         Cleartext Considered Obsolete: Use of TLS for Email Submission and
                                Access
                       draft-ietf-uta-email-deep-10

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 or mail
   access server.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on April 29, 2018.

Copyright Notice

Internet-Draft   Use of TLS for Email Submission/Access   October 2017

      the Trust Legal Provisions and are provided without warranty as
      described in the Simplified BSD License.

Table of Contents

1. Introduction

Software that provides email service via Internet Message Access
Protocol (IMAP) [RFC3501], 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 and Mail Access Services, and
between Mail User Agents and Mail Submission Services.

In brief, this memo now recommends that:

o TLS version 1.2 or greater be used for all traffic between mail
   user agents (MUAs) and mail submission servers, and also between
   MUAs and mail access servers.

o MUAs and mail service providers discourage use of cleartext
   protocols for mail access and mail submission, and deprecate use
   of cleartext protocols for these purposes as soon as practicable.

o Use of "Implicit TLS" on ports reserved for that purpose, in
   preference to STARTTLS on a port that otherwise supports
   cleartext.

This memo does not address use of TLS with SMTP for message relay
(where Message Submission [RFC6409] does not apply). Improved use of
TLS with SMTP for message relay requires a different approach. One
approach to address that topic is described in [RFC7672]; another is
in [I-D.ietf-uta-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.

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 use of STARTTLS and
similar commands in POP, IMAP, SMTP message submission, and other

protocols, that are used by client and server to explicitly negotiate
TLS on an established cleartext TCP connection.

The term "Mail Access Services" includes POP, IMAP and any other
protocol used to access or modify received messages, or to access or
modify a mail user's account configuration.

"Mail Submission Service" refers to the use of 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" (MSP) refers to a provider of Mail
Access Services and/or Mail Submission Services.

The term "Mail Account" refers to a user's identity with a Mail
Service Provider, 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, spam filtering
instructions). Most Mail User Agents (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 in
Appendix B and rules from [RFC5322].

3. Implicit TLS

Previous standards for 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. While 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 encourage a greater consistency for how TLS is used, this
specification now recommends use of Implicit TLS for POP, IMAP, SMTP
Submission, and all other protocols used between a Mail User Agent
and a mail service.

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 a +OK
   greeting, the server and client MUST enter AUTHORIZATION state, even
   if a client certificate was supplied during the TLS handshake.

   See Section 5.5 and Section 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 client enter AUTHENTICATED state. If the server issues an OK
   greeting then both server and client enter NOT AUTHENTICATED state.

   See Section 5.5 and Section 4.2 for additional information on client
   certificate authentication. See Section 7.1 and 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 a 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 10.3 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 the additional reasons discussed in
Appendix A. However, to maximize 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 both client and server are configured to require
successful negotiation of TLS prior to message submission.

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

See Section 5.5 and Section 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 ([RFC0793] section
3.5) without waiting for a TLS response from the server.

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

The following requirements and recommendations apply to Mail Access
Services and Mail Submission Services:

o Mail Service Providers (MSPs) that support POP, IMAP, and/or
   Message Submission, MUST support TLS access for those services.

o Other services than POP, IMAP and/or Message Submission provided
   by MSPs SHOULD support TLS access, and MUST support TLS access for
   those services which support authentication via username and
   password.

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

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

o MSPs SHOULD advertise their Mail Access Services and Mail
   Submission Services 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.) Services supporting TLS SHOULD be
   advertised in preference to cleartext services (if offered). In
   addition, services using Implicit TLS SHOULD be advertised in
   preference to services supporting STARTTLS (if offered). (See
   also Section 4.5.)

o MSPs SHOULD deprecate use of cleartext Mail Access Services and
   Mail Submission Services as soon as practicable. (See
   Section 4.1.)

o 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.

o Mail services MUST support TLS 1.2 or later.

o All Mail services SHOULD implement the recommended TLS cipher
   suites described in [RFC7525] or a future BCP or standards track
   revision of that document.

o Mail services currently supporting SSL 2.x, SSL 3.0, or TLS 1.0
   SHOULD transition their users to later versions of TLS, and
   discontinue support for those versions of SSL and TLS, as soon as
   practicable.

