Integrating the Healthcare Enterprise

IHE IT Infrastructure Technical Framework Supplement

Document Encryption (DEN)

Rev. 1.3 – Trial Implementation

Date: July 24, 2018

Author: IT Infrastructure Technical Committee

Email: iti@ihe.net

Please verify you have the most recent version of this document. See here for Trial Implementation and Final Text versions and here for Public Comment versions.

Foreword

This is a supplement to the IHE IT Infrastructure Technical Framework V15.0. Each supplement undergoes a process of public comment and trial implementation before being incorporated into the volumes of the Technical Frameworks.

This supplement is published on July 24, 2018 for trial implementation and may be available for testing at subsequent IHE Connectathons. The supplement may be amended based on the results of testing. Following successful testing it will be incorporated into the IT Infrastructure Technical Framework. Comments are invited and may be submitted at http://www.ihe.net/ITI_Public_Comments.

This supplement describes changes to the existing technical framework documents.

"Boxed" instructions like the sample below indicate to the Volume Editor how to integrate the relevant section(s) into the relevant Technical Framework volume.

Amend Section X.X by the following:

Where the amendment adds text, make the added text **bold underline**. Where the amendment removes text, make the removed text **bold strikethrough**. When entire new sections are added, introduce with editor's instructions to "add new text" or similar, which for readability are not bolded or underlined.

General information about IHE can be found at: http://www.ihe.net.

Information about the IHE IT Infrastructure domain can be found at http://www.ihe.net/IHE_Domains.

Information about the structure of IHE Technical Frameworks and Supplements can be found at http://www.ihe.net/IHE_Process and http://www.ihe.net/Profiles.

The current version of the IHE Technical Framework can be found at http://www.ihe.net/Technical_Frameworks.

CONTENTS

Introduction

Security threats to confidentiality can be addressed in various ways including access control, physical control and encryption. The IHE 'Audit Trail and Node Authentication' (ATNA) Profile provides encryption at the network transaction level utilizing TLS or WS-Security. IHE 'Cross-Enterprise Document Media interchange' (XDM) Profile provides for encryption through the use of S/MIME when the media is Email. IHE Radiology 'Portable Data Interchange' (PDI) Profile provides for encryption of each DICOM®[1] object on the various media. Appendix Q provides more details on the encryption alternatives available from IHE. This supplement addresses encryption mechanisms to support confidentiality in two ways:

1. The Document Encryption Profile that provides a means to encrypt any kind of documents in a transport independent way. Its approach enables access to documents to be targeted to specific recipients.
2. The IHE XDM Media Encryption Option to enable the encryption of the whole XDM media content for use with the various media types (i.e., USB-memory, CD-ROM).

This supplement will provide end-to-end confidentiality to workflows where the document progresses in unanticipated paths, where workflows utilize many different transports, or where workflows involve storage systems such as USB media. The supplement addresses the need to protect documents from intermediaries in the document exchange path and provides confidentiality to transports that do not have a confidentiality mechanism.

This supplement allows for multiple methods of identity and key management. This makes it suitable for a rich set of healthcare environments (many of which have pre-existing key management infrastructure in place). Examples include symmetric keys, X.509 digital certificates, passwords, single and multiple recipients, and out-of-band key acquisition. To cater to these different situations the supplement contains hooks to different technical key management methods.

This supplement provides guidance for use in combination with XDS/XCA/XDR/XDM to specify how it is applied. However, document encryption is generic and can be applied to a rich range of transport means such as Email, HL7®[2]v2 message exchange and HTTP REST. The XDM Media Encryption Option specifically targets XDM actors to encrypt XDM media content.

Finally, policy aspects ranging from regulatory, organizational as well as privacy or consent policies (e.g., BPPC) are out of scope of this profile. This supplement is intended to support a broad range of reasonable policies that may determine what to encrypt.

[1] DICOM is the registered trademark of the National Electrical Manufacturers Association for its standards publications relating to digital communications of medical information.

[2] HL7 is the registered trademark of Health Level Seven International.

IHE encryption overview

This supplement adds two alternatives to the IHE profile portfolio for encryption. The existing alternatives and the two new alternatives are summarized in Table 1 together with an indication when the particular mechanism could be applicable. It is intended to support selection of the right encryption tool for the problem at hand.

