The Direct Project

XDR and XDM for Direct Messaging Specification

Version 1.0

finalized 9 March 2011

Contents

Status of this Specification

This specification is Final

IPR Statement

By contributing to this specification, all contributors warrant that all applicable patient or other intellectual policy rights have been disclosed and that any of which contributors are aware of will be disclosed in accordance with the Direct Project IPR Policy.

Requirements

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.

An implementation is not compliant if it fails to satisfy one or more of the MUST or REQUIRED level requirements for the protocols it implements. An implementation that satisfies all the MUST or REQUIRED level and all the SHOULD level requirements for its protocols is said to be "unconditionally compliant"; one that satisfies all the MUST level requirements but not all the SHOULD level requirements for its protocols is said to be "conditionally compliant.

1.0 Introduction

This specification addresses use of XDR and XDM zipped packages in e-mail in the context of directed messaging to fulfill the key user stories of the Direct Project. Note that while the XDM specification includes transport options for USB-Memory and CD-ROM, in this specification XDM always means the XDM e-mail transport option (i.e., XDM file system specification in a zip package as an S/MIME attachment).

This specification defines:

1. Use of XD* Metadata with XDR and XDM in the context of directed messaging
2. Additional attributes for XDR and XDM in the context of directed messaging
3. Issues of conversion when endpoints using IHE XDR or XDM specifications interact with endpoints utilizing SMTP for delivering healthcare content.

The Direct Project has identified the use of SMTP as its primary mechanism for delivering healthcare content from a sender to a receiver. This choice supports the environments which have minimal capabilities in terms of using Web Services and generating detailed metadata. In the healthcare ecosystem there are several existing environments which have adopted the use of SOAP-based Web Services and detailed metadata. These environments have adopted a family of IHE profiles, each applied to a different type of use case, which have a common metadata model and make use of Web Services in a common way. The most applicable IHE profiles to the Direct Project environment are:

- XDR which supports a direct push model from sender to receiver using Web Services transport
- XDM which supports a direct push model of a package of content where one of several optional transports is via SMTP

This specification discusses the application of XDR and XDM to the direct messaging environment and the interaction between the primary Direct Project environment, which uses SMTP and RFC 5322 to transport and encode healthcare content, and the XDR and XDM specifications.

In applying XDR and XDM to the direct messaging environment there are modifications to the base IHE standard deemed necessary to support two primary purposes:

- To address the security recommendation from the HIT Standards committee, to separate addressing metadata from content metadata. When this specification is implemented, a HISP will not have to open a content package and possibly be exposed to PHI, simply to find the intended recipient
- To modify the strict XD* metadata requirements that may be excessive for simple push use cases among known recipients. See the "Minimal Metadata Definition" section of this specification for details.

2.0 Conversions Process Flow

The primary user stories for Conversions are the same as those for the Direct Project as a whole except that the content flows among sources and destinations which differ in their use of metadata. In other words, in any user story, it is possible that either the source or the destination, but not both, are capable of communicating using IHE XDR. One simple example is that of a physician practice with an e-mail client (or EHR module that does not speak XDR or XDM) sending a referral message with an attachment (such as a PDF file) to another practice or hospital that prefers to receive via XDR. These are Direct Project User Stories 1 and 2.

So the process flows relevant to this specification are:

- XDR Source sends message to non-XDR Destination
- Non-XDR Source sends message to XDR Destination

Both of which are discussed in detail in the following sections.

2.1 XDR Source to Non-XDR Destination

When messages flow from an XDR source to a non-XDR destination the transformation is straightforward since in this case there is more metadata than the destination expects. The transformation of the message does not have to generate any metadata, since it is already present per XDR. Rather, a transformation is done by creating an XDM package (using the XDR metadata), as an attachment to a Direct Project-compliant secure message, and transmit it using SMTP to the destination. User Stories 7 and 8 (Provider or Hospital send health information to patient) may be good examples of this flow. This specification gives guidance to this transformation.

