Network Working Group
Internet Draft
Intended status: Proposed Standard
Expires: September 2011

P. Sangster
Symantec Corporation

March 3, 2011

PT-TLS: A Posture Transport (PT) Protocol Compatible with TNC Using Transport Layer Security (TLS)
draft-sangster-nea-pt-tls-02.txt

Abstract

This document specifies PT-TLS, a Posture Transport (PT) protocol compatible with the Trusted Computing Group's IF-T Binding to TLS 1.0 protocol. The document then evaluates PT-TLS against the requirements defined in the NEA Overview and Requirements and PB-TNC specifications.

Table of Contents

1. Introduction

This document specifies PT-TLS, a Posture Transport (PT) protocol compatible with the Trusted Computing Group's IF-T Binding to TLS 1.0 protocol [IFT-TLS]. The document then evaluates PT-TLS against the applicable requirements defined in the NEA Overview and Requirements [RFC5209] and PB-TNC [RFC5793] specifications.

NEA protocols are intended to be used for pre-admission assessment of endpoints joining the network and to assess endpoints already present on the network. In order to support both usage models, two different types (or bindings) of PT protocols are necessary to operate before and after the endpoint has an assigned IP address and other network layer information. This specification focuses on the PT protocol used to assess endpoints already present on the network and thus is able to use TCP/IP based transport protocols.

The PT protocol in the NEA architecture is responsible for transporting PB-TNC batches (often containing PA-TNC [RFC5792] attributes) over the network between the Posture Transport Client component of the NEA Client and the Posture Transport Server component of the NEA Server. The PT protocol also offers strong security protections to ensure the exchanged messages are protected from a variety of threats from hostile intermediaries.

1.1. Prerequisites

This document does not define an architecture or reference model. Instead, it defines one binding of the PT protocol that works within the reference model described in the NEA Overview and Requirements specification. The reader is assumed to be thoroughly familiar with the NEA Overview and Requirements specification. No familiarity with TCG specifications is assumed.

1.2. Message Diagram Conventions

This specification defines the syntax of PT-TLS messages using diagrams. Each diagram depicts the format and size of each field in bits. Implementations MUST send the bits in each diagram as they are shown, traversing the diagram from top to bottom and then from left to right within each line (which represents a 32-bit quantity). Multi-byte fields representing numeric values must be sent in network (big endian) byte order.

Descriptions of bit field (e.g. flag) values are described referring to the position of the bit within the field. These bit positions are

numbered from the most significant bit through the least significant bit so a one octet field with only bit 0 set has the value 0x80.

1.3. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Design Considerations

This section discusses some of the key design considerations for the PT protocol. This document specifies the PT binding for use when performing an assessment or reassessment after the endpoint has been admitted to the network and is capable of using TCP/IP to communicate with the NEA Server. If the endpoint does not yet have TCP/IP layer access to the NEA Server (and vice versa), the endpoint should use the PT-EAP (Posture Transport (PT) Protocol for EAP Tunnel Methods) [PT-EAP] protocol when performing an assessment.

Because the endpoint has TCP/IP access to the NEA Server (potentially on a restricted portion of the network), the NEA Client and NEA Server have the ability to establish (or re-use) a reliable TCP/IP connection in order to perform the assessment. The TCP/IP connection enables the assessment to occur over a relatively high performance, reliable channel capable of supporting multiple roundtrip message exchanges in full duplex manner. These connection properties are very different from what is available when the endpoint is initially joining the network (e.g. during an 802.1X based assessment), therefore the design described in this specification follows a different path to maximize the benefits of the underlying TCP/IP connection.

2.1. Benefits of TCP/IP Connectivity

The PT protocol is typically able to offer to the NEA Client and NEA Server significantly higher quality of service and flexibility of operation than link layer oriented bindings such as PT-EAP (Posture Transport (PT) Protocol for EAP Tunnel Methods). However, there may be some added risks when the endpoint is on the network prior to its initial assessment (if no admission time assessment had been performed). Because of these risks, the combined use of an EAP-based assessment during admission followed by reassessment using TCP/IP may be appropriate in some environments.
Some of the benefits to having a TCP/IP based transport during an assessment include:

- Full Duplex connectivity - can send multiple assessment messages prior to receiving a response including sending of asynchronous messages (e.g. alerts of posture or policy changes)

- High Bandwidth - potentially much higher bandwidth than other transports (e.g. EAP) allowing more in-band data (e.g. remediation, verbose posture information)

- Large Messages - ability to send very large PA messages without directly fragmenting them (underlying carrier protocol may introduce fragmentation)

- Bi-directional - NEA Client and NEA Server can initiate an assessment or reassessment

- Multiple Roundtrips - NEA Client and NEA Server can exchange numerous messages without fear of infrastructure timeouts. However, the entire exchange should be kept as brief as possible if the user has to wait for its completion.

2.2. Leveraging Proven TLS Security

All PT protocol bindings must be capable of providing strong authentication, integrity and confidentiality protection for the PB-TNC batches. Rather than define a new protocol over TCP/IP to provide adequate protection, this specification requires the use of Transport Layer Security [RFC5246] to secure the connection. TLS was selected because it's a widely deployed protocol with parallel protections to a number of the EAP tunnel methods, and it meets all of the security requirements.

2.3. TLV-Oriented Based Message Encapsulation

The design of the PT-TLS protocol is based upon the use of type-length-value (TLV) oriented protocol message that identifies the type of message, the message's length and a potentially variable length payload value. The use of a TLV orientated encoding was chosen to match the Internet standard PA-TNC and PB-TNC protocols. Because the PA-TNC, PB-TNC and PT-TLS protocols are typically implemented inside the same process space, this allows a common set of message parsing code to be used. Similarly creation of debugging tools is simplified by the common encoding methodologies. TLV-based encoding was used in each of the NEA protocols in part because it enables a very space efficient representation on the network and is simpler to parse than some other encodings to benefit lower powered (or battery constrained) devices.

2.4. No Change to Base TLS Protocol

During the design of the PT-TLS protocol, several approaches were considered with different costs and benefits. Several considered approaches involved integrating the PT protocol into the TLS handshake protocol. Because the PT protocol requires the underlying TLS carrier to provide security protections, the PT protocol couldn't operate before the cipher suites were negotiated and in use. One option was to integrate into the TLS handshake protocol after the ChangeCipherSpec phase allowing the PT message to be protected. The benefit of this approach is that the assessment protocol could operate below the application protocols allowing for easier integration into applications. However, making this change would require some extensions to the TLS handshake protocol standards and existing widely deployed TLS implementations, so it wasn't clear that the cost was warranted, particularly because the application independence can also be offered by a shim library between the application and TLS library that provides the PT protocol encapsulation/decapsulation.

The other general approach considered was to have PT-TLS layer on top of TLS as an application protocol (using the standard application_data ContentType). This has the advantage that existing TLS software could be used. However, the PB-TNC traffic would need to be encapsulated/decapsulated by a new PT-TLS protocol layer before being passed to the TLS library. This didn't seem like a significant issue as PB-TNC is architected to layer on PT anyway.

After considering the different options, it was determined that layering the PT protocol on top of the TLS protocol without requiring current TLS protocol implementations to change met all the requirements and offered the best path toward rapid adoption and deployment. Therefore the following sections describe a PT protocol that is carried on top of TLS.

3. PT-TLS Protocol

This section specifies the PT-TLS protocol, a Posture Transport (PT) protocol carried by the Transport Layer Security (TLS) protocol over a TCP/IP network. This protocol runs directly on top of TLS as an application. This means PT-TLS is encapsulated within the TLS Record Layer protocol using the standard ContentType for applications (application_data).

3.1. Initiating a PT-TLS Session

The PT-TLS protocol may be initiated by a Posture Transport Client or a Posture Transport Server. This flexibility supports different use cases. For example, a Posture Transport Client that wishes to trigger a NEA assessment to determine whether its security posture is good can start up a PT-TLS session and request a posture assessment. On the other hand, when an endpoint requests access to a protected network or resource, a Posture Transport Server can start up a PT-TLS session and perform a posture assessment before deciding whether to grant access.

The party that initiates a PT-TLS session is known as the "PT-TLS session initiator". The other party in the session (which receives the request to open a PT-TLS session) is known as the "PT-TLS session responder".

3.1.1. Issues with Server Initiated PT-TLS Sessions

In order for a NEA Server to establish a PT-TLS session, the NEA Client needs to be listening for a connection request on a TCP port known by the NEA Server. In many deployments, the security policies (e.g. firewall software) of an endpoint are designed to minimize the number of open inbound TCP/UDP ports that are available to the network to reduce the potential attack footprint. This is one issue that makes it difficult for a NEA Server to initiate a PT-TLS session.

Another issue with this scenario involves X.509 certificates. When the NEA Server creates a TLS session to the NEA Client, the NEA Client is effectively acting as the TLS server during the TLS protocol exchange. This means the NEA Client would typically need to possess an X.509 certificate to protect the initial portion of the TLS handshake. In situations where the NEA Server initiates the creation of the TLS session, both the NEA Client and NEA Server MUST possess X.509 certificates to fully authenticate the session. For many deployments, provisioning X.509 certificates to all NEA Clients has scalability and cost issues; therefore, it is recommended that the NEA Client not listen for connection requests from the NEA Server but instead establish and maintain a TLS session to the NEA Server proactively, so either party can initiate an assessment using the preexisting TLS session as required.

Therefore, NEA Clients SHOULD be capable of establishing and holding open a TLS session with the NEA Server immediately after obtaining network access. A NEA Client MAY listen for connection requests from the NEA Server and establish a new PT-TLS session when one does not already exist. Having an existing PT-TLS session allows either party

to initiate an assessment without requiring the NEA Client to be listening for new connection requests.

3.1.2. Establish or Re-Use Existing PT-TLS Session

A single PT-TLS session can support multiple NEA assessments, which can be started by either party (the PT-TLS session initiator or the PT-TLS session responder). The party that starts a NEA assessment is known as the "assessment initiator" and the other party is known as the "assessment responder".

If the assessment initiator already has a PT-TLS session to the assessment responder, the initiator can re-use this session; otherwise, a new PT-TLS session must be established.

3.2. TCP Port Usage

In order for a PT-TLS session initiator to establish a TCP connection to a PT-TLS session responder, the initiator needs to know the TCP port number on which the responder is listening for assessment requests. Therefore, this specification requests the IANA reserve a well known TCP port number for use with the PT-TLS protocol upon publication of this specification as an Internet standard RFC.