Table 1: IHE encryption solution overview

Profile	When to use?
IHE ATNA (point-to-point using TLS)	• environment uses networking transactions (e.g., XDS/XDR); or • to be protected data concerns (representation of) XDS/XDR transactions and packages; and • confidentiality need applies between internet hosts (point-to-point)
IHE ATNA (end-to-end using WS-Security)	• environment uses web-services (SOAP); and • to be protected data concerns (representation of) transactions and packages (e.g., XDS/XDR); and • (partial) confidentiality need applies to intermediaries between end-points (end-to-end); and • where encryption between hosts is not sufficient
IHE XDM Email Option (using S/MIME)	• environment uses XDM with exchanges based on Email (SMTP); and • to be protected data concerns (representation of) XDM media content; and • confidentiality need extends from the sender up to the final recipient's Email system (end-to-end)
IHE Document Encryption	• environment uses any means for data exchange, in particular non-XD* means; or • to be protected data concerns (representation of) arbitrary data (documents), in particular non-XD* packages; or • confidentiality need applies between arbitrary end-points (end-to-end), in particular where intermediaries or unanticipated workflows are involved
IHE XDM Media Encryption Option	• environment uses XDM; and • to be protected data concerns (representation of) XDM media content (content and metadata) on physical media; and • confidentiality need matches path from creator to receiver (importer) of media
IHE PDI privacy Option (using CMS)	• environment uses PDI; and • to be protected data concerns (representation of) DICOM data on media; and • confidentiality need matches path from creator to receiver (importer) of media

Notes:

1. Table 1 summarizes the main options for encryption available in IHE. However, there are other non-IHE solutions available that may be more suitable in particular situations.
2. The table provides suggestions in which situation to apply which solution. Nevertheless, it is the responsibility of an implementer or adopter to diligently analyze his situation and select the proper solution.
3. Certain encryption solutions may be used in combination if needed, e.g., host-to-host encryption to protect transactional metadata and entity-to-entity encryption for the actual health information.
4. XDS and XDR and other IHE profiles require grouping with ATNA.

Use cases: encryption in IHE profiles

Table 2 presents the applicable use cases for encryption options offered by IHE profiles including existing profiles and ones new in this supplement (Document Encryption and XDM Media Encryption Option). The new use cases are defined in the next section. The 'other' use cases are derived from the respective IHE profiles.

Table 2: Use cases for existing and new IHE profiles with encryption
X = applicable (designed-for); (x) = suboptimal

Use case		Document Encryption	XDM Media Encryption Option	ATNA (TLS)	ATNA (WS-Security)	XDM Email Option (S/MIME)	PDI privacy Option (CMS)
Use cases (this supplement)							
1	Exchange health records using media	X	X				(x)
2	Media to media transfer	X	(x)				
3	File clerk import	X	X				
4	Unanticipated work-flows	X	(x)				
5	Clinical trial	X	X				
6	Multiple recipients of secure document	X	X				
7	Sharing with receivers only partially known a priori, a group or a role	X	X				(x)
8	Partial encrypted XDM submission set	X					
Other use cases (from other profiles)							
	Point-to-point network exchange between machines	(x)		X	(x)		
	Network exchange between machines in different trust domains	(x)		X	(x)		
	Online exchange of documents where partially trusted intermediaries are necessary	X			X		
	Exchange of medical documents using person-to-person Email	(x)				X	

Use case		Document Encryption	XDM Media Encryption Option	ATNA (TLS)	ATNA (WS-Security)	XDM Email Option (S/MIME)	PDI privacy Option (CMS)
	Media data (DICOM) exchange between healthcare enterprises using physical media	(x)	(x)				X

Notes:

1. Scenarios often have more candidate solutions; the exact circumstances determine the most applicable solution

2. Some profiles may be used in combination

Use cases: Document Encryption & XDM Media Encryption

The use cases for this supplement (Document Encryption Profile and XDM Media Encryption Option) are introduced below. These use cases focus on situations where existing point-to-point transport encryption (such as for example, IHE ATNA for IHE XDR and S/MIME for IHE XDM Zip-over-Email) is not appropriate or not available.