2.2 Non-XDR Source to XDR Destination

The more complicated case is to handle the case of direct messaging between a **non-XDR source** (such as an e-mail client or an EHR without the capabilities to generate XDR transactions) and an **XDR destination** (such as an EHR or HIO). If an XDM package was sent by the non-XDR source, conversion to XDR is a matter of re-packaging. In other cases, it is difficult or impossible for the source to provide all the metadata required in the IHE ITI specification. Often, the source will create a simple e-mail message with an attachment, and its HISP should transform the message into an XDR transaction that can be consumed by the destination. In doing the transformation, it will value the metadata where known, per the **Metadata Definition** section of this specification. However, it is not appropriate for a HISP to create default metadata values (just to fill in all required fields) if they are misleading or assert something that is not known to be true. For example, defaulting that confidentialityCode is "normal" or any other value, if it is in fact unknown, is inappropriate, and it is better to leave the metadata unvalued. Similarly, defaulting a "pseudo" patientID that might be confused with a real patientID is also inappropriate.

3.0 Interaction Patterns

This specification focuses on two IHE specifications (XDM & XDR) and how they interact with the Direct Project SMTP/RFC 5322 specification.

The following cases are considered:

1. RFC 5322 + MIME: In this case, the message is transported via SMTP, and the content is a MIME (possibly multipart) body to an RFC 5322 document and the content does not conform to IHE requirements for an XDM e-mail

2. RFC 5322 + XDM: In this case, the message is transported via SMTP, the content conforms to the IHE requirements for an XDM e-mail
3. SOAP + XDR: In this case, the message is transported via SOAP over HTTP and the content is XDR (MTOM-encoded documents in an XD* Metadata package)

The following table shows the cases of conversion that SHALL be performed.

		Receivers		
		RFC5322 + MIME	RFC 5322 + XDM	SOAP + XDR
Senders	RFC 5322 + MIME	No Conversion	No conversion - receiver expected to be able to use non-XDM format	- Transport Conversion - Metadata is created
	RFC 5322 + XDM	No Conversion - receiver is expected to be able to handle XDM package	No conversion	- Transport conversion - metadata simply transformed
	SOAP + XDR	- Transport conversion - metadata is simply transformed - delivered as XDM package	- Transport conversion - metadata is simply transformed - delivered as XDM package	No conversion

Note that in both case (1) and (2) the receiver is expected to support the reception of both type (1) and (2) content. Where the reception of (2) does require that a system have the ability to open a ZIP file, which is built into most modern operating systems today, and where the XDM format does require that an index.htm is available to allow access to the content from a simple web browser.

Transport Conversion details how to map between SMTP and RFC 5322 header constructs to semantically identical constructs in SOAP, and vice versa

Packaging Conversion details how to transform an XDM package to the equivalent SOAP XDR package and vice versa

Metadata Conversion details how to create XD* Metadata from an RFC 5322 document.

The subsequent sections detail requirements forTransport, Metadata, and Packaging, including requirements for all XDR-based Direct transactions and requirements for conversions.

4.0 Transport Requirements

4.1 SOAP headers in support of addressing

This section is not specific to conversion, and applies to use of XDR for Direct-compliant messaging.

In cases where an intermediary performs relay functions but does not need to view or examine content or content metadata, the origination and destination addresses should to be carried outside of the contents of the SOAP container to support minimization of PHI access. In these cases the origination point will replicate the origination and destination addresses into the appropriate SOAP headers:

- from - is specified in the SOAP Header using the direct:from element and contains a value conformant to the anyURI type. For example: <direct:from>direct@direct.org</direct:from>
- to - is specified in the SOAP Header using the direct:to element and contains a value conformant to the anyURI type. For example: <direct:to>direct@direct.org</direct:to>

Note that the message identifier is carried within the WS Addressing MessageID header, for example <wsa:MessageID>uuid:db00ed94-951b-4d47-8e86-585b31fe01bf@nhin.sunnyfamilypractice.example.org</wsa:MessageID>.