3.3. Preventing MITM Attacks with Channel Bindings

As described in the NEA Asokan Attack Analysis [ASOKAN], a sophisticated MITM attack can be mounted against NEA systems. The attacker forwards PA-TNC messages from a healthy machine through an unhealthy one so that the unhealthy machine can gain network access. Because there are easier attacks on NEA systems, like having the unhealthy machine lie about its configuration, this attack is generally only mounted against machines with an External Measurement Agent (EMA). The EMA is a separate entity, difficult to compromise, which measures and attests to the configuration of the endpoint.

To protect against NEA Asokan attacks, the Posture Broker on an EMA-equipped endpoint should pass the tls-unique channel binding [RFC5929] for PT-TLS's underlying TLS session to the EMA. This value can then be included in the EMA's attestation and the Posture Validator responsible for communicating with the EMA may then confirm that the value matches the tls-unique channel binding for its end of the connection. If the values match, the posture sent by the EMA and NEA Client is from the same endpoint as the client side of the TLS connection (since the endpoint knows the tls-unique value), so no man-in-the-middle is forwarding posture. If they differ, an attack has been detected. The Posture Validator SHOULD fail its verification of the endpoint if an attack has been detected.

3.4. PT-TLS Message Flow

This section discusses the general flow of messages between the NEA Client's Posture Transport Client and the NEA Server's Posture Transport Server in order to perform NEA assessments using the PT-TLS protocol.

3.4.1. Assessment Triggers

Initially, the NEA Client or NEA Server will decide that an assessment is needed. What stimulates the decision to perform an assessment is outside the scope of this specification, but some examples include:

- NEA Server becoming aware of suspicious behavior on an endpoint
- NEA Server receiving new policies requiring immediate action
- NEA Client noticing a change in local security posture
- NEA Client wishing to access a protected network or resource

Because either the NEA Client or NEA Server can trigger the establishment of the TLS session and initiate the assessment, this document will use the terms "assessment initiator" and the "assessment responder". This nomenclature allows either NEA component to fill either of the PT-TLS roles.

3.4.2. PT-TLS Message Exchange Phases

The PT-TLS message exchange occurs in three distinct phases:

- TLS Setup (including TLS Handshake protocol)
- PT-TLS Negotiation
- PT-TLS Data Transport

The TLS Setup phase is responsible for the establishment of the TCP connection and the TLS protections for the PT-TLS messages. The TLS Setup phase normally starts with the establishment of a TCP connection between the Posture Transport Client and Posture Transport Server. The new connection triggers the TLS Handshake protocol to establish the cryptographic protections for the TLS session. The TLS Setup phase SHOULD NOT be repeated after the PT-TLS Data Transport phase has been reached unless a change of TLS cipher suite or keying material is required to properly protect the session. This phase

also enables the establishment of the tls-unique shared secret that can be used in a later phase to bind the posture sent with this TLS connection.

The PT-TLS Negotiation phase is only performed at the start of the first assessment on a TLS session. During this phase, the NEA Client and NEA Server discover each other's PT-TLS capabilities and establish a context that will apply to all future PT-TLS messages sent over the TLS session. The PT-TLS Negotiation phase MUST NOT be repeated after the session has entered the Data Transport phase. NEA assessment messages (PB-TNC batches) MUST NOT be sent by the NEA Client or NEA Server prior to the completion of the PT-TLS Negotiation phase to ensure that the security protections for the session are properly established and applied to the NEA assessment messages.

Finally the Data Transport phase allows the NEA Client and NEA Server to exchange PT messages under the protection of the TLS session consistent with the capabilities established in earlier phases. The exchanged messages can be a PT-TLS protected NEA assessment as described in this specification or other vendor-defined PT-TLS exchanged messages.

3.4.2.1. TLS Setup Phase

After a new TCP connection is established between the Posture Transport Client and Posture Transport Server, a standard TLS exchange is performed to negotiate a common security context for protecting subsequent communications. As discussed in section 3.4.1. , the TCP connection establishment and/or the TLS handshake protocol could be initiated by either the NEA Client or NEA Server. The most common situation would be for the assessment initiator to trigger the creation of the TCP connection and TLS handshake, so an assessment could begin when no session already exists. When the NEA Server has initiated the TLS Setup, the NEA Server is acting as a TLS client and the NEA Client is the TLS server (accepting the inbound TLS session request). The expected normal case is that the NEA Client initiates this phase, so that the NEA Server is acting as the TLS server and therefore the bootstrapping of the security of the TLS session is using the NEA Server's certificate. Having the NEA Client initiate the TLS session avoids the need for the NEA Client to also possess a certificate.

During the TLS Setup phase of PT-TLS, the PT-TLS session initiator contacts the listening port of the TLS session responder and performs a TLS handshake. The PT-TLS session responder MUST possess a trustworthy X.509 certificate used to authenticate to the TLS initiator and used to bootstrap the security protections of the TLS

session. The PT-TLS session initiator MAY also use an X.509

certificate to authenticate to the PT-TLS session responder providing for a bi-directional authentication of the PT-TLS session. Due to deployment issues with issuing and distributing certificates to a potentially large number of NEA Clients, this specification allows the NEA Client to be authenticated during the PT-TLS Negotiation phase using other more cost effective methods. At the conclusion of a successful initial TLS Setup phase, the NEA Client and NEA Server have a protected session to exchange messages. This allows the protocol to transition to the PT-TLS Negotiation phase.

3.4.2.2. PT-TLS Negotiation Phase

Once a TLS session has been established between Posture Transport Client and Posture Transport Server, the PT-TLS session initiator sends a Version Request Message indicating it is supported PT-TLS protocol version range. Next, the PT-TLS session responder sends a Version Response Message which selects a protocol version from within the range offered. The PT-TLS session responder SHOULD select the preferred version offered if supported; otherwise, the highest version that the responder is able to support from the received Version Request Message. If the PT-TLS session responder is unable or unwilling to support any of the versions included in the Version Request Message, the responder SHOULD send a Version Not Supported error message.

If no client side authentication has occurred during the TLS Setup phase, the Posture Transport Server can authenticate the client using PT-TLS client authentication messages. If the Posture Transport Server wishes to trigger a client authentication exchange, the Posture Transport Server SHOULD send a Client Authentication Request message (see section 3.8.1. for details). The Posture Transport Server MAY skip the Client Authentication Request exchange and instead start with the client authentication by sending a Client Authentication Challenge message if it only supports one type of authentication.

When the Posture Transport Client receives the Client Authentication Request, the Posture Transport Client responds with a Client Authentication Selection message indicating the method of authentication to be used. Upon selecting an appropriate authentication method, the Posture Transport Server requests the client's identity and authenticator information using the PT-TLS Client Authentication Challenge message. The Posture Transport Client responds with the requested information following the selected authentication scheme in a Client Authentication Response message. The Posture Transport Client and Server might exchange multiple roundtrips of client authentication messages in order to perform the

authentication depending on the type of authentication selected. When the client authentication successfully completes, the PT-TLS session transitions into the Data Transport phase, where it will remain for the duration of the session.

3.4.2.3. PT-TLS Data Transport Phase

Once a PT-TLS session is available to carry NEA assessments, either the Posture Transport Client or Server can start an assessment when provided a PB-TNC batch for transmission. The assessment initiator first envelopes the PB-TNC batch in a PT-TLS message, then assigns a message identifier to the message and finally transmits it over the session. The assessment responder validates the PT-TLS message and delivers the encapsulated PB-TNC batch to its upstream component (Posture Broker Client or Server).

Most PT-TLS messages contain PB-TNC batches that house PA-TNC requests for posture information or a response containing the requested posture information. The Posture Transport Client and Posture Transport Server may also exchange messages between them, such as a PT-TLS Error Message indicating that a problem occurred processing a message. During an assessment, the Posture Transport Client and Server merely encapsulate and exchange the PB-TNC batches and are unaware of the state of the assessment.

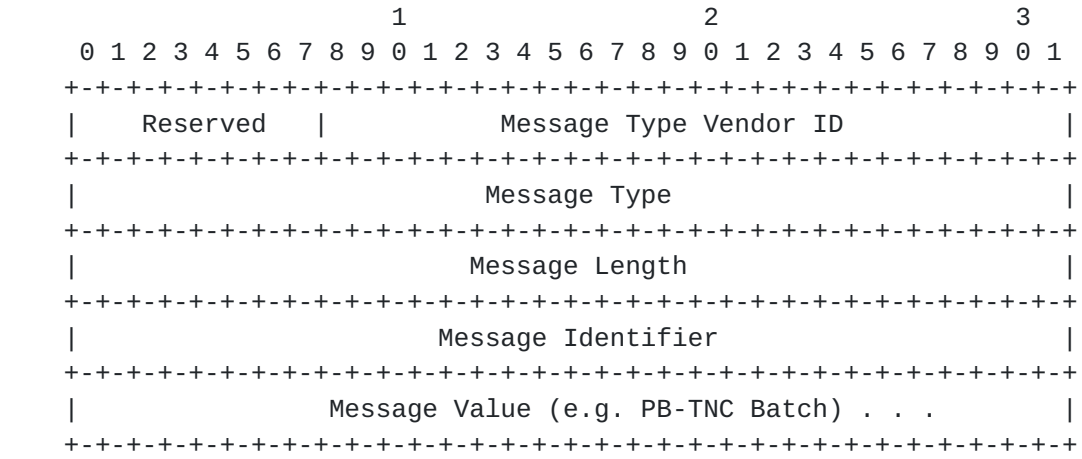
The PT-TLS protocol allows either party to send a PT-TLS message at any time, reflecting the full duplex nature of the underlying TLS session. For example, an assessment initiator may send several PT-TLS messages prior to receiving any responses from the assessment responder. All implementations of PT-TLS MUST support full duplex PT-TLS message exchange. However, some NEA protocols may not be able to make use of the full-duplex message exchange.