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

o All Mail services 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 < 1.1

   The specific means employed for deprecation of cleartext Mail Access
   Services and Mail Submission Services 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 services, thus encouraging those users to migrate
   to Implicit TLS. Access to cleartext services should eventually be
   either disabled, or limited strictly for use by legacy systems which
   cannot be upgraded.

   After a user's ability to authenticate to a service 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 public Internet to access
   mail without encryption, 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 an channel which 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 which support only
   earlier versions of TLS, or only cleartext.

4.2. Mail Server Use of Client Certificate Authentication

   Mail servers MAY implement client certificate authentication on the
   Implicit TLS port. 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 SASL EXTERNAL [RFC4422] mechanism.
   An IMAPS server MAY issue a PREAUTH greeting instead of enabling SASL
   EXTERNAL.

4.3. Recording TLS Cipher Suite in Received Header

   The ESMTPS transmission type [RFC3848] provides trace information
   that can indicate TLS was used when transferring mail. However, TLS
   usage by itself is not a guarantee of confidentiality or security.
   The TLS cipher suite provides additional information about the level
   of security made available for a connection. This defines a new SMTP
   "tls" Received header additional-registered-clause that is used to
   record the TLS cipher suite 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 cipher
   suite name (e.g., TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256) included in
   the TLS cipher suite registry. In the event the implementation does
   not know the name of the cipher suite (a situation that should be
   remedied promptly), a four-digit hexadecimal cipher suite 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 [ FWS "group" FWS dh-group ]

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

tls-cipher-name    =  ALPHA *(ALPHA / DIGIT / "_")
; as registered in IANA cipher suite registry

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

dh-group           = ALPHA *(ALPHA / DIGIT / "_" / "-")
; as registered in IANA TLS Supported Groups Registry
```

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
   extension to TLS [RFC6066]. 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 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 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 determination of
   proper configuration of servers, per the instructions in [RFC6186].

   MSPs SHOULD advertise servers that support Implicit TLS in preference
   to those which 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 Servers providing mail access
   and mail submission services, including SMTP-based spam/virus
   filters, it is generally necessary to support the same and/or a newer
   version of TLS and the same security directives that were previously
   advertised.

5. Use of TLS by Mail User Agents

   The following requirements and recommendations apply to Mail User
   Agents:

   o MUAs SHOULD be capable of using DNS SRV records to discover Mail
      Access Services and Mail Submission Services that are advertised
      by a 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.)

   o 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.)

   o 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.)

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

   o 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) prompting
      the user 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 requesting 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.

o 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.

o All MUAs SHOULD implement the recommended TLS cipher suites
   described in [RFC7525] or a future BCP or standards track revision
   of that document.

o 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 section 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 use of [RFC6186] for account
setup. However, when using configuration information obtained by
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 [RFC6186] configuration information, Mail User Agents
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, a 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 the user that his 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 either verify that the SRV records are verifiability signed using DNSSEC, or that the target 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 [RFC6186] section 6).

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 subsequently editing an account configuration, but MUST warn users that such a configuration will not assure privacy for either passwords or messages.

An MUA which 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 an
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 "easily 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 which 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. Certificiate Validation

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

MUAs MAY also support DANE [RFC6698] as a means of validating server
certificates in order to meet minimum confidentiality requirements.

MUAs MAY support 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 (the
previous section 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 the certificate does not validate due
to an expired certificate, lack of appropriate chain of trust, or
lack of identifier match, the MUA MAY offer to create a persistent
binding between that certificate and the saved host name 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" [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 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
the such confidentiality is provided. Additional advice on
certificate pinning is present 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 use of that certificate with other accounts.
This is not intended to prohibit site-specific authorization
mechanisms, such as a site-administrator-controlled mechanism to
authorize use of a client certificate with a given account, or 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. [RFC5246] section 7.4.6.
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
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 to 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 [RFC4422] mechanism using the
   appropriate authentication command (AUTH for POP3 [RFC5034], AUTH for
   SMTP Submission [RFC4954], AUTHENTICATE for IMAP [RFC3501]).

6. Considerations related to Anti-Virus/Anti-Spam Software and Services

   There are multiple ways to connect an Anti-Virus and/or Anti-Spam
   (AVAS) service 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 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, while 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. Use of AVAS services for
   submitted email therefore remains necessary.

7. IANA Considerations

7.1. POP3S Port Registration Update

   IANA is asked to update the registration of the TCP well-known port
   995, and also UDP well-known port 995, using the following templates
   ([RFC6335]):

     Service Name: pop3s
     Transport Protocol: TCP
     Assignee: IETF <iesg@ietf.org>
     Contact: IESG <iesg@ietf.org>
     Description: POP3 over TLS protocol
     Reference: RFC XXXX (this document once published)
     Port Number: 995