An example of the SOAP header is as follows:

```
<direct:addressBlock xmlns:direct="urn:direct:addressing"
env:role="urn:direct:addressing:destination"
env:relay="true">
<direct:from>mailto:entity1@direct.example.org</direct:from>
<direct:to>mailto:entity2@direct.example.org</direct:to>
</direct:addressBlock>
```

4.2 Transport Conversions Overview

The transport conversions required are between SOAP and SMTP. There are two cases of this conversion, converting from SMTP to SOAP and vice versa. The conversion from SMTP to SOAP has two flavors, one where SMTP with only RFC 5322 applied and one where SMTP is carrying an XDM zip file. The conversion of transport is the same in these two cases.

4.3 Transport conversion from SMTP to SOAP

The conversion from SMTP to SOAP involves using the SMTP headers to identify the correct SOAP endpoint and setting the appropriate SOAP headers to assure proper transport and processing of the message.
The key headers for the purposes of conversion are:

1) Addressing lists
2) Date header
3) Message-ID

Addressing lists MUST be taken from SMTP TO and RCPT FROM SMTP commands. Addressing lists are used to populate SOAP headers in the resulting SOAP message (see next section).

Implementations MUST be prepared to handle multiple receivers in a single SMTP transaction, including multiple receivers at the same or different organizations.

Each address identified in the SMTP headers MUST be converted to a Web Services address and an intended Recipient value. The Web Services address is used to identifier the SOAP endput for the XDR message. The intended Recipient MUST be set in at least the intended Recipient value of the metadata, see section 6.3.2, and MAY also be set in the SOAP header as the content of a direct:to element, see section 4.1. If more than one address is mapped to the same Web Services address the sender may sent a single XDR message for both recipients, setting both values in the intended Recipient field.

Message-ID MUST populate the MessageID WS-Addressing header.

The Date header is used for package conversion, and must populate the submission Time XD* metadata attribute, see section 6.2.2.

Handling of other headers is unspecified.

4.4 Transport conversion from SOAP to SMTP

The conversion from SOAP to SMTP involves using the SOAP headers and/or XDR metadata to identify the correct SMTP server and setting the appropriate SMTP headers to assure proper transport and processing of the message.

Implementations will construct a new RFC 5322 message and send via SMTP. The SMTP TO and RCPT FROM commands MUST carry the recipients and sender of the transaction, which SHOULD be taken from the SOAP header values, if available, or the metadata `SubmissionSet.author` and `SubmissionSet.intendedRecipient` values, if SOAP headers are not available.

The to header of the RFC 5322 message MUST contain the addresses noted in the SubmissionSet.intendedRecipient metadata field.. The from header MUST contain the adresses noted in SubmissionSet.author field. The Date field MUST contain the value of the submissionTime XD* metadata Submission Set attribute. The Message-ID header SHOULD contain the value of the WS-Addressing MessageID if it is an appropriate value for the Message-ID header.

The use of other RFC 5322 headers is unspecified.

5.0 Packaging Conversion

5.1 Packaging Conversion from RFC 5322 to XDR

Implementations will extract a source document and construct a single Document Entry for each part of the RFC 5322 message.

For messages originating from e-mail systems, the first text part of a multipart/mixed or the text portion of a multipart/alternative represents by convention the simple text of the e-mail message itself. In such cases, implementations SHOULD use a classCode of 56444-3 (Healthcare Communication). Note that this LOINC code is not defined in C80 table 2-144.

Implementations MUST Base64 Decode a Base64 Encoded source document, if a Base64 Encoded source document is provided (because MTOM-XOP performs binary encoding, and because XDR implementations expect to receive the actual source document after SOAP+MTOM processing, failure to decode the document will lead both to document bloat and processing problems at the receiver).

Implementations will construct a single Submission Set with one Association for each Document Entry between Document Entry and the the Submission Set. Implementations will provide values for metadata attributes for the Submission Set and Document Entry(ies) following the guidelines provided in Section 6.0, Metadata Conversions.

Implementations will package the MTOM-encoded SOAP transaction from the Document Entry(ies), the Submission Set, Associations, and source documents according to the rules of XDR.