3.4.3. TLS Requirements

In order to ensure that strong security is always available for deployers and to improve interoperability, this section discusses some requirements on the underlying TLS transport used by PT-TLS. Implementations of PT-TLS MUST support use of TLS 1.1 [RFC4346] and SHOULD also include support for TLS 1.2 [RFC5246]. For each TLS version supported, implementations of the PT-TLS MUST at least support the TLS_RSA_WITH_AES_128_CBC_SHA cipher suite. This cipher suite requires the server to provide a certificate that can be used during the key exchange. Implementations SHOULD NOT include support for cipher suites that do not minimally offer PT-TLS session responder (typically Posture Transport Server) authentication, such as the anonymous Diffie-Hellman cipher suites (e.g. TLS_DH_anon_WITH_AES_128_CBC_SHA).

3.5. PT-TLS Message Format

This section describes the format and semantics of the PT-TLS message. Every message sent over a PT-TLS session MUST start with the PT-TLS header described in this section.
The following is the PT-TLS header:



Reserved

Reserved for future use. This field MUST be set to 0 on transmission and ignored upon reception.

Message Type Vendor ID

This field indicates the owner of the name space associated with the Message Type. This is accomplished by specifying the 24 bit SMI Private Enterprise Number (Vendor ID) of the party who owns the Message Type name space. IETF Standard PT-TLS Message Types MUST use zero (0) in this field.

The PT-TLS Message Type Vendor ID 0xffffff is reserved. Posture Transport Clients and Servers MUST NOT send PT-TLS messages in which the PT-TLS Message Type Vendor ID has this reserved value (0xffffff). If a Posture Transport Client or Posture Transport Server receives a message containing this reserved value (0xffffff) in the PT-TLS Message Type Vendor ID, the recipient SHOULD respond with an Invalid Parameter error code in a PT-TLS Error message.

Message Type

This field defines the type of the PT-TLS message within the scope of the specified Message Type Vendor ID that is included in the Message Value field. The specific IETF standard values allowable in this field when the Message Type Vendor ID is the IETF SMI Private Enterprise Number value (0) are defined in section 3.6. Recipients of a message containing a Message Type Vendor ID and Message Type that is unrecognized SHOULD respond with a Type Not Supported error code in a PT-TLS Error message.

Posture Transport Clients and Posture Transport Servers MUST NOT require support for particular vendor-defined PT-TLS Message Types and MUST interoperate with other parties despite any differences in the set of vendor-defined PT-TLS Message Types supported (although they MAY permit administrators to configure them to require support for specific vendor-defined PT-TLS message types).

If the PT-TLS Message Type Vendor ID field has the value zero (0), then the PT-TLS Message Type field contains an IETF Standard PT-TLS Message Type, as listed in the IANA registry. IANA maintains a registry of PT-TLS Message Types. Entries in this registry are added by Expert Review with Specification Required, following the guidelines in section 6.1. Section 3.6. of this specification defines the initial set of IETF Standard PT-TLS Message Types.

The PT-TLS Message Type 0xffffffff is reserved. Posture Transport Clients and Posture Transport Servers MUST NOT send PT-TLS messages in which the PT-TLS Message Type has this reserved value (0xffffffff). If a Posture Transport Client or Posture Transport Server receives a message in which the PT-TLS Message Type has this reserved value (0xffffffff), it SHOULD respond with an Invalid Parameter error code in a PT-TLS Error message.

Message Length

This field contains the length in octets of the entire PT-TLS message (including the entire header). Therefore, this value MUST always be at least 16. Any Posture Transport Client or Posture Transport Server that receives a message with a PT-TLS Message Length field whose value is less than 16 SHOULD respond with an Invalid Parameter PT-TLS error code. Similarly, if a Posture Transport Client or Posture Transport Server receives a PT-TLS message for a Message Type that has a known Message Length and the Message Length indicates a different value (greater or less than the expected value), the recipient SHOULD respond with an Invalid Parameter PT-TLS error code.

Message Identifier

This field contains a value that uniquely identifies the PT-TLS message on a per message sender (Posture Transport Client or Server) basis. This value can be copied into the body of a response message to indicate which message was received and caused the response. For example, this field is included in the PT-TLS Error Message so the recipient can determine which message sent caused the error.

The Message Identifier MUST be a monotonically increasing counter starting at zero indicating the number of the messages the sender has transmitted over the TLS session. It is possible that a busy or long lived session might exceed 2^32-1 messages sent, so the message sender MUST roll over to zero upon reaching the 2^32nd message, thus restarting the increasing counter. During a rollover, it is feasible that the message recipient could be confused if it keeps track of every previously received Message Identifier, so recipients MUST be able to handle roll over situations without generating errors.

Message Value

The contents of this field vary depending on the particular Message Type Vendor ID and Message Type given in the PT-TLS header for this PT-TLS message. This field most frequently contains a PB-TNC batch. The contents of this field for each of the IETF Standard PT-TLS Message Types are defined in this specification.

3.6. IETF Standard PT-TLS Message Types

This section defines the NEA standard PT-TLS Message Types used to carry PT-TLS messages and PB-TNC batches between the Posture Transport Client and Posture Transport Server.

The following table summarizes the initial set of IETF standard message type values, which are used with the PT-TLS Message Type Vendor ID field set to the IETF SMI PEN (0).

Value (Name)	Definition
0 (Experimental)	Reserved for experimental use. This type will not offer interoperability but allows for experimentation. This message type MUST only be sent when the NEA Client and NEA Server are in

	the Data Transport phase and only on a restricted, experimental network. Production code MUST send an Invalid Message error code in a PT-TLS Error message if an Experimental message is received.
1 (Version Request)	Version negotiation request including the range of versions supported by the sender. This message type MUST only be sent by the TLS session initiator as the first PT-TLS message in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Version Request is received at another time.
2 (Version Response)	PT-TLS protocol version selected by the responder. This message type MUST only be sent by the TLS session responder as the second message in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Version Response is received at another time.
3 (Client Auth Request)	Request for authentication of client (PT-TLS session initiator). This message includes the PT-TLS session responder's supported set of authentication methods. This message can be used to start an authentication of the PT-TLS session initiator. This message type MUST only be sent by the PT-TLS session initiator in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Client Auth Request message is received at another time.
4 (Client Auth Selection)	Authentication method selected by PT-TLS session initiator. This message type MUST only be sent by the PT-TLS session initiator in response to a

	Client Auth Request message sent in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Client Auth Selection message is received at another time.
5 (Client Auth Challenge)	Client authentication challenge from the PT-TLS session responder (normally NEA Server). This message type MUST only be sent by the PT-TLS session responder in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Client Auth Challenge is received after the PT-TLS Negotiation phase.
6 (Client Auth Response)	Identity and authenticator information from the PT-TLS session initiator (normally NEA Client). This message type MUST only be sent by the PT-TLS session initiator in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Client Auth Response message is received after the PT-TLS Negotiation phase.
7 (Client Auth Success)	Indication that client authentication was completed successfully so PT-TLS data messages may now be sent. This message type MUST only be sent by the PT-TLS session responder when the NEA Client and NEA Server are in the PT-TLS Negotiation phase. Recipients MUST send an Invalid Message error code in a PT-TLS Error message if a Client Auth Success is received after the PT-TLS Negotiation phase.
8 (PB-TNC Batch)	Contains a PB-TNC batch. For more information on PB-TNC batches see section 4 of the PB-TNC specification. This message type MUST only be sent when the NEA Client and NEA Server are in the PT-TLS Data Transport phase.

	Recipients SHOULD send an Invalid Message error code in a PT-TLS Error message if a PB-TNC Batch is received outside of the Data Transport phase.
9 (PT-TLS Error)	PT-TLS Error message as described in section 3.9. This message type may be used during any PT-TLS phase.
10+ (Reserved)	These values are reserved for future allocation following guidelines defined in the IANA Considerations section 6.1. Recipients of messages of type 13 or higher that do not support the PT-TLS Message Type Vendor ID and PT-TLS Message Type of a received PT-TLS message MUST respond with a Type Not Supported PT-TLS error code in a PT-TLS Error message.

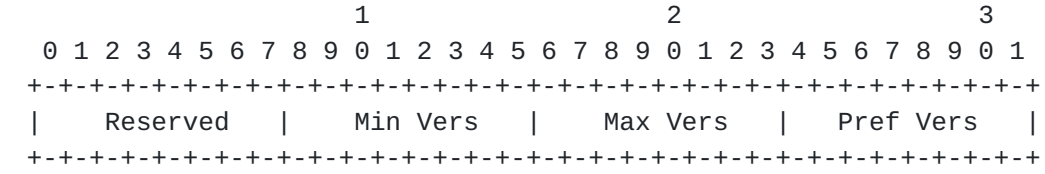
3.7. PT-TLS Version Negotiation

This section describes the message format and semantics for the PT-TLS protocol version negotiation. This exchange is used by the PT-TLS Session Initiator to trigger a version negotiation at the start of an assessment. The PT-TLS session initiator MUST send a Version Request message as its first PT-TLS message and MUST NOT send any other PT-TLS messages on this connection until it receives a Version Response message or an Error message. The PT-TLS session responder MUST complete the version negotiation (or cause an error) prior to sending or accepting reception of any additional messages. After the successful completion of the version negotiation, both the Posture Transport Client and Posture Transport Server MUST only send messages compliant with the negotiated protocol version. Subsequent assessments on the same session MUST use the negotiated version number and therefore SHOULD NOT send additional version negotiation messages.

3.7.1. Version Request Message

This message is sent by a PT-TLS Session Initiator as the first PT-TLS message in a PT-TLS session. This message discloses the sender's supported versions of the PT-TLS protocol. To ensure compatibility, this message MUST always be sent using version 1 of the PT-TLS protocol. Recipients of this message MUST respond with a Version Response, or a PT-TLS Error message (Version Not Supported or Invalid

Message). The following diagram shows the format of the Version Request Message:



Reserved

Reserved for future use. This field MUST be set to 0 on transmission and ignored upon reception.

Min Vers

This field contains the minimum version of the PT-TLS protocol supported by the sender. This field MUST be set to 1 indicating support for the first version of PT-TLS. However, future versions of this specification will probably remove this requirement so PT-TLS Session Responders MUST be prepared to receive other values.