Use Case 1. Exchange health records using media (USB drives, CD-ROM)

Scenario flow:

a. A doctor gives his patient a CD-ROM with his record summary in encrypted form

b. The patient shares the encrypted document with other healthcare providers

Alternatively, the doctor saves the patient's record on a USB drive. The record on the medium may take the form of single file or XDM media content.

Document encryption enables the secure exchange of documents on a USB drive or CD-ROM, which do not have their own protection mechanism. As recipients are typically not yet known at encryption time. Recipients may obtain access to the encrypted record through a password supplied to them, through some out-of-band key management mechanism, etc.

A special case is formed by long term archival—either on storage media or in a repository—where there is a long time between the moment of encrypting and storing a document and the moment of retrieving and decrypting it for use by some party. This requires flexibility in key management as the final user of the document is not known at time of encryption nor storage.

Use Case 2. Media to media transfer

Media transfer is a variant of the use case to exchange records on media. Scenario flow:

a. A doctor Emails record summary to a patient as an encrypted document

b. The patient detaches the document and saves it on his USB drive

c. The patient shares the encrypted document with other healthcare providers

In a variant the patient downloads his record instead of having it Emailed by his doctor. Alternatively, IHE XDR is used. In an extension of the previous use case the patient transfers (copies) the contents from the CD-ROM to a USB drive as it is more travel-friendly.

As document encryption provides a generic means to encrypt sensitive health data independent of the transport medium it allows transfer of encrypted data between media without affecting protection during the process.

Use Case 3. File clerk import

Scenario flow:

a. Practice of doctor receives encrypted document

b. Clerk prepares digital file for doctor including encrypted document and brings it to the doctor

c. The doctor with help of his system decrypts the encrypted document

This scenario reflects two common healthcare workflow aspects namely import of documents (e.g., radiology images, see IHE RAD IRWF) and the involvement of support staff like clerks and nurses. Document encryption makes it possible to defer decryption to actual user of the data such as a doctor. Although such intermediary or environment could be trusted, technically preventing unnecessary access lowers the risks further.

In a common case the patient brings the files himself protected by a password known to him. In this scenario it should be avoided that a password is provided to the person who is responsible for the import of the document (separation of duty).

In another common case the file for import arrives by a courier on CD-ROM from another care provider, which has encrypted the document for the doctor, typically using PKI. A variant where it is encrypted by the doctor's department is supported by a method presented in use case 7.

Here the document is left intact though the clerk may manipulate metadata. In a variant of this use case the document is decrypted, its embedded metadata modified to reconcile, for example patient identifiers, and re-encrypted as part of the import.

Use Case 4. Unanticipated work-flows

Third party opinion scenario flow:

a. Disease management organization transfers encrypted CDA®[3] document to a GP in a different affinity domain.

b. GP accesses the document.

c. GP forwards the encrypted document to an expert specialist using IHE XDR.

d. Expert specialist accesses document for 2nd opinion.

[3] CDA is the registered trademark of Health Level Seven International.

These work-flows typically involve multiple transports, parties and exchanges of the encrypted document. Document encryption enables control over the access an intermediary person or system has to sensitive data in contrast to classic transport-level security.

Often recipients are known before the data is sent or forwarded in which case encryption can use a PKI-based approach with certificates and public/private keys. However, in more complex work-flows also passwords and out-of-band key retrieval may be used.

Although IHE XDR is used here as an example any transport could be used including Email, HL7v2 messages, IHE XCA, HTTP REST and IHE XDS.

Although policy aspects are out of scope of this profile, authorized parties are expected to handle the document in accordance with applicable policies.

Use Case 5. Accidental loss of media

This scenario addresses when accidents happen and the media where sensitive stored information is exposed or lost. This use case includes those with highly sensitive or even anonymized data of a clinical trial.

The typical scenario flow for a clinical trial consists of steps to prepare the data, transfer the data and use the data for research and analysis. Preparation of data typically involves—as mandated by regulation—anonymization, pseudonymization or blinding to protect the privacy of the subjects involved. While such measures sufficiently counter risks involving honest parties the data should still be considered sensitive.

Use Case 6. Multiple recipients of secure document

In this scenario a user encrypts a document for multiple recipients resulting in one efficient package (e.g., for a doctor and a specialist).

For reasons of efficiency, the same encrypted document can be used by multiple recipients each of whom have their own symmetric or asymmetric decryption keys or password. The encrypted document can be sent using the same or different transports.