5.2 Packaging Conversion from XDM to XDR

The IHE specification specifies that the e-mail message has a subject that contains the string "XDM/1.0/DDM", and specifies the format and contents of the XDM zip archive, but does not specify how the XDM zip archive or archives are attached to the e-mail message. This may lead to situations where there is a combination of XDM and non-XDM content, where there are multiple XDM attachments to an e-mail, or where there is a combination of XDM zip and non-XDM zip attachments. Accordingly, the following advice is given to implementations:

If the subject line contains the substring "XDM/1.0/DDM", implementations MUST identify all zip content parts in the e-mail message. Such content parts will generally have a content-type of `application/zip`. If there are several such attachments, implementations MUST test each such content part for compliance with XDM. If no XDM content parts exist, the implementation MAY error or MAY process as an RFC 5322 message as documented above. Implementations MUST process all XDM content parts and MAY assume that content outside of the XDM attachment or attachments does not need to be converted.

An XDM Zip package contains a root directory, which includes a set of manifest content, and an IHE_XDM directory, which contains a set of subdirectories, each of which contain packaged context with a metadata file equivalent to that provided in a XDR transaction.

Implementations MUST construct a single XDR transaction for each XDM subdirectory containing a metadata file.

Implementations MAY ignore all content that is not specified in the metadata file (including the index manifest files, and other content that is outside of the IHE_XDM directory. Implementations MAY add the full XDM ZIP file to the submission set for each XDR transaction constructed from the XDM file.

Implementations SHOULD follow the provided steps to perform the package conversion for each directory in the IHE_XDM directory:

1. Read and interpret the "METADATA.XML" file (containing an XML document with a single SubmitObjectsRequest root element)
2. Locate all Document Entries, and correlate the URI property of the document entry to the appropriate source document, re-constructing multipart documents from subdirectories. (That is, if the mimeType attribute is `multipart/foo`, construct a MIME multipart document, where the parts are suitably encoded representations of the source documents found in the multipart XDM folder)
3. As the URI property no longer refers to the XDM packaged file, it SHOULD be removed or supplied with a value that makes sense in an XDR context.

Package the XD* XML file and the source documents as an XDR transaction.

5.3 Packaging Conversion from XDR to XDM

Each XDR transaction MUST be repackaged as a single XDM zipped file.

Implementations SHOULD follow the provided steps to perform the package conversion:

1. Extract the documents and SubmitObjectsRequest XML element
2. Construct an INDEX.HTM and README.TXT file. the INDEX.HTM file SHOULD list all documents with "file://" URIs pointing to the files as placed in the directories. The README.TXT file MUST follow XDM guidelines
3. Construct an IHE_XDM directory, with a single subdirectory (named anything valid for XDM) to contain the XDR contents
4. For each source document, create an appropriate unique file name, and place in the directory, repackaging multipart documents as folders as specified in the XDM specification
5. For each source document, add a URI element for the associated Document Entry (ExtrinsicObject) element, pointing to the appropriate file
6. Serialize the resulting SubmitObjectsRequest as a valid XML document entitled METADATA.XML
7. Package the whole structure as a ZIP file

6.0 Metadata

In XDM and XDR, metadata is used to describe the content of the message to enable automatic routing and integration into recievers eletronic medical system. The metadata used by XDR and XDM was designed in the context of the XDS environment, a query/retrieve model, and requires adjustment for use in a directed exchange. This section describes this adjustment in terms of:

- Levels of conformance to XDS metadata requirements
- Metadata Requirements and Conformance
- Special Considerations
- Conversion

6.1 Levels of conformance to XDS metadata requirements

There SHALL be two levels of XD* metadata compliance for the purposes of the use of XDR and XDM in a directed exchange:

- Full XDS Metadata - requires the same level of optional/required conformanc as specified in IHE ITI TF Rev7 V3, section 4.1.

- Minimal Metadata, as specified in section 6.1.2 *Minimal Metadata Definition*

When converting into XDR with minimal metadata, the conversion process SHALL create as much of the appropriate metadata as possible. If all applicable XDR metadata as currently dictated by the specification is available this means the conversion process is compliant with the "Full XDS Metdata" level. If less than that level of metadata is available then the conversion process is compliant with the "Minimal Metadata" level.