Max Vers

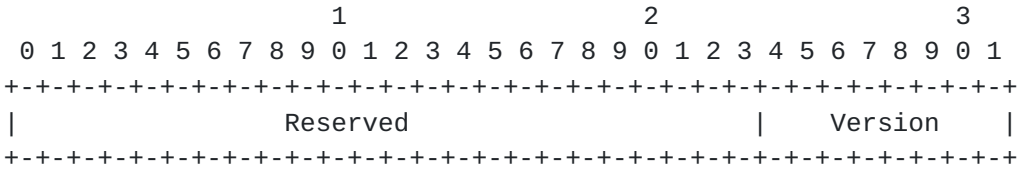
This field contains the maximum version of the PT-TLS protocol supported by the sender. This field MUST be set to 1 indicating support for the first version of PT-TLS. However, future versions of this specification will probably remove this requirement so PT-TLS Session Responders MUST be prepared to receive other values.

Pref Vers

This field contains the sender's preferred version of the PT-TLS protocol. This is a hint to the recipient that the sender would like this version selected if supported. The value of this field MUST fall within the range of Min Vers to Max Vers. This field MUST be set to 1 indicating support for the first version of PT-TLS. However, future versions of this specification will probably remove this requirement so PT-TLS Session Responders MUST be prepared to receive other values.

3.7.2. Version Response Message

This message is sent in response to receiving a Version Request Message at the start of a new assessment session. If a recipient receives a Version Request after a successful version negotiation has occurred on the session, the recipient SHOULD send an Invalid Message error code in a PT-TLS Error message and have TLS close the session. This message MUST be sent using the syntax, semantics, and requirements of the protocol version specified in this message.



Reserved

Reserved for future use. This field MUST be set to 0 on transmission and ignored upon reception.

Version

This field contains the version selected by the sender of this message. The version selected MUST be within the Min Vers to Max Vers inclusive range sent in the Version Request Message. If a PT-TLS Session Initiator receives a message with an invalid Version selected, the PT-TLS Session Initiator MUST respond with a Version Not Supported PT-TLS error message.

3.8. Client Authentication Message Exchange

This section includes a description of the message format and contents necessary to perform client authentication (authentication of the PT-TLS Session Initiator) over PT-TLS. The general model used for providing a client side authentication using PT-TLS messages over TLS is to have a simple authentication exchange roughly equivalent to basic authentication for HTTP [RFC2617] while also allowing for extensibility so stronger methods can be added in the future.

Implementations compliant with the PT-TLS specification MUST implement the Basic authentication type described in this section. Future specifications are expected to include additional types of authentication. For example, it is expected that a widely used extensible authentication

technology such as EAP [RFC3748] will be included in the future.

Because either the NEA Client or NEA Server can initiate the TLS session used for the assessment, either could act as the TLS server and be authenticated as part of the TLS exchange. Therefore, either the NEA Client or NEA Server could also be the party not authenticated during the TLS handshake (assuming that TLS mutual authentication is not used) and be required to authenticate using the PT-TLS client authentication. Typically the NEA Client would setup the PT-TLS session (see section 3.1.), so the NEA Server would be triggering the client authentication message exchanges and the NEA Client would be the party being authenticated, thus the name "client authentication".

If a client authentication is required, the TLS session responder (typically the NEA Server) MUST initiate the client authentication exchange by sending a Client Authentication Request message or a Client Authentication Challenge message. The Client Authentication Request message SHOULD be sent when the TLS session responder is willing to authenticate the client using multiple alternative authentication methods. The Client Authentication Request message includes a prioritized list of the authentication methods that the TLS session responder (often the NEA Server) is willing to use and allows for the selection of one for use with this session.

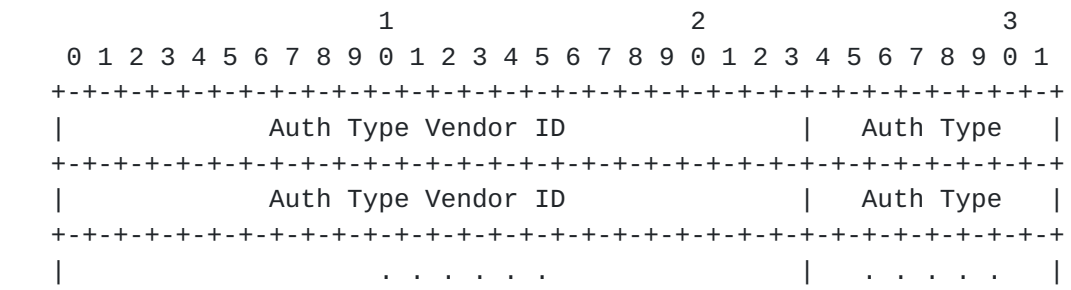
When a TLS session responder is only willing to accept the use of a single authentication method, the TLS session responder SHOULD optimistically start the authentication exchange by sending a Client Authentication Challenge in hopes that the other party is willing and able to use the supported type of authentication. If the PT-TLS Session Responder requires an authentication of the other party that was not performed during the TLS handshake and receives a PT-TLS Data Transport Phase message prior to client authentication successfully completing, the PT-TLS Session Responder SHOULD ignore the message and start the client authentication exchange (if it has not already done so). If a TLS Session Initiator receives a Client Authentication Challenge or Client Authentication Request as the next PT-TLS message after sending its first PT-TLS Data Transport Phase message, the initiator MUST assume that the TLS session responder requires an authentication prior to entering the PT-TLS Data Transport phase.

Upon reception of a Client Authentication Request, the recipient MUST send a Client Authentication Selection message that selects a single authentication method from the list in the Client Authentication Request message or send an Authentication Error error code in a PT-TLS Error message. When the TLS session responder (e.g. NEA Server) receives the Client Authentication Selection message, it MUST respond with a Client Authentication Challenge message containing the challenge information relevant to the selected type of authentication. Some authentication schemes might not require an initial challenge from the server so the Client Authentication Challenge message might contain minimal information and largely serve to start the authentication exchange. After the successful selection of an authentication method, the Client Authentication Request and Client Authentication Selection messages MUST NOT be used again on the session.

Now that an authentication method has been established, the client authentication involves a potentially multi-roundtrip message exchange until the PT-TLS Session Responder has confirmed the identity of the PT-TLS Session Initiator. The number of roundtrip messages and the contents of each message depend on the type of authentication selected. The client authentication messages are described in the following sub-sections.

3.8.1. Client Authentication Request Message

This message is sent when the TLS session responder (e.g. NEA Server) has decided that a client authentication is required. For example, this situation could occur following the initial establishment of the TLS session performing authentication only of the NEA Server when the NEA Server requires an authentication of the NEA Client.
The following diagram shows the format of the Client Authentication Request message. Note that this message contains a list of Auth Type Vendor ID and associated Auth Type fields. The overall length of the PT-TLS message is used by the recipient to determine the number of authentication types offered in this message since each entry is 32 bits in length.



Auth Type Vendor ID

This field indicates the owner of the name space associated with the following Auth Type field. Note that this field and the following Auth Type field will appear in pairs with one pair for every type of authentication being offered by the party requesting the authentication.

This field is the 24 bit SMI Private Enterprise Number (Vendor ID) of the party who owns the Auth Type name space for the subsequent field. The IETF standard Auth Type values defined in this specification MUST use the IETF SMI Private Enterprise Number value (0) in this field.

Auth Type

This field indicates a type of authentication that the PT-TLS Session Responder is willing to perform in order to authenticate the PT-TLS Session Initiator's identity. The ordering of the authentication types in the list SHOULD be in the order of preference of the sender with the most preferred type first. Client Authentication Request message recipients SHOULD process the list of authentication types in the order received and select the first type that is acceptable based on local policies.

3.8.1.1. Auth Type Values

This section defines the IETF standard PT-TLS Auth Types used to identify the method of client authentication being used within a PT-TLS session.

Posture Transport Clients and Posture Transport Servers MUST NOT require support for particular vendor-specific PT-TLS Auth Types and MUST interoperate with other parties despite any differences in the set of vendor-specific PT-TLS Auth Types supported (although they MAY permit administrators to configure vendor defined authentication types to be used).

When the PT-TLS Auth Type Vendor ID is set to zero (0), the PT-TLS Auth Type is an IETF Standard PT-TLS authentication method. IANA maintains a registry of the IETF standard and vendor-specific PT-TLS Auth Types. Entries in this registry are added by Expert Review with Specification Required, following the guidelines in section 6.1.

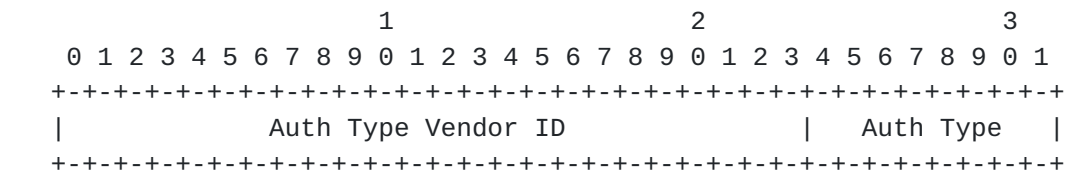
The following table summarizes the Auth Type values used when the Auth Type Vendor ID is set to the IETF SMI PEN (0).

Value (Name)	Definition
0 (Experimental)	Reserved for experimental use. This type will not offer interoperability but allows for experimentation. This value MUST be used only on a restricted, experimental network. Production code MUST NOT send an Experimental Auth Type and MUST send an Invalid Message error code in a PT-TLS Error message if an Experimental Auth Type is received.
1 (Basic Auth)	Indicates that the Authentication Information field contains a username and password as described in section 3.8.4.1.

3.8.2. Client Authentication Selection Message

This message is sent by the PT-TLS Session Initiator in response to reception of a Client Authentication Request message. This message indicates the TLS session initiator's (typically the NEA Client's) selection of an authentication method offered in the Client Authentication Request message. The values in this message (Auth Type Vendor ID and Auth Type) must match one of the options listed in the preceding Client Authentication Request message. During the establishment of the TLS session, the TLS session initiator (e.g. NEA Client) MAY authenticate using a TLS defined client authentication method such as using client side X.509 certificates. If the TLS client authentication did not occur and is required by the TLS session responder, then it SHOULD request the authentication using the PT-TLS Client Authentication Request message.

The following message shows the format of the Client Authentication Selection message:



Auth Type Vendor ID

This field indicates the owner of the name space associated with the following Auth Type field that was selected. The name space owner information is expressed as the 24 bit SMI Private Enterprise Number (Vendor ID) of the party who owns the Auth Type name space for the subsequent Auth Type field. IETF standard values defined in this specification MUST use the IETF SMI Private Enterprise Number value of zero (0) in this field.