Use Case 7. Sharing with receivers only partially known a priori a group or a role

In this scenario a nurse working for a disease management organization encrypts a patient's record for a doctor in an external hospital cardiology department. Upon receiving the encrypted document, the department assigns a particular cardiologist on duty. This doctor receives the record and uses it after decryption. A similar situation may be present with the exchange of cardiology information using XDM on physical media between institutions where the final user is unknown upfront.

The general case here is that the end-users are not fully known a priori (i.e., a group is known but not a particular member of this group). This can be a role instead of group as long as this role's membership is managed (e.g., a pharmacist registered in a national registry of pharmacists.)

In these cases the document is encrypted for (and using a key associated to) a particular entity like organization or system representing the group or role. A responsible representative or system with access to the proper keys then resolves the unspecific recipient to a specific recipient. It adds the specific recipients and amends the keys of the encrypted document to give a particular employee or role holder access. Alternatively, an end-user out-of-band requests proper keys at the party responsible for the encryption or its delegate.

Use Case 8. Partial encrypted set of documents

In this scenario flow, a care provider prepares an XDM medium with a document set containing an encrypted health document, and a non-encrypted document containing a subset of the health document content plus reference to the encrypted document. A receiving care provider uses the non-encrypted document to determine what to do with the document, and if he is an intended recipient, decrypts the encrypted document and uses the data.

Document encryption can be applied selectively to documents, notably to documents part of an XDM/XDR/XDS submission set.

Open Issues and Questions

285

#	Issue/(Question)
1	Determine file is encrypted using Document Encryption It is not possible to directly determine from a file itself if is encrypted using Document Encryption. There is no 'magic byte(s)' for CMS as exists for example, for DICOM, to determine file type easy and reliably from the file itself (without parsing). Document Encryption specifies the use of certain file extension and mime-type. It assumes these may be in a majority of cases, especially those where it really matters, however this may not old for all cases. Is current solution with file extension, mimeType and (partial) parsing sufficient?
2	XDM Security Considerations w.r.t. auditing in case of encryption Remark with respect to auditing may need expansion into full table in XDM Vol 2 Security Considerations.
3	Shared symmetric key method This supplement has enabled shared symmetric key methods. It addresses use cases where password or PKI are not appropriate or suboptimal (e.g., (a combination of) no human interaction, 'local' use (possibly longer term), recipient with no certificate, unknown final recipient, strong keys requirements, etc.). Complexity in implementation and use (potential confusion for adopters which method to use) should be considered. For this method, compliant implementations must support the related CMS options (common as it is a subset of functionality for password). Use is not mandated and use typically will be in environments that have a key management infrastructure in place. For testing pre-configured keys may be used, similar to the password method. Is the inclusion of shared symmetric key justified? Will digital certificate (PKI) and password methods be sufficient?
4	Is the relationship between XDM Media Encryption open and IHE PDI Privacy Option adequately described? The supplement introduction discusses both items. The XDM Profile text does not mention PDI. Assuming that in most cases the basic profile – XDM or PDI – is a given and that for confidentiality the applicable option is used, a detailed discussion in the XDM Profile is not required.

5	Algorithm support for actor responsible for encrypting Must the actor responsible for encrypting (Content Creator and XDM Portable Media creator) support all three algorithms, i.e., AES-128 CBC and AES-192 CBC and AES-256 CBC, or is one of them sufficient? Although supporting all three is preferred as it allows algorithm selection to be truly a policy decision, it may affect resource constraint devices. In any way, mandatory support does not imply mandatory use in a deployment. Currently, the actor responsible for encrypting (Content Creator and XDM Portable Media creator) must at least support one out of three. (The actor responsible for decryption must support all three (to ensure interoperability regardless of what the originating actor uses.)
6	The DEN Profile description arguably does not follow the format as defined in the introduction of Volume 3 Section 5: "This section follows the documentation pattern found in the IHE PCC Technical Framework. The reader should be familiar with the IHE PCC Technical Framework." Also, the format of Volume 3 Section 5 is arguably in flux. Once the format of Volume 3 has been settled, this is to be revisited.
7	Password based encryption does not yet have an upgrade path to phase out SHA1 for key derivation (PBKDF2), i.e., recommending ("should") a Portable Media Importer and Content Consumer to also support HMAC-SHA256. Reasons are that CMS and PBKDF2 standards do not yet identify HMAC-SHA256 as well as no OID seems to yet be defined. Once external standards have progressed this may be revisited.