Metadata conversion is required when converting from a transport with minimal metadata, RFC 5322 without an XDM attachment, to a transport requiring more significant metadata. Conversion in the other direction retains all the metadata available by coding the content in an XDM package where the receiver can ignore the metadata if preferred.

6.1.1 SOAP Headers

This section provides a SOAP header element that is used to communicate to receiving systems the level of metadata used. Example uses of this header include triggering rejection of the message (for organizations that wish only to receive full metadata) or triggering specific downstream processing.

If minimal metadata is being sent, an implementation MUST include the metadata-level header with content of "minimal". An implementation that sends full metadata is RECOMMENDED to include the metadata-level header. If the metadata-level SOAP element is not specified the receiver MUST assume conformance with full XDS Metadata.

To indicate the use of minimal metadata an implementation specifies metadata-level SOAP element as follows:

```
<direct:metadata-level>minimal</direct:metadata-level>
```

To indicate the use of full XDS metadata an implementation specifies the metadata-level SOAP element as follows:

```
<direct:metadata-level>XDS</direct:metadata-level>
```

6.1.2 Minimal Metadata Definition

The use of Minimal Metadata is required when converting from a transport with minimal metadata, RFC 5322 without an XDM attachment, to a transport requiring more significant metadata. Conversion in the other direction retains all the metadata available by coding the content in an XDM package where the receiver can ignore the metadata if prefered.
The use of minimal metadata covers the following cases:

- The system creating the XDR/XDM transaction does not have access to full XDS metadata, including cases where:
    - The original content was created with a system that does not store all relevant metadata items as discrete values (e.g., e-mail client sending a text message with PDF attachment)
    - The original content was received by the XDR creating system encrypted
- The system creating the XDR/XDM transaction is not able to or by policy prefers not to examine the content to construct available metadata
- The content payload does not conform to XDS Metadata expectations, including cases where:
    - The payload is not patient specific (e.g., summary level quality reporting)

It is possible that IHE will formalize levels of metadata conformance, and include additional levels of metadata conformance than the two proposed here. The minimal metadata definition proposed in this section may be deprecated in the future. Implementations are RECOMMENDED to implement metadata conformance in a way that is open to future modification.

The minimal metadata proposed here makes the following kinds of changes from the original XDS Metadata definitions:

1. Alterations of the `Source` specification (which controls optionality), in general moving from `R`, or "Required" to `R2`, or "Required if available". Implementations shall specified all R2 elements whenever the information is available.

6.2 Metadata Requirements and Conformance

This section lists all metadata that is expected to be applicable to a directed exchange.

This section lists the metadata that is mostly likely to be valued by implementations implementing XDR or XDM in a directed exchange environment. There is a few other metadata defined by IHE but not listed here because it is optional and not expected to be valued in this environment. Implementations are encouraged to consider that additional metadata if they find that they wish to encode further information in metadata.

6.2.1 Document Entry Metadata

This section lists the metadata associated with the content of the message (called document by IHE).

The following table lists each of the applicable metadata elements, the optionality specified in the IHE XDS specification and the adjusted optionality defined by the Minimal Metadata specification. The table also gives a few details regarding conformance of the value of the metadata element.

Metadata Attribute	**XDS Source**	**Minimal Metadata Source**	**Value Conformance**
author	R2	R2	If supplied, MUST indicate the document's author, which may be different from the message sender
classCode	R	R2	When available, implementations SHOULD draw values from HITSP C80, version 2.0.1, table 2-144
confidentialityCode	R	R2	When available, implementations SHOULD draw values from HITSP C80, version 2.0.1, table 2-150. Implementations SHOULD NOT use codes that reveal the specific trigger causes of confidentiality (e.g., ETH, HIV, PSY, SDV)
creationTime	R	R2	Implementations MUST NOT use transaction-related dates/times, including the value of the RFC 5322 Date header
entryUUID	R	R	MUST be a unique value internal to this transaction, MAY be a symbolic or UUID form as per the XDS Metadata specification
formatCode	R	R2	Implementations SHOULD draw values from HITSP C80, version 2.0.1, table 2-152, when the specific listed codes apply
healthcareFacilityTypeCode	R	R2	When available, implementations SHOULD draw values from HITSP C80, version 2.0.1, table 2-146. Implementations SHOULD populate mapped by configuration to sending organization
languageCode	R	R2	Coded identifiers as described by the IETF (Internet Engineering Task Force) RFC 3066, conformant with IHE requirements
mimeType	R	R	On conversion to/from MIME Entities, MUST contain the same media type as the applicable Content-Type header for