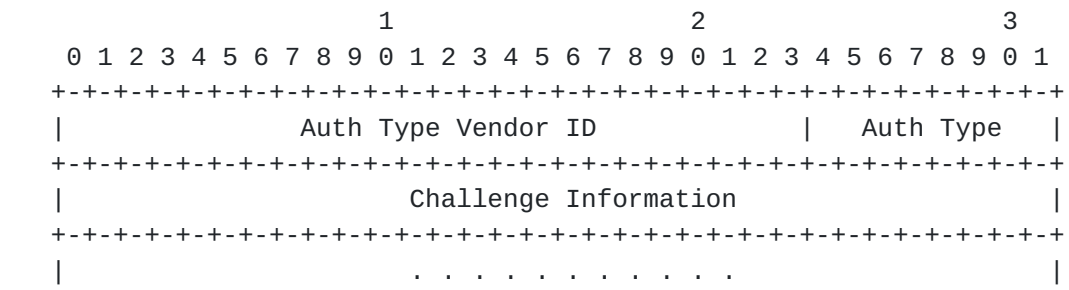
Auth Type

This field indicates a type of authentication that was selected from the list in the Client Authentication Request message received. The PT-TLS Session Initiator MUST select one authentication type (Auth Type Vendor ID and Auth Type) from the list sent in the Client Authentication Request message or send an Authentication error code in a PT-TLS Error message. The authentication type selection process SHOULD process the list in order and select the first type that is acceptable based upon its policies.

3.8.3. Client Authentication Challenge Message

This message is sent by the PT-TLS Session Responder (typically by the NEA Server) to initiate the authentication of the PT-TLS Session Initiator. Based upon the type of authentication being performed, the contents of the Challenge Information field will vary. For the details of the Challenge Information field for the Basic Authentication type see section 3.8.4.1.

The following message shows the format of the Client Authentication Challenge message:



Auth Type Vendor ID

This field indicates the owner of the name space associated with the following Auth Type field that was selected. The name space owner information is expressed as the 24 bit SMI Private Enterprise Number (Vendor ID) of the party who owns the Auth Type name space for the subsequent field. IETF standard values defined in this specification MUST use the IETF SMI Private Enterprise Number value of zero (0) in this field.

Auth Type

This field indicates the type of client authentication in use on the session. This field also indicates to the recipient the contents of the Challenge Information field (whose information varies based on authentication type and state).

Challenge Information

This field contains the authentication challenge in a format indicated by the type of authentication. The detailed format and semantics of this field for authentication types specified in this document are found in the following subsections.

3.8.3.1. Basic Authentication Challenge

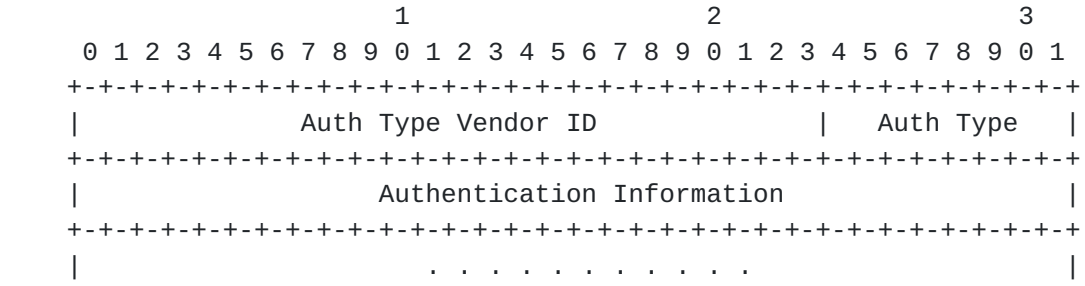
This type of authentication is modeled on HTTP basic authentication. This authentication involves the client sending a username and password (or passphrase) to the server for authentication. Note that the password will travel over the PT-TLS session without special protection but it is afforded the full protections of TLS, so passive attacks should be unable to steal these credentials.

For the Basic Authentication type of authentication, the Challenge Information field is empty. Basic authentication does not allow for the server to send information that alters the authentication response.

3.8.4. Client Authentication Response Message

This message is sent by the PT-TLS Session Initiator to prove its identity to the PT-TLS Session Responder. The format and contents of the Authentication Information vary depending on the type of authentication being performed and the state of the authentication exchange (e.g. when multi-roundtrip authentication protocols are used).

The following message shows the format of the Client Authentication Response message:



Auth Type Vendor ID

This field indicates the owner of the name space associated with the following Auth Type field that was selected. The name space owner information is expressed as the 24 bit SMI Private Enterprise Number (Vendor ID) of the party who owns the Auth Type name space for the subsequent field. IETF standard values defined in this specification MUST use the IETF SMI Private Enterprise Number value of zero (0) in this field.

Auth Type

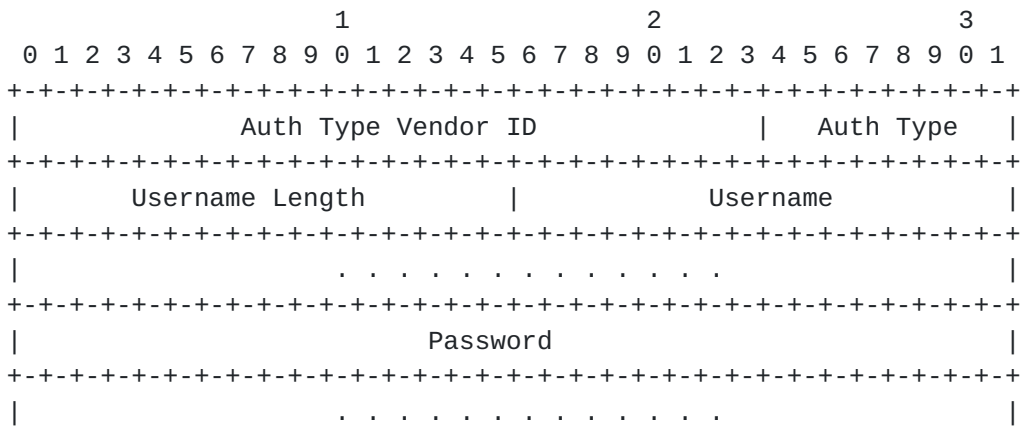
This field indicates the type of client authentication in use on the session. This field also indicates to the recipient the contents of the Challenge Information field (whose information varies based on authentication type and state).

Authentication Information

This field contains the authentication information in a format indicated by the type of authentication. The detailed format and semantics of this field for authentication types specified in this document are found in the following subsections.

3.8.4.1. Basic Authentication Information

This type of authentication is modeled on the HTTP basic authentication. This authentication involves the party being authenticated (the PT-TLS Session Initiator) sending a username and password (or passphrase) as a credential for authentication. Typically, the Authentication Information field will include the username and password for the NEA Client. The format and semantics are as follows:



Auth Type Vendor ID

This field indicates the owner of the name space associated with the following Auth Type field that was selected. The name space owner information is expressed as the 24 bit SMI Private Enterprise Number (Vendor ID) of the party who owns the Auth Type name space for the subsequent field. IETF standard values defined in this specification MUST use the IETF SMI Private Enterprise Number value of zero (0) in this field.

Auth Type

This field indicates the type of authentication in use on the session. This field also indicates to the recipient the contents of the Challenge Information field (whose information varies based on authentication type and state).

Username Length

This unsigned integer field indicates the octet length of the subsequent Username field. The Username field is variable length and is followed by the Password field that is also variable length, so the recipient needs to be able to identify the end of the Username and the start of the password.

Username

This field contains a string containing the identity of the party being authenticated. The Username MUST be encoded as a UTF-8 [RFC3629] string. NUL termination MUST NOT be employed.

Password

This field contains a string containing the authenticator associated with the claimed identity in the Username field. For the Basic type of authentication, the Password field MUST include a UTF-8 encoded string. NUL termination MUST NOT be employed.

3.8.5. Client Authentication Successful Message

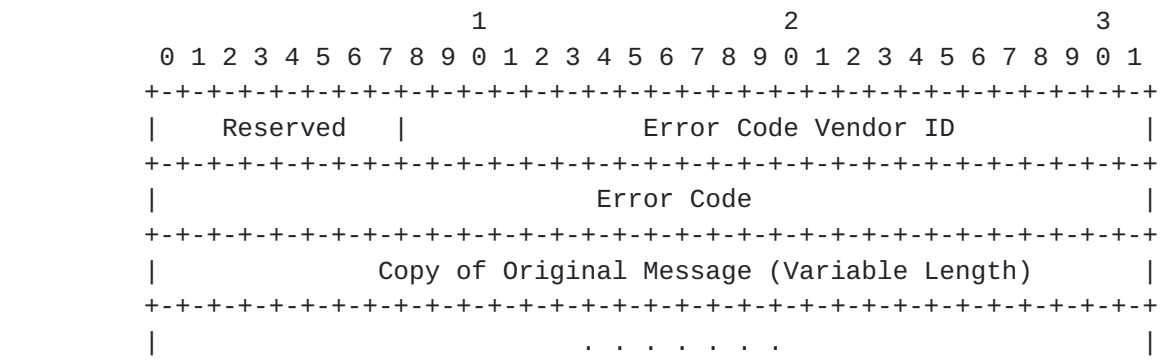
This message is sent by the PT-TLS Session Responder to indicate that it has successfully completed authentication of the claimed identity and the PT-TLS session will now enter the PT-TLS Data Transport Phase. This message does not contain a Message Value field since the Message Type carries the only needed semantic (authentication was successful). The Client Authentication Successful message MUST be sent by a PT-TLS Session Responder (typically the NEA Server) at the completion of a successful authentication to indicate that the PT-TLS Session Initiator may now start sending NEA assessment messages.

3.9. Error Message

This section describes the format and contents of the PT-TLS Error Message sent by the NEA Client or NEA Server when it detects a PT-TLS level protocol error. Each error message contains an error code indicating the error that occurred, followed by a copy of the message that caused the error.
When a PT-TLS error is received, the recipient MUST NOT respond with a PT-TLS error because this could result in an infinite loop of error messages being sent. Instead, the recipient MAY log the error,

modify its behavior to avoid future errors, ignore the error, terminate the assessment, or take other action as appropriate (as long as it is consistent with the requirements of this specification).

