expose the user's password to the attacker (if a password is used). Therefore, the use of a pinned certificate does not meet the requirement for a minimum confidentiality level, and an MUA MUST NOT indicate to the user that such confidentiality is provided. Additional advice on certificate pinning is presented in [RFC6125].

5.5. Client Certificate Authentication

MUAs MAY implement client certificate authentication on the Implicit TLS port. An MUA MUST NOT provide a client certificate during the TLS handshake unless the server requests one and the MUA has been authorized to use that client certificate with that account. Having the end user explicitly configure a client certificate for use with a given account is sufficient to meet this requirement. However, installing a client certificate for use with one account MUST NOT automatically authorize the use of that certificate with other accounts. This is not intended to prohibit site-specific authorization mechanisms, such as (a) a site-administrator-controlled mechanism to authorize the use of a client certificate with a given account or (b) a domain-name-matching mechanism.

Note: The requirement that the server request a certificate is just a restatement of the TLS protocol rules, e.g., Section 7.4.6 of [RFC5246]. The requirement that the client not send a certificate not known to be acceptable to the server is pragmatic in multiple ways: the current TLS protocol provides no way for the client to know which of the potentially multiple certificates it should use; also, when the client sends a certificate, it is potentially disclosing its identity (or its user's identity) to both the server and any party with access to the transmission medium, perhaps unnecessarily and for no useful purpose.

A client supporting client certificate authentication with Implicit TLS MUST implement the SASL EXTERNAL mechanism [RFC4422], using the appropriate authentication command (AUTH for POP3 [RFC5034], AUTH for SMTP Submission [RFC4954], or AUTHENTICATE for IMAP [RFC3501]).

6. Considerations Related to Antivirus/Antispam Software and Services

There are multiple ways to connect an AVAS service (e.g., "Antivirus & Antispam") to a mail server. Some mechanisms, such as the de facto "milter" protocol, are out of scope for this specification. However, some services use an SMTP relay proxy that intercepts mail at the application layer to perform a scan and proxy or forward to another Mail Transfer Agent (MTA). Deploying AVAS services in this way can cause many problems [RFC2979], including direct interference with this specification, and other forms of confidentiality or security reduction. An AVAS product or service is considered compatible with this specification if all IMAP, POP, and SMTP-related software (including proxies) it includes are compliant with this specification.

Note that end-to-end email encryption prevents AVAS software and services from using email content as part of a spam or virus assessment. Furthermore, although a minimum confidentiality level can prevent a man-in-the-middle from introducing spam or virus content between the MUA and Submission server, it does not prevent other forms of client or account compromise. The use of AVAS services for submitted email therefore remains necessary.

7. IANA Considerations

7.1. POP3S Port Registration Update

IANA has updated the registration of the TCP well-known port 995 using the following template [RFC6335]:

```
Service Name: pop3s
Transport Protocol: TCP
Assignee: IESG <iesg@ietf.org>
Contact: IETF Chair <chair@ietf.org>
Description: POP3 over TLS protocol
Reference: RFC 8314
Port Number: 995
```

7.2. IMAPS Port Registration Update

IANA has updated the registration of the TCP well-known port 993 using the following template [RFC6335]:

```
Service Name: imaps
Transport Protocol: TCP
Assignee: IESG <iesg@ietf.org>
Contact: IETF Chair <chair@ietf.org>
Description: IMAP over TLS protocol
Reference: RFC 8314
Port Number: 993
```

No changes to existing UDP port assignments for pop3s or imaps are being requested.

7.3. Submissions Port Registration

IANA has assigned an alternate usage of TCP port 465 in addition to the current assignment using the following template [RFC6335]:

```
Service Name: submissions
Transport Protocol: TCP
Assignee: IESG <iesg@ietf.org>
Contact: IETF Chair <chair@ietf.org>
Description: Message Submission over TLS protocol
Reference: RFC 8314
Port Number: 465
```

This is a one-time procedural exception to the rules in [RFC6335]. This requires explicit IESG approval and does not set a precedent. Note: Since the purpose of this alternate usage assignment is to align with widespread existing practice and there is no known usage of UDP port 465 for Message Submission over TLS, IANA has not assigned an alternate usage of UDP port 465.

Historically, port 465 was briefly registered as the "smtps" port. This registration made no sense, as the SMTP transport MX infrastructure has no way to specify a port, so port 25 is always used. As a result, the registration was revoked and was subsequently reassigned to a different service. In hindsight, the "smtps" registration should have been renamed or reserved rather than revoked. Unfortunately, some widely deployed mail software interpreted "smtps" as "submissions" [RFC6409] and used that port for email submission by default when an end user requested security during account setup. If a new port is assigned for the submissions service, either (a) email software will continue with unregistered use of port 465 (leaving the port registry inaccurate relative to

de facto practice and wasting a well-known port) or (b) confusion between the de facto and registered ports will cause harmful interoperability problems that will deter the use of TLS for Message Submission. The authors of this document believe that both of these outcomes are less desirable than a "wart" in the registry documenting real-world usage of a port for two purposes. Although STARTTLS on port 587 has been deployed, it has not replaced the deployed use of Implicit TLS submission on port 465.