			the entity
patientId	R	R2	Formatted as a HL7 CX as described in ITI TF-3 Table 4.1-3.
practiceSettingCode	R	R2	When available, implementations SHOULD draw from HITSP C80, version 2.0.1, table 2-149 which is a list of members of the value set in table 2-148.
sourcePatientId	R	R2	Formatted as a HL7 CX as described in ITI TF-3 Table 4.1-3.
sourcePatientInfo	R2	R2	Formatted as defined in ITI TF-3 Table 4.1-5.
typeCode	R	R2	When available, implementations SHOULD draw values from HITSP C80, version 2.0.1, table 2-144 and SHOULD be the same value as classCode
uniqueId	R	R	Implementations SHOULD use a unique ID extracted from the content, if a single such value can be determined. If not, implementations SHOULD use a UUID URN, generated for the transaction. This value must be different from the uniqueId specified on the Submission Set.

6.2.2 Submission Set Metadata

This section lists the metadata associated with the set of content of the message (called submission set by IHE). Note that IHE allows multiple documents (content parts) and this set of metadata groups this set of documents and gives metadata that is common to all.
The following table lists each of the applicable metadata elements, the optionality specified in the IHE XDS specification and the adjusted optionality defined by the Minimal Metadata specification. The table also gives a few details regarding conformance of the value of the metadata element.

Attribute	**XDS Source**	**Minimal Metadata Source**	**Value Conformance**
author	R2	R	MUST indicate the message sender as a slot named "authorTelecommunication". See Extensions. When converted from an RFC 5322

			message, MUST indicate the value of the from header. Even though the authorPerson slot is required by IHE, since authorTelecommunication is valued the authorPerson may be omitted.
contentTypeCode	R	R2	When available, implementations SHOULD draw from HITSP C80, version 2.0.1, table 2-144
entryUUID	R	R	MUST be a unique value internal to this transaction, MAY be a symbolic or UUID form as per the XDS Metadata specification
intendedRecipient	O	R	MUST indicate the message receivers. When converted from RFC 5322, MUST carry the combined recipients. Implementations SHOULD handle bcc consistent with the relevant discussion in RFC 5322. See Extensions for how to carry the Direct Address.
patientId	R	R2	MUST be identical to the Document Entry patientId
sourceId	R	R	Implementations SHOULD use a UUID URN mapped by configuration to sending organization
submissionTime	R	R	In cases of transformation from RFC 5322, implementations SHOULD use the value of the Date header
title	O	O	It is RECOMMENDED that the Subject of the RFC 5322 message be put in this attribute
uniqueId	R	R	Implementations SHOULD use a unique ID extracted from the content, if a single such value can be determined. If not, implementations SHOULD use a UUID URN, generated for the transaction. This value must be different than the uniqueId specified on the Document.

6.3 Special Considerations and Extensions

6.3.1 Metadata Extensions to Submission Set Metadata

In order to code Direct Address elements in the metadata, extensions are defined for the author and intendedRecipient attributes.

Attribute	**XDS Source**	**Minimal Metadata Source**	**Additional constraints**
author subattribute: authorTelecommunication	N/A	R	MUST be a single valued slot, where the single slot value is an XTN data type string. See the example
intendedRecipient	O	R	Individual Value elements MUST contain a string of type XON|XCN|XTN, of which the XTN portion is required. See the example

6.3.2 Use of XTN

The HL7 datatype XTN is templated for use in metadata as follows:

```
XTN = "^^Internet^" direct-address
```

As with any HL7 datatype, the XTN value may have trailing delimiters ("^" characters).
See the examples section.