The Message Value portion of a PT-TLS Error Message contains the following information:



Reserved

Reserved for future use. This field MUST be set to 0 on transmission and ignored upon reception.

Error Code Vendor ID

This field contains the IANA assigned SMI Private Enterprise Number for the vendor whose Error Code name space is being used in the message. For IETF standard Error Code values this field MUST be set to zero (0). For other vendor-defined Error Code name spaces this field MUST be set to the SMI Private Enterprise Number of the vendor.

Error Code

This field contains the error code. This error code exists within the scope of Error Code Vendor ID in this message. Posture Transport Clients and Posture Transport Servers MUST NOT require support for particular vendor-specific PT-TLS Error Codes and MUST interoperate with other parties despite any differences in the set of vendor-specific PT-TLS Error Codes supported (although they MAY permit administrators to configure them to require support for specific PT-TLS error codes).

When the Error Code Vendor ID is set to the IETF Private Enterprise Number, the following table lists the supported IETF standard numeric error codes:

Value (Name)	Definition
0 (Reserved)	Reserved value indicates that the PT-TLS Error Message SHOULD be ignored by all recipients. This MAY be used for debugging purposes to allow a sender to see a copy of the message that was received while a receiver is operating on its contents.
1 (Malformed Message)	PT-TLS message unrecognized or unsupported. This error code SHOULD be sent when the basic message content sanity test fails. The sender of this error code MUST consider it a fatal error and abort the assessment.
2 (Version Not Supported)	This error SHOULD be sent when a PT-TLS session responder receives a PT-TLS Version Request message containing a range of version numbers that doesn't include any version numbers that the recipient is willing and able to support on the session. All PT-TLS messages carrying the Version Not Supported error code MUST use a Version number of 1. All parties that receive or send PT-TLS messages MUST be able to properly process an error message that meets this description, even if they cannot process any other aspect of PT-TLS version 1. The sender and receiver of this error code MUST consider this a fatal error and close the TLS session after sending or receiving this PT-TLS message.
3 (Type Not Supported)	PT-TLS message type unknown or not supported. When a recipient receives a PT-TLS message type that it does not support, it MUST send back this error, ignore the message and proceed. For

	example, this could occur if the sender used a Vendor ID for the Message Type that is not supported by the recipient. This error message does not indicate a fatal error has occurred, so the assessment is allowed to continue.
4 (Failed Authentication)	The authentication of the identity of the client failed. This could occur if the sent Username and Password (for the Basic authentication type) did not match those expected by the authenticating party. This error message does not indicate a fatal error has occurred, so the authentication is allowed to be re-started.
5 (Invalid Message)	PT-TLS message received was invalid based on the protocol state. For example, this error would be sent if a recipient receives a message associated with the PT-TLS Negotiation Phase (such as Version messages) after the protocol has reached the PT-TLS Data Transport Phase. The sender and receiver of this error code MUST consider it a fatal error and close the TLS session after sending or receiving this PT-TLS message.
6 (Authentication Error)	A fatal error occurred while trying to perform the client authentication. For example, the NEA Client is unable to support any of the offered types of authentication. The sender and receiver of this error code MUST consider it a fatal error and close the TLS session after sending or receiving this PT-TLS message.

Copy of Original Message

This variable length value contains a copy (up to 1024 bytes) of the original PT-TLS message that caused the error.

If the original message is longer than 1024 bytes, only the initial 1024 bytes will be included in this field. This field is included so the error recipient can determine which message sent caused the error. In particular, the recipient can use the Message Identifier field from the Copy of Original Message to determine which message caused the error.

4. Security Considerations

This section discusses the major threats potentially faced by each binding of the PT protocol and countermeasures provided by the PT-TLS protocol.

4.1. Trust Relationships

In order to understand where security countermeasures are necessary, this section starts with a discussion of where the NEA architecture envisions some trust relationships between the processing elements of the PT-TLS protocol. The following sub-sections discuss the trust properties associated with each portion of the NEA reference model directly involved with the processing of the PT-TLS protocol.

4.1.1. Posture Transport Client

The Posture Transport Client is trusted by the Posture Broker Client to:

- Not observe, fabricate or alter the contents of the PB-TNC batches received from the network
- Not observe, fabricate or alter the PB-TNC batches passed down from the Posture Broker Client for transmission on the network
- Transmit on the network any PB-TNC batches passed down from the Posture Broker Client
- Deliver properly security protected messages received from the network that are destined for the Posture Broker Client
- Provide configured security protections (e.g. authentication, integrity and confidentiality) for the Posture Broker Client's PB-TNC batches sent on the network
- Expose the authenticated identity of the Posture Transport Server

- Verify the security protections placed upon messages received from the network to ensure the messages are authentic and protected from attacks on the network

- Provide a secure, reliable, in order delivery, full duplex transport for the Posture Broker Client's messages

The Posture Transport Client is trusted by the Posture Transport Server to:

- Not send malicious traffic intending to harm (e.g. denial of service) the Posture Transport Server

- Not send malformed messages (e.g. messages lacking PT-TLS header)

- Not send invalid or incorrect responses to messages (e.g. errors when no error is warranted)

- Not ignore or drop messages causing issues for the protocol processing (e.g. dropping PT-TLS Client Authentication Challenge messages)

- Verify the security protections placed upon messages received from the network to ensure the messages are authentic and protected from attacks on the network

4.1.2. Posture Transport Server

The Posture Transport Server is trusted by the Posture Broker Server to:

- Not observe, fabricate or alter the contents of the PB-TNC batches received from the network

- Not observe, fabricate or alter the PB-TNC batches passed down from the Posture Broker Server for transmission on the network

- Transmit on the network any PB-TNC batches passed down from the Posture Broker Server

- Deliver properly security protected messages received from the network that are destined for the Posture Broker Server

- Provide configured security protections (e.g. authentication, integrity and confidentiality) for the Posture Broker Server's messages sent on the network

- Expose the authenticated identity of the Posture Transport Client
- Verify the security protections placed upon messages received from the network to ensure the messages are authentic and protected from attacks on the network
- Provide a secure, reliable, in order delivery, full duplex transport for the Posture Broker Server's messages

The Posture Transport Server is trusted by the Posture Transport Client to:

- Not send malicious traffic intending to harm (e.g. denial of service) the Posture Transport Server
- Not send malformed messages (e.g. messages lacking PT-TLS header)
- Not send invalid or incorrect responses to messages (e.g. errors when no error is warranted)
- Not ignore or drop messages causing issues for the protocol processing (e.g. dropping PT-TLS Client Authentication Successful messages)
- Verify the security protections placed upon messages received from the network to ensure the messages are authentic and protected from attacks on the network

4.2. Security Threats and Countermeasures

Beyond the trusted relationships assumed in section 4.1. the PT-TLS protocol faces a number of potential security attacks that could require security countermeasures.

Generally, the PT-TLS protocol is responsible for offering strong security protections for all of the NEA protocols so any threats to its ability to protect NEA protocol messages could be very damaging to deployments. Once the message is delivered to the Posture Broker Client or Posture Broker Server, the posture brokers are trusted to properly and safely process the messages.

4.2.1. Message Theft

When PT-TLS messages are sent over unprotected network links or spanning local software stacks that are not trusted, the contents of the messages may be subject to information theft by an intermediary party. This theft could result in information being recorded for

future use or analysis by the adversary. Messages observed by eavesdroppers could contain information that exposes potential weaknesses in the security of the endpoint, or system fingerprinting information easing the ability of the attacker to employ attacks more likely to be successful against the endpoint. The eavesdropper might also learn information about the endpoint or network policies that either singularly or collectively is considered sensitive information. For example, if PT-TLS does not provide confidentiality protection, an adversary could observe the PA-TNC attributes included in the PT-TLS message and determine that the endpoint is lacking patches, or particular sub-networks have more lenient policies.

In order to protect against NEA assessment message theft, the PT-TLS protocol provides strong cryptographic authentication, integrity and confidentiality protection. Deployers are strongly encouraged to employ best practice of the day TLS ciphers to ensure the information remains safe despite advances in technology and discovered cipher weaknesses. The use of bi-directional authentication of the assessment transport session ensures that only properly authenticated and authorized parties may be involved in an assessment dialog. The PT-TLS protocol also provides strong cryptography for all of the PB-TNC and PA-TNC protocol messages traveling over the network allowing the message contents to be hidden from potential theft by the adversary even if the attacker is able to observe the encrypted PT-TLS session.

4.2.2. Message Fabrication

Attackers on the network or present within the NEA system could introduce fabricated PT-TLS messages intending to trick or create a denial of service against aspects of an assessment. For example, an adversary could attempt to insert into the message exchange fake PT-TLS error codes in order to disrupt communications.

The PT-TLS protocol provides strong security protections for the complete message exchange over the network. These security protections prevent an intermediary from being able to insert fake messages into the assessment. In particular, the TLS's protocol use of hashing algorithms provides strong integrity protections that allow for detection of any changes in the content of the message stream. Additionally, adversaries are unable to observe the PT-TLS protocol exchanges because they are encrypted by the TLS ciphers, so would have difficulty in determining where to insert the falsified message, since the attacker is unable to determine where the message boundaries exist. Even a successful message insertion did occur; the recipient would be able to detect it due to the TLS cipher suite's integrity checking failing.

4.2.3. Message Modification

This attack could allow an active attacker capable of intercepting a message to modify a PT-TLS message or transported PA-TNC attribute to a desired value to ease the compromise of an endpoint. Without the ability for message recipients to detect whether a received message contains the same content as what was originally sent, active attackers can stealthily modify the attribute exchange.

The PT-TLS protocol leverages the TLS protocol to provide strong authentication and integrity protections as a countermeasure to this theat. The bi-directional authentication prevents the attacker from acting as an active man-in-the-middle to the protocol that could be used to modify the message exchange. The strong integrity protections (e.g. hashing) offered by TLS allows PT-TLS message recipients to detect message alterations by other types of network based adversaries.

4.2.4. Denial of Service

A variety of types of denial of service attacks are possible against the PT-TLS protocol if the message exchanges are left unprotected while traveling over the network. The Posture Transport Client and Posture Transport Server are trusted not to participate in the denial of service of the assessment session, leaving the threats to come from the network.