```
Service Name: pop3s
Transport Protocol: UDP
Assignee: IETF <iesg@ietf.org>
Contact: IESG <iesg@ietf.org>
Description: POP3 over TLS protocol
Reference: RFC XXXX (this document once published)
Port Number: 995
```

7.2. IMAPS Port Registration Update

IANA is asked to update the registration of the TCP well-known port 993, and also UDP well-known port 993, using the following templates ([RFC6335]):

```
Service Name: imaps
Transport Protocol: TCP
Assignee: IETF <iesg@ietf.org>
Contact: IESG <iesg@ietf.org>
Description: IMAP over TLS protocol
Reference: RFC XXXX (this document once published)
Port Number: 993
```

```
Service Name: imaps
Transport Protocol: UDP
Assignee: IETF <iesg@ietf.org>
Contact: IESG <iesg@ietf.org>
Description: IMAP over TLS protocol
Reference: RFC XXXX (this document once published)
Port Number: 993
```

Note: the updates to the UDP port assignments preserve the pre-existing alignment in which both TCP and UDP ports were allocated to the same protocols (in this case POP3+TLS and IMAP4+TLS) even though there is no specification for using either protocol over UDP. This seems undesirable as it wastes UDP port assignments. However, it is not within the scope of this document to revisit old conventions for port assignments.

7.3. Submissions Port Registration

IANA is asked to assign 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: IETF <iesg@ietf.org>
      Contact: IESG <iesg@ietf.org>
      Description: Message Submission over TLS protocol
      Reference: RFC XXXX (this document once published)
      Port Number: 465

   This is a one-time procedural exception to the rules in RFC 6335.
   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 is not being
   asked to assign 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 requests security during
   account setup. If a new port is assigned for the submissions
   service, email software will either continue with unregistered use of
   port 465 (leaving the port registry inaccurate relative to de-facto
   practice and wasting a well-known port), or confusion between the de-
   facto and registered ports will cause harmful interoperability
   problems that will deter use of TLS for message submission. The
   authors believe 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 deployed, it has not
   replaced deployed use of Implicit TLS submission on port 465.

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 use of client certificates with TLS
   1.2 reveals the user's identity to any party with 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
[I-D.ietf-tls-tls13] appears to reduce the privacy risk somewhat.

9. References

9.1. Normative References

[RFC0793] 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>.

[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>.

[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>.

9.2. Informative References

   [I-D.ietf-tls-tls13]
              Rescorla, E., "The Transport Layer Security (TLS) Protocol
              Version 1.3", draft-ietf-tls-tls13-21 (work in progress),
              July 2017.

   [I-D.ietf-uta-mta-sts]
              Margolis, D., Risher, M., Ramakrishnan, B., Brotman, A.,
              and J. Jones, "SMTP MTA Strict Transport Security (MTA-
              STS)", draft-ietf-uta-mta-sts-09 (work in progress),
              September 2017.

   [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>.

   [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>.

Appendix A. Design Considerations

   This section is not normative.

   The first version of this was written independently from draft-moore-
   email-tls-00.txt; subsequent versions merge ideas from both drafts.

   One author of this document was also the author of RFC 2595 that
   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 cipher suite). However, they can result
   in TLS deployment where it would not otherwise happen which is a
   sufficiently important goal that it overrides the 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

(CERT vulnerability ID #555316). Although there's nothing inherently wrong with STARTTLS, the fact it resulted in a common implementation error (made independently by multiple implementers) suggests 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 use of SRV records as described in RFC 6186 resolves that critique for email. The second bullet is correct as well, but 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 and latch-on TLS if available". The fourth bullet may be correct, but 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 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 widely used for that purpose already 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 and server deployments and software to change significantly which is contrary to the goal of promoting more TLS use. Encouraging use of STARTTLS on port 587 would not create interoperability problems, but is unlikely to have impact on 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.

Appendix B. Change Log

Changes since draft-ietf-uta-email-deep-07:

o After discussion with the WG in Prague, removed BCP language and once again made unambiguous that this is intended as a standards-track document.

o Server implementations now MUST implement TLS 1.2, consistent with
   RFC 7525. MUAs may still consider a TLS 1.1 session as meeting
   minimum confidentiality requirements.

o MSPs now MUST support TLS for POP, IMAP, Submission, and any other
   services that use username/password authentication.

o Added text to clarify the purpose of recommending that MSPs use
   DNS SRV records to advertise services.

o Changed text about MUAs not blindly trusting unsigned SRV records,
   to instead restate RFC 6186 requirements.

Changes since draft-ietf-uta-email-deep-06:

o On the recommendation of one of the co-chairs and some working
   group members, rewrote document with the intended status of BCP.
   This involved removing a great deal of text that consisted
   essentially of new protocol specification, especially the STS
   features, on the theory that a BCP should base its recommendations
   on current practice, and that new protocol features should be
   subject to the interoperability test requirements associated with
   normal standards-track documents.

Changes since draft-ietf-uta-email-deep-05:

o Clarify throughout that the confidentiality assurance level
   associated with a mail account is a minimum level; attempt to
   distinguish this from the current confidentiality level provided
   by a connection between client and server.

o Change naming for confidentiality assurance levels: instead of
   "high" or "no" confidence, assign numbers 1 and 0 to them
   respectively. This because it seems likely that in the not-too-
   distant future, what was defined in -05 as "high" confidence will
   be considered insufficient, and calling that "high" confidence
   will become misleading. For example, relying entirely on a list
   of trusted CAs to validate server certificates from arbitrary
   parties, appears to be less and less reliable in practice at
   thwarting MITM attacks.

o Clarify that if some services associated with a mail account don't
   meet the minimum confidentiality assurance level assigned to that
   account, other services that do meet that minimum confidentiality
   assurance level may continue to be used.

   o Clarify that successful negotiation of at least TLS version 1.1 is
      required as a condition of meeting confidentiality assurance level
      1.

   o Clarify that validation of a server certificate using either DANE
      or PKIX is sufficient to meet the certificate validation
      requirement of confidentiality assurance level 1.

   o Clarify that minimum confidentiality assurance levels are separate
      from security directives, and that the requirements of both
      mechanisms must be met.

   o Explicitly cite an example that a security directive of tls-
      version=1.2 won't be saved if the currently negotiated tls-version
      is 1.1. (This example already appeared a bit later in the text,
      but for author KM it seemed to make the mechanism clearer to use
      this example earlier.)

   o Clarify some protocol examples as to whether PKIX or DANE was used
      to verify a server's certificate.

   o Remove most references to DEEP as the conversion from DEEP to MUA-
      STS seemed incomplete, but kept the DEEP command for use in POP3
      on the assumption that author CN wanted it that way.

   o Removed most references to "latch" and derivative words.

   o Added pkix+dane as a value for the tls-cert directive, to indicate
      (from a server) that both PKIX and DANE validation will be
      supported, or (from a client) that both PKIX and DANE were used to
      validate a certificate. Also clarified what each of any, pkix,
      dane, and pkix+dane mean when advertised by a server and in
      particular that tls-cert=any provides no assurance of future PKIX
      verifiability in contrast to tls-cert=pkix or tls-cert=pkix+dane.
      It seemed important to support the ability to evolve to using
      multiple trust anchors for certificate validation, but also to
      allow servers to have the option to migrate from PKIX to DANE if
      that made sense for them. This change seemed less disruptive than
      either defining additional directives, or allowing multiple
      instances of the same directive with different values to appear in
      the same advertisement.

   o Clarify interaction of this specification with anti-virus / anti-
      spam mechanisms.

   Changes since draft-ietf-uta-email-deep-04:

o Swap sections 5.1 and 5.3 ("Email Security Tags" and "Server DEEP
   Status") as that order may aid understanding of the model. Also
   rewrote parts of these two sections to try to make the model
   clearer.

o Add text about versioning of security tags to make the model
   clearer.

o Add example of security tag upgrade.

o Convert remaining mention of TLS 1.0 to TLS 1.1.

o Change document title from DEEP to MUA STS to align with SMTP
   relay STS.