7.4. Additional Registered Clauses for "Received" Fields

Per the provisions in [RFC5321], IANA has added two additional-registered-clauses for Received fields as defined in Section 4.3 of this document:

- "tls": Indicates the TLS cipher used (if applicable)
- "group": Indicates the Diffie-Hellman group used with the TLS cipher (if applicable)

The descriptions and syntax of these additional clauses are provided in Section 4.3 of this document.

8. Security Considerations

This entire document is about security considerations. In general, this is targeted to improve mail confidentiality and to mitigate threats external to the email system such as network-level snooping or interception; this is not intended to mitigate active attackers who have compromised service provider systems.

Implementers should be aware that the use of client certificates with TLS 1.2 reveals the user’s identity to any party with the ability to read packets from the transmission medium and therefore may compromise the user’s privacy. There seems to be no easy fix with TLS 1.2 or earlier versions, other than to avoid presenting client certificates except when there is explicit authorization to do so. TLS 1.3 [TLS-1.3] appears to reduce this privacy risk somewhat.

9. References

9.1. Normative References

[RFC793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>.

[RFC1939] Myers, J. and M. Rose, "Post Office Protocol - Version 3", STD 53, RFC 1939, DOI 10.17487/RFC1939, May 1996, <https://www.rfc-editor.org/info/rfc1939>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, DOI 10.17487/RFC3207, February 2002, <https://www.rfc-editor.org/info/rfc3207>.

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, DOI 10.17487/RFC3501, March 2003, <https://www.rfc-editor.org/info/rfc3501>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC5034] Siemborski, R. and A. Menon-Sen, "The Post Office Protocol (POP3) Simple Authentication and Security Layer (SASL) Authentication Mechanism", RFC 5034, DOI 10.17487/RFC5034, July 2007, <https://www.rfc-editor.org/info/rfc5034>.

[RFC5234] Crocker, D., Ed., and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <https://www.rfc-editor.org/info/rfc5322>.

[RFC6186] Daboo, C., "Use of SRV Records for Locating Email Submission/Access Services", RFC 6186, DOI 10.17487/RFC6186, March 2011, <https://www.rfc-editor.org/info/rfc6186>.

[RFC6409] Gellens, R. and J. Klensin, "Message Submission for Mail", STD 72, RFC 6409, DOI 10.17487/RFC6409, November 2011, <https://www.rfc-editor.org/info/rfc6409>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <https://www.rfc-editor.org/info/rfc7525>.

[RFC7672] Dukhovni, V. and W. Hardaker, "SMTP Security via Opportunistic DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS)", RFC 7672, DOI 10.17487/RFC7672, October 2015, <https://www.rfc-editor.org/info/rfc7672>.

[RFC7817] Melnikov, A., "Updated Transport Layer Security (TLS) Server Identity Check Procedure for Email-Related Protocols", RFC 7817, DOI 10.17487/RFC7817, March 2016, <https://www.rfc-editor.org/info/rfc7817>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

9.2. Informative References

[CERT-555316]
CERT, "Vulnerability Note VU#555316: STARTTLS plaintext command injection vulnerability", Carnegie Mellon University Software Engineering Institute, September 2011, <https://www.kb.cert.org/vuls/id/555316>.

[Email-TLS]
Moore, K., "Recommendations for use of TLS by Electronic Mail Access Protocols", Work in Progress, draft-moore-email-tls-00, October 2013.

[MTA-STS] Margolis, D., Risher, M., Ramakrishnan, B., Brotman, A., and J. Jones, "SMTP MTA Strict Transport Security (MTA-STS)", Work in Progress, draft-ietf-uta-mta-sts-14, January 2018.

[POP3-over-TLS]
Melnikov, A., Newman, C., and M. Yevstifeyev, Ed., "POP3 over TLS", Work in Progress, draft-melnikov-pop3-over-tls-02, August 2011.

[RFC2595] Newman, C., "Using TLS with IMAP, POP3 and ACAP", RFC 2595, DOI 10.17487/RFC2595, June 1999, <https://www.rfc-editor.org/info/rfc2595>.

[RFC2979] Freed, N., "Behavior of and Requirements for Internet Firewalls", RFC 2979, DOI 10.17487/RFC2979, October 2000, <https://www.rfc-editor.org/info/rfc2979>.

[RFC3848] Newman, C., "ESMTP and LMTP Transmission Types Registration", RFC 3848, DOI 10.17487/RFC3848, July 2004, <https://www.rfc-editor.org/info/rfc3848>.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, DOI 10.17487/RFC4346, April 2006, <https://www.rfc-editor.org/info/rfc4346>.

[RFC4422] Melnikov, A., Ed., and K. Zeilenga, Ed., "Simple Authentication and Security Layer (SASL)", RFC 4422, DOI 10.17487/RFC4422, June 2006, <https://www.rfc-editor.org/info/rfc4422>.