The PT-TLS protocol provides bi-directional authentication capabilities in order to prevent a man-in-the-middle on the network from becoming an undetected active proxy of PT-TLS messages. Because the PT-TLS protocol runs after the TLS handshake and thus cipher establishment/use, all of the PT-TLC messages are protected from undetected modification that could create a denial of service situation. However it is possible for an adversary to alter the message flows causing each message to be rejected by the recipient because it fails the integrity checking.

The PT-TLS protocol operates as an application protocol on top of TLS and thus TCP/IP protocols, so is subject to denial of service attacks against the TLS, TCP and IP protocols.

4.2.5. NEA Asokan Attacks

As described in section 3.3. and in the NEA Asokan Attack Analysis [ASOKAN], a sophisticated MITM attack can be mounted against NEA systems. The attacker forwards PA-TNC messages from a healthy machine through an unhealthy one so that the unhealthy machine can

gain network access. Section 3.3. and the NEA Asokan Attack Analysis provide a detailed description of this attack and of the countermeasures that can be employed against it.

Because lying endpoint attacks are much easier than Asokan attacks and the only known effective countermeasure against lying endpoint attacks is the use of an External Measurement Agent (EMA), countermeasures against an Asokan attack are not necessary unless an EMA is in use. However, PT-TLS implementers may not know whether an EMA will be used with their implementation. Therefore, PT-TLS implementers SHOULD support the Asokan attack countermeasures by providing the value of the tls-unique channel binding to higher layers in the NEA reference model: Posture Broker Clients, Posture Broker Servers, Posture Collectors, and Posture Validators.

5. Privacy Considerations

The role of PT-TLS is to act as a secure transport for PB-TNC and other higher layer protocols. As such, PT-TLS does not directly utilize personally identifiable information (PII) except when client authentication is enabled. When client authentication is being used, the NEA Client will be asked to disclose a local identifier (e.g. username) associated with the endpoint and an authenticator (e.g. password) to authenticate that identity. Because the identity and authenticator are potentially privacy sensitive information, the NEA Client MUST offer a mechanism to restrict which NEA Servers will be sent this information. Similarly, the NEA Client should provide an indication to the person being identified that a request for their identity has been made in case they choose to opt out of the authentication to remain anonymous.

PT-TLS provides cryptographic peer authentication, message integrity and data confidentiality protections to higher layer NEA protocols that may exchange data potentially including PII. These security services can be used to protect any PII involved in an assessment from passive and active attackers on the network. Endpoints sending potentially privacy sensitive information should ensure that the PT-TLS security protections (TLS cipher suites) negotiated for an assessment of the endpoint are adequate to avoid interception and off-line attacks of any long term privacy sensitive information.

6. IANA Considerations

This section defines the contents of three new IANA registries: PT-TLS Message Types, PT-TLS Auth Types, and PT-TLS Error Codes. This section explains how these registries work.

All of the registries defined in this document support IETF standard values and vendor-defined values. To explain this phenomenon, we will use the PT-TLS Message Type as an example but the other registries work the same way.

Whenever a PT-TLS Message Type appears on a network, it is always accompanied by an SMI Private Enterprise Number (PEN), also known as a vendor ID. If this vendor ID is zero, the accompanying PT-TLS Message Type is an IETF standard value listed in the IANA registry for PT-TLS Message Types and its meaning is defined in the specification listed for that PT-TLS Message Type in that registry. If the vendor ID is not zero, the meaning of the PT-TLS Message Type is defined by the vendor identified by the vendor ID (as listed in the IANA registry for SMI PENs). The identified vendor is encouraged but not required to register with IANA some or all of the PT-TLS Message Types used with their vendor ID and publish a specification for each of these values.

This delegation of namespace is analogous to the technique used for OIDs. It can result in interoperability problems if vendors require support for particular vendor-specific values. However, such behavior is explicitly prohibited by this specification, which dictates that "Posture Transport Clients and Posture Transport Servers MUST NOT require support for particular vendor-specific PT-TLS Error Codes and MUST interoperate with other parties despite any differences in the set of vendor-specific PT-TLS Error Codes supported (although they MAY permit administrators to configure them to require support for specific PT-TLS error codes)." Similar requirements are included for PT-TLS Message Types and PT-TLS Auth Types.

6.1. Designated Expert Guidelines

For all of the IANA registries defined by this specification, new values are added to the registry by Expert Review with Specification Required, using the Designated Expert process defined in RFC 5226 [RFC5226].

This section provides guidance to designated experts so that they may make decisions using a philosophy appropriate for these registries.

The registries defined in this document have plenty of values. In most cases, the IETF has approximately 2^32 values available for it to define and each vendor has the same number of values for its use. Because there are so many values available,

designated experts should not be terribly concerned about exhausting the set of values.

Instead, designated experts should focus on the following requirements. All values in these IANA registries MUST be documented in a specification that is permanently and publicly available. IETF standard values MUST also be useful, not harmful to the Internet, and defined in a manner that is clear and likely to ensure interoperability.

Designated experts should encourage vendors to avoid defining similar but incompatible values and instead agree on a single IETF standard value. However, it is beneficial to document existing practice.

There are several ways to ensure that a specification is permanently and publicly available. It may be published as an RFC. Alternatively, it may be published in another manner that makes it freely available to anyone. However, in this latter case, the vendor MUST supply a copy to the IANA and authorize the IANA to archive this copy and make it freely available to all if at some point the document becomes no longer freely available to all through other channels.

The following three sections provide guidance to the IANA in creating and managing the new IANA registries defined by this specification.

6.2. Registry for PT-TLS Message Types

The name for this registry is "PT-TLS Message Types". Each entry in this registry should include a human-readable name, an SMI Private Enterprise Number, a decimal integer value between 0 and 2^32-1, and a reference to the specification where the contents of this message type are defined. This specification must define the meaning of the PT-TLS message type and the format and semantics of the PT-TLS Message Value field that include the designated Private Enterprise Number in the PT-TLS Message Type Vendor ID field and the designated numeric value in the PT-TLS Message Type field.

The following entries for this registry are defined in this document. Once this document becomes an RFC, they should become the initial entries in the registry for PT-TLS Message Types. Additional entries to this registry are added by Expert Review with Specification Required, following the guidelines in section 6.1.

PEN	Value	Name	Defining Specification
0	0	Experimental	RFC # Assigned to this I-D
0	1	Version Request	RFC # Assigned to this I-D
0	2	Version Response	RFC # Assigned to this I-D
0	3	Client Auth Request	RFC # Assigned to this I-D
0	4	Client Auth Selection	RFC # Assigned to this I-D
0	5	Client Auth Challenge	RFC # Assigned to this I-D
0	6	Client Auth Response	RFC # Assigned to this I-D
0	7	Client Auth Success	RFC # Assigned to this I-D
0	8	PT-TLS Batch	RFC # Assigned to this I-D
0	9	Reserved	RFC # Assigned to this I-D
0	10	Reserved	RFC # Assigned to this I-D
0	11	PT-TLS Error	RFC # Assigned to this I-D
0	12	Reserved	RFC # Assigned to this I-D
0	0xffffffff	Reserved	RFC # Assigned to this I-D

6.3. Registry for PT-TLS Error Codes

The name for this registry is "PT-TLS Error Codes". Each entry in this registry should include a human-readable name, an SMI Private Enterprise Number, a decimal integer value between 0 and 2^32-1, and a reference to the specification where this error code is defined. This specification must define the meaning of this error code and the format and semantics of the Error Information field for PT-TLS messages that have a PT-TLS Vendor ID of 0, a PT-TLS Message Type of PT-TLS Error, the designated Private Enterprise Number in the PT-TLS Error Code Vendor ID field, and the designated numeric value in the PT-TLS Error Code field.

The following entries for this registry are defined in this document. Once this document becomes an RFC, they should become the initial entries in the registry for PT-TLS Error Codes. Additional entries to this registry are added by Expert Review with Specification Required, following the guidelines in section 6.1.

PEN	Value	Name	Defining Specification
0	0	Reserved	RFC # Assigned to this I-D
0	1	Malformed Message	RFC # Assigned to this I-D
0	2	Version Not Supported	RFC # Assigned to this I-D
0	3	Type Not Supported	RFC # Assigned to this I-D
0	4	Failed Authentication	RFC # Assigned to this I-D
0	5	Invalid Message Error	RFC # Assigned to this I-D
0	6	Authentication Error	RFC # Assigned to this I-D

6.4. Registry for PT-TLS Auth Types

The name for this registry is "PT-TLS Auth Types". Each entry in this registry should include a human-readable name, an SMI Private Enterprise Number, a decimal integer value between 0 and 255, and a reference to the specification where this authentication type is defined. This specification must define the defined authentication mechanism including the format and semantics of the Authentication Information and Challenge Information fields for PT-TLS client authentication message exchange described in section 3.8.

The following entries for this registry are defined in this document. Once this document becomes an RFC, they should become the initial entries in the registry for PT-TLS Auth Types. Additional entries to this registry are added by Expert Review with Specification Required, following the guidelines in section 6.1.

PEN	Value	Name	Defining Specification
0	0	Experimental	RFC # Assigned to this I-D
0	1	Basic Auth	RFC # Assigned to this I-D

7. Acknowledgments

The author of this draft would also like to acknowledge the following people who have contributed to or provided substantial input on the preparation of this document or predecessors to it: Stuart Bailey, Lauren Giroux, Steve Hanna, Josh Howlett, Scott Kelly, Sung Lee, Lisa Lorenzin, Ravi Sahita, and Mark Townsend.

This document was prepared using 2-Word-v2.0.template.dot.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2617] Franks, J., P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

[RFC3629] Yergeau F., "UTF-8, a transformation format of ISO 10646", RFC 3629, November 2003.

[RFC4346] Dierks T., Rescorla E., "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

[RFC5226] Narten T., Alvestrand H., "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, May 2008.

[RFC5246] Dierks T., Rescorla E., "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5792] Sangster P., Narayan K., "PA-TNC: A Posture Attribute Protocol (PA) Compatible with TNC", RFC 5792, March 2010.

[RFC5793] Sahita, R., Hanna, S., and R. Hurst, "PB-TNC: A Posture Broker Protocol (PB) Compatible with TNC", RFC 5793, March 2010.

8.2. Informative References

[ASOKAN] Salowey, J., Hanna, S., "NEA Asokan Attack Analysis", draft-salowey-nea-asokan-00.txt (work in progress), October 2010.