Closed Issues

Selected closed issues have been documented on the IHE wiki at http://wiki.ihe.net/index.php?title=Document_Encryption_-_Discussion.

Volume 1 – Integration Profiles

1.7 History of Annual Changes

Add the following bullet to the end of the bullet list in Section 1.7

- Added the Document Encryption Profile which provides a means to encrypt health documents independent of particular transports and applications thereby offering end-to-end confidentiality.

Update Section 1.8

1.8 Security Implications

IHE transactions often contain information that must be protected in conformance with privacy laws and regulations, such as HIPAA or similar requirements in other regions. IHE includes a few security and privacy-focused profiles listed below. **In addition, Appendix Q provides an overview of encryption mechanisms in the IHE profile portfolio.** Other IHE Profiles generally do not have specific privacy protections, but rather expect a proper grouping with one or more of the security profiles:

- The Audit Trail and Node Authentication (ATNA) Profile specifies a means to ensure that nodes in a network are authenticated.
- The ATNA Profile specifies an audit message for reporting security- and privacy-relevant events.
- The Enterprise User Authentication (EUA) Profile specifies a means to authenticate system users and to share knowledge of the authenticated users among applications.
- The Personnel White Pages (PWP) Profile provides a repository that may be used to hold system users' identification data.
- **The Document Encryption (DEN) Profile specifies a means to ensure confidentiality of documents.**

Implementers may follow these IHE profiles to fulfill some of their security needs. It is understood that institutions must implement policy and workflow steps to satisfy enterprise needs and to comply with regulatory requirements.

2.1 Dependencies among Integration Profiles

Add the following to Table 2-1

Document Encryption	None	None	

Add the following section to Section 2.2

2.2.32 Document Encryption Profile

Document Encryption Profile provides a means to encrypt health documents independent of particular transport means, healthcare applications and document types, thereby supporting end-to-end confidentiality in heterogeneous workflows and unanticipated workflows. It enables access to documents to be targeted to specific recipients. It addresses the need to protect documents from certain intermediaries in the document exchange path and provides confidentiality to transports that do not have a confidentiality mechanism. The Document Encryption Profile allows for multiple alternatives for identity and key management which makes it suitable for a rich set of healthcare environments.

Add Section 32

32 Document Encryption Profile

The Document Encryption Profile addresses confidentiality interoperability needs. Security threats to confidentiality can be addressed in various ways including access control, physical control and encryption. IHE profiles alternatives for encryption include IHE 'Audit Trail and Node Authentication' (ATNA) for encryption at the network transaction level, IHE 'Cross-Enterprise Document Media interchange' (XDM) for encryption when the media is Email, and IHE Radiology 'Portable Data Interchange' (PDI) for encryption of DICOM objects on media. (Appendix Q provides more details on the encryption mechanisms available from IHE). The Document Encryption Profile extends this portfolio addresses confidentiality needs not addressed by the aforementioned profiles.

The Document Encryption Profile provides a means to encrypt health documents independent of particular transports means, healthcare applications and document types thereby supporting end-to-end confidentiality in heterogeneous workflows and unanticipated workflows. It enables access to documents to be targeted to specific recipients. It addresses the need to protect documents from certain intermediaries in the document exchange path and provides confidentiality to transports that do not have a confidentiality mechanism.

The Document Encryption Profile allows for multiple alternatives for identity and key management which makes it suitable for a rich set of healthcare environments (many of which have a pre-existing key management infrastructure in place). Examples include symmetric keys, X.509 digital certificates, passwords, single and multiple recipients, and out-of-band key acquisition. To cater to these different situations the profile contains hooks to different technical key management methods.

This profile provides guidance on use in combination with XDS/XCA/XDR/XDM to specify how it is applied. However, document encryption is generic and can be applied to a rich range of transport means such as Email, HL7v2 message exchange, and HTTP REST.

This profile does not define any specific policies. Instead, this supplement is intended to support a broad range of reasonable policies, e.g., stemming from regulatory, organizational as well as privacy or consent policies (e.g., BPPC). Policies may determine what to encrypt.