[RFC4954] Siemborski, R., Ed., and A. Melnikov, Ed., "SMTP Service Extension for Authentication", RFC 4954, DOI 10.17487/RFC4954, July 2007, <https://www.rfc-editor.org/info/rfc4954>.

[RFC5068] Hutzler, C., Crocker, D., Resnick, P., Allman, E., and T. Finch, "Email Submission Operations: Access and Accountability Requirements", BCP 134, RFC 5068, DOI 10.17487/RFC5068, November 2007, <https://www.rfc-editor.org/info/rfc5068>.

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, DOI 10.17487/RFC5321, October 2008, <https://www.rfc-editor.org/info/rfc5321>.

[RFC6066] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, DOI 10.17487/RFC6066, January 2011, <https://www.rfc-editor.org/info/rfc6066>.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <https://www.rfc-editor.org/info/rfc6125>.

[RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", BCP 165, RFC 6335, DOI 10.17487/RFC6335, August 2011, <https://www.rfc-editor.org/info/rfc6335>.

[RFC7469] Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", RFC 7469, DOI 10.17487/RFC7469, April 2015, <https://www.rfc-editor.org/info/rfc7469>.

[TLS-1.3] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", Work in Progress, draft-ietf-tls-tls13-23, January 2018.

Appendix A. Design Considerations

This section is not normative.

The first version of this document was written independently from the October 2013 version of [Email-TLS] ("Recommendations for use of TLS by Electronic Mail Access Protocols"). Subsequent versions merge ideas from both documents.

One author of this document was also the author of RFC 2595, which became the standard for TLS usage with POP and IMAP, and the other author was perhaps the first to propose that idea. In hindsight, both authors now believe that that approach was a mistake. At this point, the authors believe that while anything that makes it easier to deploy TLS is good, the desirable end state is that these protocols always use TLS, leaving no need for a separate port for cleartext operation except to support legacy clients while they continue to be used. The separate-port model for TLS is inherently simpler to implement, debug, and deploy. It also enables a "generic TLS load-balancer" that accepts secure client connections for arbitrary foo-over-TLS protocols and forwards them to a server that may or may not support TLS. Such load-balancers cause many problems because they violate the end-to-end principle and the server loses the ability to log security-relevant information about the client unless the protocol is designed to forward that information (as this specification does for the ciphersuite). However, they can result in TLS deployment where it would not otherwise happen, which is a sufficiently important goal that it overrides any problems.

Although STARTTLS appears only slightly more complex than separate-port TLS, we again learned the lesson that complexity is the enemy of security in the form of the STARTTLS command injection vulnerability (Computer Emergency Readiness Team (CERT) vulnerability ID #555316 [CERT-555316]). Although there’s nothing inherently wrong with STARTTLS, the fact that it resulted in a common implementation error (made independently by multiple implementers) suggests that it is a less secure architecture than Implicit TLS.

Section 7 of RFC 2595 critiques the separate-port approach to TLS. The first bullet was a correct critique. There are proposals in the HTTP community to address that, and the use of SRV records as described in RFC 6186 resolves that critique for email. The second bullet is correct as well but is not very important because useful deployment of security layers other than TLS in email is small enough to be effectively irrelevant. (Also, it’s less correct than it used to be because "export" ciphersuites are no longer supported in modern versions of TLS.) The third bullet is incorrect because it misses the desirable option of "use TLS for all subsequent connections to

this server once TLS is successfully negotiated". The fourth bullet may be correct, but it is not a problem yet with current port consumption rates. The fundamental error was prioritizing a perceived better design based on a mostly valid critique over real-world deployability. But getting security and confidentiality facilities actually deployed is so important that it should trump design purity considerations.

Port 465 is presently used for two purposes: for submissions by a large number of clients and service providers and for the "urd" protocol by one vendor. Actually documenting this current state is controversial, as discussed in the IANA Considerations section. However, there is no good alternative. Registering a new port for submissions when port 465 is already widely used for that purpose will just create interoperability problems. Registering a port that’s only used if advertised by an SRV record (RFC 6186) would not create interoperability problems but would require all client deployments, server deployments, and software to change significantly, which is contrary to the goal of promoting the increased use of TLS. Encouraging the use of STARTTLS on port 587 would not create interoperability problems, but it is unlikely to have any impact on the current undocumented use of port 465 and makes the guidance in this document less consistent. The remaining option is to document the current state of the world and support future use of port 465 for submission, as this increases consistency and ease of deployment for TLS email submission.

Acknowledgements

Thanks to Ned Freed for discussion of the initial concepts in this document. Thanks to Alexey Melnikov for [POP3-over-TLS], which was the basis of the POP3 Implicit TLS text. Thanks to Russ Housley, Alexey Melnikov, and Dan Newman for review feedback. Thanks to Paul Hoffman for interesting feedback in initial conversations about this idea.

Authors' Addresses

Keith Moore
Windrock, Inc.
PO Box 1934
Knoxville, TN 37901
United States of America

Email: moore@network-heretics.com

Chris Newman
Oracle
440 E. Huntington Dr., Suite 400
Arcadia, CA 91006
United States of America

Email: chris.newman@oracle.com