[IFT-TLS] Trusted Computing Group, "TNC IF-T: Binding to TLS", http://www.trustedcomputinggroup.org/files/resource_files/51F0757E-1D09-3519-AD63B6FD099658A6/TNC_IFT_TLS_v1_0_r16.pdf, May 2009.

[PT-EAP] Hanna, S., Sangster, P., "PT-EAP: Posture Transport (PT) Protocol For EAP Tunnel Methods", draft-hanna-nea-pt-eap-01.txt (work in progress), March 2011.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

[RFC5209] Sangster, P., Khosravi, H., Mani, M., Narayan, K., and J. Tardo, "Network Endpoint Assessment (NEA): Overview and Requirements", RFC 5209, June 2008.

[RFC5929] Altman, J., Williams, N., Zhu L., "Channel Bindings for TLS", RFC 5929, July 2010.

Appendix A. Evaluation Against NEA Requirements

This section evaluates the PT-TLS protocol against the PT requirements defined in the NEA Overview and Requirements and PB-TNC specifications. Each subsection considers a separate requirement and highlights how PT-TLS meets the requirement.

A.1. Evaluation Against Requirement C-1

Requirement C-1 says:

C-1 NEA protocols MUST support multiple round trips between the NEA Client and NEA Server in a single assessment.

PT-TLS meets this requirement. Use of the TLS protocol over TCP/IP allows for multiple round trips of PT-TLS messages, which can carry multiple round trips of PB-TNC batches.

A.2. Evaluation Against Requirements C-2

Requirement C-2 says:

C-2 NEA protocols SHOULD provide a way for both the NEA Client and the NEA Server to initiate a posture assessment or reassessment as needed.

PT-TLS meets this requirement. PT-TLS allows the NEA Client or the NEA Server to initiate a posture assessment or reassessment.

A.3. Evaluation Against Requirements C-3

Requirement C-3 says:

C-3 NEA protocols including security capabilities MUST be capable of protecting against active and passive attacks by intermediaries and endpoints including prevention from replay based attacks.

PT-TLS meets this requirement. The use of TLS provides strong cryptographic authentication, integrity and confidentiality services for the NEA protocols.

A.4. Evaluation Against Requirements C-4

Requirement C-4 says:

C-4 The PA and PB protocols MUST be capable of operating over any PT protocol. For example, the PB protocol must provide a transport independent interface allowing the PA protocol to operate without change across a variety of network protocol environments (e.g. EAP/802.1X, PANA, TLS and IKE/IPsec).

While this requirement is not applicable to PT, the PT-TLS protocol is independent of PA and PB allowing those protocols to operate over other PT protocols.

A.5. Evaluation Against Requirements C-5

Requirement C-5 says:

C-5 The selection process for NEA protocols MUST evaluate and prefer the reuse of existing open standards that meet the requirements before defining new ones. The goal of NEA is not to create additional alternative protocols where acceptable solutions already exist.

Based on this requirement, PT-TLS should receive a strong preference. PT-TLS is equivalent with IF-T Binding to TLS 1.0, an open TCG specification. Selecting PT-TLS as the basis for the PT protocol will ensure compatibility with IF-T Binding to TLS, and with its implementations.

A.6. Evaluation Against Requirements C-6

Requirement C-6 says:

C-6 NEA protocols MUST be highly scalable; the protocols MUST support many Posture Collectors on a large number of NEA Clients to be assessed by numerous Posture Validators residing on multiple NEA Servers.

PT-TLS meets this requirement. The PT-TLS protocol is independent of the quantity or size of the PA-TNC messages and the number of Posture Collectors and Posture Validators. PT-TLS provides the Posture Broker Client and Posture Broker Server a transport capable of carrying PT-TNC batches up to 2^32-16 octets in length. Posture Broker Clients and Posture Broker Servers wishing to send a PB-TNC batch longer than 2^32-16 octets could opt to split up set of attributes into multiple PB-TNC batches and send them sequentially since PT-TLS is full duplex.

The fields present in the PT-TLS protocol are also very scalable, allowing for the definition of a large (2^32) number of IETF standard and vendor-defined PT-TLS message types and message identifiers.

A.7. Evaluation Against Requirements C-7

Requirement C-7 says:

C-7 The protocols MUST support efficient transport of a large number of attribute messages between the NEA Client and the NEA Server.

PT-TLS meets this requirement. PT-TLS will allow for transport of a very large number of attributes leveraging the underlying TCP/IP network access. The PT-TLS protocol only adds 16 octets of overhead per PT-TLS message, which is negligible since a single PT-TLS message might carry very many PA-TNC attributes within a single PB-TNC batch.

A.8. Evaluation Against Requirements C-8

Requirement C-8 says:

C-8 NEA protocols MUST operate efficiently over low bandwidth or high latency links.

PT-TLS protocols meet this requirement. TLS will operate well over high latency or low bandwidth links leveraging TCP's ability to adjust to the underlying network carrier. The NEA protocols encapsulated by the PT-TLS protocol are designed to be able to operate over EAP with long RADIUS proxy chains so they can adapt to high latency or low bandwidth links. With the small amount of overhead added by PT-TLS, TLS, and TCP/IP, these protocols should still be efficient over high latency or low bandwidth networks.

A.9. Evaluation Against Requirements C-9

Requirement C-9 says:

C-9 For any strings intended for display to a user, the protocols MUST support adapting these strings to the user's language preferences.

PT-TLS meets this requirement. The PT-TLS protocol does not include messages intended for display to the user.

A.10. Evaluation Against Requirements C-10

Requirement C-10 says:

C-10 NEA protocols MUST support encoding of strings in UTF-8 format.

PT-TLS meets this requirement. All strings in the PT-TLS protocol are encoded in UTF-8 format. This allows the protocol to support a wide range of languages efficiently.

A.11. Evaluation Against Requirements C-11

Requirement C-11 says:

C-11 Due to the potentially different transport characteristics provided by the underlying candidate PT protocols, the NEA Client and NEA Server MUST be capable of becoming aware of and adapting to the limitations of the available PT protocol. For example, some PT protocol characteristics that might impact the operation of PA and PB include restrictions on: which end can initiate a NEA connection, maximum data size in a message or full assessment, upper bound on number of roundtrips, and ordering (duplex) of messages exchanged. The selection process for the PT protocols MUST consider the limitations the candidate PT protocol would impose upon the PA and PB protocols.

PT-TLS meets this requirement. The PT-TLS protocol leverages the underlying TLS connection to offer a reliable, full duplex session capable of being initiated by the NEA Client or NEA Server. This TLS session allows for transmission of large PB-TNC batches with many roundtrips with very low overhead (only 16 octets of protocol overhead per PT-TLS message).

A.12. Evaluation Against Requirements PT-1

Requirement PT-1 says:

PT-1 The PT protocol MUST NOT interpret the contents of PB messages being transported, i.e., the data it is carrying must be opaque to it.

PT-TLS meets this requirement. The PT-TLS protocol encapsulates PB-TNC batches without interpreting their contents.

A.13. Evaluation Against Requirements PT-2

Requirement PT-2 says:

PT-2 The PT protocol MUST be capable of supporting mutual authentication, integrity, confidentiality, and replay protection of the PB messages between the Posture Transport Client and the Posture Transport Server.

PT-TLS meets this requirement. The PT-TLS protocol leverages TLS to provide mutual authentication, integrity protection and confidentiality as well as replay protection. For more information see the Security Considerations section 4.

A.14. Evaluation Against Requirements PT-3

Requirement PT-3 says:

PT-3 The PT protocol MUST provide reliable delivery for the PB protocol. This includes the ability to perform fragmentation and reassembly, detect duplicates, and reorder to provide in-sequence delivery, as required.

PT-TLS meets this requirement. The PT-TLS protocol operates over TCP/IP which provides fragmentation/reassembly services and can detect/discard duplicate message and re-order messages if they arrive out of order over the network. PT-TLS provides a reliable, in-order delivery NEA message transport to the Posture Broker Client and Posture Broker Server components.

A.15. Evaluation Against Requirements PT-4

Requirement PT-4 says:

PT-4 The PT protocol SHOULD be able to run over existing network access protocols such as 802.1X and IKEv2.

PT-TLS does NOT meet this requirement as it's intended for a different usage. PT-TLS protocol requires the use of a TCP/IP connection to the network. PT-EAP (PT Binding to EAP Tunnel Methods) meets this requirement. PT-TLS is intended to be used after the endpoint has been admitted to the network.

A.16. Evaluation Against Requirements PT-5

Requirement PT-5 says:

PT-5 The PT protocol SHOULD be able to run between a NEA Client and NEA Server over TCP or UDP (similar to Lightweight Directory Access Protocol (LDAP)).

PT-TLS meets this requirement. The PT-TLS protocol operates on top of an existing TCP/IP connection using TLS for network security.

A.17. Evaluation Against Requirements PT-6 (from PB-TNC specification)

Requirement PT-6 says:

PT-6 The PT protocol MUST be connection oriented; it MUST support confirmed initiation and close down.

PT-TLS meets this requirement. The PT-TLS protocol operates on top of an existing TCP/IP connection which is connection oriented and supports confirmed initiation and tear down of the connection.

A.18. Evaluation Against Requirements PT-7 (from PB-TNC specification)

Requirement PT-7 says:

PT-7 The PT protocol MUST be able to carry binary data.

PT-TLS meets this requirement. The PT-TLS protocol is capable of carrying binary data.

A.19. Evaluation Against Requirements PT-8 (from PB-TNC specification)

Requirement PT-8 says:

PT-8 The PT protocol MUST provide mechanisms for flow control and congestion control.

PT-TLS meets this requirement. The PT-TLS protocol operates on top of TCP/IP which provides flow and congestion control.

A.20. Evaluation Against Requirements PT-9 (from PB-TNC specification)

Requirement PT-9 says:

PT-9 PT protocol specifications MUST describe the capabilities that they provide for and limitations that they impose on the PB protocol (e.g. half/full duplex, maximum message size).

PT-TLS meets this requirement. This specification discusses the level of transport service provided to the Posture Broker Client and Posture Broker Server. Generally, the PT-TLS protocol supports the post network admission usages discussed in RFC 5209. The maximum message size for PT-TLS is only 16 octets less then the maximum message size allowable by PB-TNC.

Authors' Addresses

Paul Sangster
Symantec Corporation
6825 Citrine Dr
Carlsbad, CA 92009

Email: paul_sangster@symantec.com