6.3.3 patientId, sourcePatientId, sourcePatientInfo

The metadata fields patientId, sourcePatientID and sourcePatientInfo all carry identifying information about the specific patient associated with the content. If any one of these contains a value, then the content shall be associated with a single patient and that patient will be identified by the content of the metadata attribute. In particular:

- patientId - this metadata field, if valued, shall contain an identifier known to the receiving system that uniquely identifies the patient whose content is being transmitted.
- sourcePatientId - this metadata field, if valued, shall contain an identifier know to the sending system that uniquely identifies the patient whose content is being transmitted.
- sourcePatientInfo - this metadata field, if valued, shall contain the demographics which identify the patient whose content is being transmitted.

If none of these fields are valued the receiver must use some other means to determine if the content is associated with a single patient, multiple patients or not patient specific. Other metadata may help with this determination, for example the formatCode may suggest that the content is a quality report so would not be specific to a single patient. In many cases the recipient will have to inspect the contents to determine the correct processing related to patient associations.

When none of the patient identifying fields is specified this might mean that the conversion of the message did not have any way to determine the type of content being converted so the recipient should not assume that there is non-patient specific content being received. Inspection of other metadata fields and the content is the only current mechanism available to determine the patient association when the patient identifying fields are not specified.

7.0 Security Considerations

These security considerations are based on the published threat model. The system that performs conversion to and from SMTP + S/MIME and XDR will be termed the Gateway for the purposes of these considerations.

The threats and associated security considerations applicable to XDR and to Secure Health Transport are documented in the respective specifications. This section deals only with the risks that are applicable to the combined conversion.

The Gateway is a highly trusted component, and ensuring the trust of the Gateway is the main threat mitigation. An attacker who compromises part or all of the Gateway system, either by gaining direct access to the Gateway or to the operating system on which the Gateway runs, or by causing untrusted code to be injected into the Gateway could perform a number of attacks, not limited to:

- Traffic analysis
- Access to unencrypted messages
- Ability to spoof communication
- Ability to modify communication

The Gateway must be subject to security audit and remediation and secured from known attacks using standard and well-known security mechanisms. Access to the Gateway must be appropriately controlled, and the risk of inappropriate use of the Gateway by personnel with access to the Gateway must be mitigated through mechanisms such as security training, audit, etc.

Some risks of access to unencrypted messages may be mitigated by storing data at rest in an encrypted state (but note that access to memory or access to keys may defeat this mitigation).

Risks to modification of communication in flight may be mitigated by content-level signatures.

8.0 Examples

This section is non-normative.

authorTelecommunication

The following example shows the extended slot for the `author` classification:

```
<rim:Slot name="authorTelecommunication">
  <rim:ValueList>
     <rim:Value>^^Internet^drsmith@direct.example.org</rim:Val
ue>
  </rim:ValueList>
</rim:Slot>
```

Intended Recipient

The following example shows the extended metadata for `intendedRecipient`

```
<rim:Slot name="intendedRecipient">
  <rim:ValueList>
    <rim:Value>
Some
Hospital^^^^^^^^^1.2.3.4.5.6.7.8.9.1789.45|^Wel^Marcus^^^Dr^MD
|^^Internet^marcus.wel@direct.example.org
    </rim:Value>
    <rim:Value>
Some
Hospital^^^^^^^^^1.2.3.4.5.6.7.8.9.1789.45|^Al^Peter^^^Dr^MD|^
^Internet^peter.al@direct.example.org
    </rim:Value>
    <rim:Value>
|12345^John^Smith^^^Dr^MD|^^Internet^john.smith@direct.example
.com
    </rim:Value>
    <rim:Value>
Main
Hospital^^^^^^^^^1.2.3.4.5.6.7.8.9.1789.2364||^^Internet^mainh
ospital@direct.example.net
    </rim:Value>
  </rim:ValueList>
</rim:Slot>
```

Complete conversion from SMTP to SOAP/XDR:

This section illustrates the specification by providing some examples of conversion from SMTP to SOAP/XDR.