32.1 Actors/Transactions

The Document Encryption Profile is a Document Content profile so as to be independent of transport yet includes guidance for specific transports when they are chosen. Document Content Profiles utilize the defined actors of Content Creator and Content Consumer represented in Figure 32.1-1. The Content Creator is where the encryption of the document will take place, and the Content Consumer is where the decryption will take place. The key distribution is not shown as this profile does not single out a single key management methodology, and many key management methodologies are out-of-band or already in place (e.g., through the PWP or HPD Profiles).

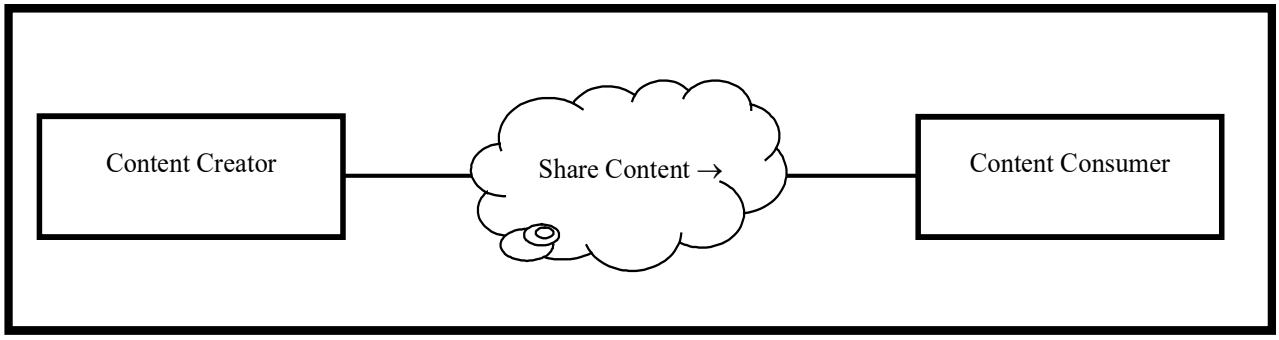

Figure 32.1-1: Document Encryption Actor Diagram

32.2 Document Encryption Profile Options

Options that may be selected for this profile are listed in the Table 32.2-1 along with the actors to which they apply. Dependencies between options when applicable are specified in notes.

Table 32.2-1: Document Encryption - Actors and Options

Actor	Options	Volume & Section
Content Creator	*No options defined*	- -
Content Consumer	*No options defined*	- -

32.3 Document Encryption Process Flow

32.3.1 Use Cases

The Document Encryption Profile provides a generic means to encrypt health documents. However, the primary use cases addressed by the profile are those where existing point-to-point transport encryption (e.g., IHE ATNA for IHE XDR, S/MIME for IHE XDM Zip-over-Email) is not appropriate or not available.

The primary use cases for document encryption are:

Use Case 1. Exchange health records using media (USB drives, CD-ROM)

Scenario flow:

a. A doctor gives his patient a CD-ROM with his record summary in encrypted form

b. The patient shares the encrypted document with other healthcare providers

Alternatively, the doctor saves the patient's record on a USB drive. The record on the medium may take the form of single file or XDM media content.

Document encryption enables the secure exchange of documents on a USB drive or CD-ROM, which do not have their own protection mechanism. As recipients are typically not yet known at encryption time. Recipients may obtain access to the encrypted record through a password supplied to them, through some out-of-band key management mechanism, etc.

A special case is formed by long term archival—either on storage media or in a repository—where there is a long time between the moment of encrypting and storing a document and the moment of retrieving and decrypting it for use by some party. This requires flexibility in key management as the final user of the document is not known at time of encryption nor storage.

Use Case 2. Media to media transfer

Media transfer is a variant of the use case to exchange records on media. Scenario flow:

a. A doctor Emails record summary to a patient as an encrypted document

b. The patient detaches the document and saves it on his USB drive

c. The patient shares the encrypted document with other healthcare providers

In a variant the patient downloads his record instead of having it Emailed by his doctor. Alternatively, IHE XDR is used. In an extension of the previous use case the patient transfers (copies) the contents from the CD-ROM to a USB drive as it is more travel-friendly.