   * Slight updates to abstract and introductions.

   * Rename security latches/tags to security directives.

   * Rename server DEEP status to STS policy.

   * Change syntax to use directive-style HSTS syntax.

o Make HSTS reference normative.

o Remove SMTP DSN header as that belongs in SMTP relay STS document.

Changes since draft-ietf-uta-email-deep-03:

o Add more references to ietf-uta-email-tls-certs in implementation
   requirements section.

o Replace primary reference to RFC 6125 with ietf-uta-email-tls-
   certs, so move RFC 6125 to informative list for this
   specification.

Changes since draft-ietf-uta-email-deep-02:

o Make reference to design considerations explicit rather than
   "elsewhere in this document".

o Change provider requirement so SMTP submission services are
   separate from SMTP MTA services as opposed to the previous
   phrasing that required the servers be separate (which is too
   restrictive).

o Update DANE SMTP reference

Changes since draft-ietf-uta-email-deep-01:

o Change text in tls11 and tls12 registrations to clarify
   certificate rules, including additional PKIX and DANE references.

o Change from tls10 to tls11 (including reference) as the minimum.

o Fix typo in example 5.

o Remove open issues section; enough time has passed so not worth
   waiting for more input.

Changes since draft-ietf-uta-email-deep-00:

o Update and clarify abstract

o use term confidentiality instead of privacy in most cases.

o update open issues to request input for missing text.

o move certificate pinning sub-section to account setup section and
   attempt to define it more precisely.

o Add note about end-to-end encryption in AVAS section.

o swap order of DNSSEC and TLSA sub-sections.

o change meaning of 'tls10' and 'tls12' latches to require
   certificate validation.

o Replace cipher suite advice with reference to [RFC 7525](RFC 7525). Change
   examples to use TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as cipher
   suite.

o Add text to update IMAP, POP3 and Message Submission standards
   with newer TLS advice.

o Add clearer text in introduction that this does not cover SMTP
   relay.

o Update references to uta-tls-certs.

o Add paragraph to Implicit TLS for SMTP Submission section
   recommending that STARTTLS also be implemented.

Changes since draft-newman-email-deep-02:

o Changed "privacy assurance" to "confidentiality assurance"

o Changed "low privacy assurance" to "no confidentiality assurance"

o Attempt to improve definition of confidentiality assurance level.

o Add SHOULD indicate when MUA is showing list of mail accounts.

o Add SHOULD NOT latch tls10, tls12 tags until TLS negotiated.

o Removed sentence about deleting and re-creating the account in
   latch failure section.

o Remove use of word "fallback" with respect to TLS version
   negotiation.

o Added bullet about changes to Internet facing servers to MSP
   section.

o minor wording improvements based on feedback

Changes since -01:

o Updated abstract, introduction and document structure to focus
   more on mail user agent privacy assurance.

o Added email account privacy section, also moving section on
   account setup using SRV records to that section.

o Finished writing IANA considerations section

o Remove provisional concept and instead have server explicitly list
   security tags clients should latch.

o Added note that rules for the submissions port follow the same
   rules as those for the submit port.

o Reference and update advice in [RFC5068].

o Fixed typo in Client Certificate Authentication section.

o Removed tls-pfs security latch and all mention of perfect forward
   secrecy as it was controversial.

o Added reference to HSTS.

Changes since -00:

o Rewrote introduction to merge ideas from draft-moore-email-tls-00.

o Added Implicit TLS section, Account configuration section and IANA
   port registration updates based on draft-moore-email-tls-00.

o Add protocol details necessary to standardize implicit TLS for
   POP/IMAP/submission, using ideas from draft-melnikov-pop3-over-
   tls.

o Reduce initial set of security tags based on feedback.

o Add deep status concept to allow a window for software updates to
   be backed out before latches make that problematic, as well as to
   provide service providers with a mechanism they can use to assist
   customers in the event of a privacy failure.

o Add DNS SRV section from draft-moore-email-tls-00.

o Write most of the missing IANA considerations section.

o Rewrite most of implementation requirements section based more on
   draft-moore-email-tls-00. Remove new cipher requirements for now
   because those may be dealt with elsewhere.

Appendix C. **Acknowledgements**

   Thanks to Ned Freed for discussion of the initial latch concepts in
   this document. Thanks to Alexey Melnikov for draft-melnikov-pop3-
   over-tls-02, 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
   US

   Email: moore@network-heretics.com

Chris Newman
Oracle
440 E. Huntington Dr., Suite 400
Arcadia, CA 91006
US

Email: chris.newman@oracle.com