Example A

Given this simple, text message sent via SMTP (note only a subset of SMTP client commands shown):

```
MAIL FROM: <drjones@direct.sunnyfamily.example.org>
RCPT TO: <drsmith@direct.happyvalley.example.com>
DATA
MIME-Version: 1.0
From: Doctor Jones <drjones@direct.sunnyfamily.example.org>
Date: Thu, 11 Nov 2010 11:55:40 -0800
Message-ID:
<AANLkTikLULq=xbbxBFPEnbMwQFZmN6CrtT7pz2EmXPVK@mail.gmail.com>
Subject: Clinical data communication
To: drsmith@direct.happyvalley.example.com
Content-Type: text/plain; charset=ISO-8859-1

Dr. Jones,

I saw your lovely patient, Ms. Norah Jones, this past Thursday,
and found her in splendid health.

Hope all is well with you,
Dr. Smith
.
QUIT
```

The generated code might look like this: Example A

Example B

Given this message with C32 attachment:

```
MAIL FROM: <drsmith@direct.happyvalley.example.com>
RCPT TO: <drjones@direct.sunnyfamily.example.org>
DATA
MIME-Version: 1.0
From:  drsmith@direct.happyvalley.example.com
Date: Thu, 11 Nov 2010 11:53:50 -0800
Message-ID:
<AANLkTik0fF+3stN0favnbp8XKJuzHm43asg4N3n=dXRQ@mail.gmail.com>
Subject: Clinical data communication
To: Doctor Jones <drjones@direct.sunnyfamily.example.org>
Content-Type: multipart/mixed;
boundary=00163630f4cb65eece0494cc5690

--00163630f4cb65eece0494cc5690
Content-Type: text/plain; charset=ISO-8859-1

Dear Dr. Smith,

Attached, please find the clinical summary for the patient I am
referring to you.

Best regards,
Dr. Jones

--00163630f4cb65eece0494cc5690
Content-Type: text/xml; charset=US-ASCII;
name="HITSP_C32v2.5_Rev4_11Sections_Entries_MinimalErrors.xml"
Content-Disposition: attachment;
filename="HITSP_C32v2.5_Rev4_11Sections_Entries_MinimalErrors.x
ml"
Content-Transfer-Encoding: base64
X-Attachment-Id: f_gge23hgk0

Data removed for clarity
```

The generated code might look like this: Example B

The above generated code could be expanded by parsing the C32 document and extracting richer metadata for the XDR message.

Authors

Direct Project group focused on Direct Project (Arien Malec, Karen Witting, Vassil Peytchev, David Tao, Dragon, Vince Lewis, Beau Grantham, Chaminda Gunaratne, others...)

References

RFC 2119: Bradner, Key words for use in RFCs to Indicate Requirement Levels, RFC 2119

XDR: As of August 2010 the IHE XDR profile has been promoted to Final Text. Direct access to sections related to XDR are:

- XDR Introduction/Actors/Transactions - ITI TF Volume 1 Section 15 page 125
- XDR Transaction: Provide and Register Document Set - ITI TF Volume 2b Section 3.41 page 93
- XDR Integration with other IHE profiles - ITI TF Volume 1 Appendix E.7 & E.8 page 165

XDM: Direct access to sections related to XDM are:

- XDM Introduction/Actors/Transactions - ITI TF Volume 1 Section 16 page 128
- XDM Transaction: Distribute Document Set on Media - ITI TF Volume 2b Section 3.32 page 79
- Use of eMail -ITI TF Volume 2x Appendix T Use of eMail page 69
- XDR Integration with other IHE profiles - ITI TF Volume 1 Appendix E.7 & E.8 page 165

XDS Metadata Model: Both XDR and XDM use XDS Metadata to express information about the content being transmitted. Sections describing the XDS Metadata Model are:

- XDS Metadata - ITI TF Volume 3 Section 4.1 page 3
- Content and Format of XDS Documents - ITI TF Volume 1 Appendix J page 171
- XDS Concept Details - ITI TF Volume 1 Appendix K page 173

Copyright

By contributing to this specification, all contributors agree to license contributions according to the Creative Commons Attribution 3.0 License which is incorporated into this document by reference.