As document encryption provides a generic means to encrypt sensitive health data independent of the transport medium it allows transfer of encrypted data between media without affecting protection during the process.

Use Case 3. File clerk import

Scenario flow:

a. Practice of doctor receives encrypted document

b. Clerk prepares digital file for doctor including encrypted document and brings it to the doctor

c. The doctor with help of his system decrypts the encrypted document

This scenario reflects two common healthcare workflow aspects namely import of documents (e.g., radiology images, see the IHE RAD Import Reconciliation Workflow

(IRWF) Profile) and the involvement of support staff like clerks and nurses. Document encryption makes it possible to defer decryption to actual user of the data such as a doctor. Although such intermediary or environment could be trusted, technically preventing unnecessary access lowers the risks further.

In a common case the patient brings the files himself, protected by a password known to him. In this scenario it should be avoided that a password is provided to the person who is responsible for the import of the document (separation of duty).

In another common case the file for import arrives by a courier on CD-ROM from another care provider, which has encrypted the document for the doctor, typically using PKI. A variant where it is encrypted by the doctor's department is supported by a method presented in use case 7.

Here the document is left intact though the clerk may manipulate metadata. In a variant of this use case the document is decrypted, its embedded metadata modified to reconcile, for example patient identifiers, and re-encrypted as part of the import.

Use Case 4. Unanticipated work-flows

Third party opinion scenario flow:

a. Disease management organization transfers encrypted CDA document to a GP in a different affinity domain.

b. GP accesses the document.

c. GP forwards the encrypted document to an expert specialist using IHE XDR.

d. Expert specialist accesses document for 2nd opinion.

These work-flows typically involve multiple transports, parties and exchanges of the encrypted document. Document encryption enables control over the access an intermediary person or system has to sensitive data in contrast to classic transport-level security.

Often recipients are known before the data is sent or forwarded in which case encryption can use a PKI-based approach with certificates and public/private keys. However, in more complex work-flows also passwords and out-of-band key retrieval may be used.

Although IHE XDR is used here as an example any transport could be used including Email, HL7v2 messages, IHE XCA, HTTP REST and IHE XDS.

Although policy aspects are out of scope of this profile, authorized parties are expected to handle the document in accordance with applicable policies.

Use Case 5. Accidental loss of media

This scenario addresses when accidents happen and the media where sensitive stored information is exposed or lost. This use case includes those with highly sensitive or even anonymized data of a clinical trial.

The typical scenario flow for a clinical trial consists of steps to prepare the data, transfer the data and use the data for research and analysis. Preparation of data typically involves—as mandated by regulation—anonymization, pseudonymization or blinding to protect the privacy of the subjects involved. While such measures sufficiently counter risks involving honest parties the data should still be considered sensitive.

Use Case 6. Multiple recipients of secure document

In this scenario a user encrypts a document for multiple recipients resulting in one efficient package (e.g., for a doctor and a specialist).

For reasons of efficiency the same encrypted document can be used by multiple recipients each of who have their own symmetric or asymmetric decryption keys or password. The encrypted document can be sent using the same or different transports.

Use Case 7. Sharing with receivers only partially known a priori a group or a role

In this scenario a nurse working for a disease management organization encrypts a patient's record for a doctor in an external hospital cardiology department. Upon receiving the encrypted document, the department assigns a particular cardiologist on duty. This doctor receives the record and uses it after decryption. A similar situation may be present with the exchange of cardiology information using XDM on physical media between institutions where the final user is unknown upfront.

The general case here is that the end-users are not fully known a priori (i.e., a group is known but not a particular member of this group). This can be a role instead of group as long as this role's membership is managed (e.g., a pharmacist registered in a national registry of pharmacists.)

In these cases the document is encrypted for (and using a key associated to) a particular entity like organization or system representing the group or role. A responsible representative or system with access to the proper keys then resolves the unspecific recipient to a specific recipient. It adds the specific recipients and amends the keys of the encrypted document to give a particular employee or role holder access. Alternatively, an end-user out-of-band requests proper keys at the party responsible for the encryption or its delegate.

Use Case 8. Partial encrypted set of documents

In this scenario flow, a care provider prepares an XDM medium with a document set containing an encrypted health document, and a non-encrypted document containing a subset of the health document content plus reference to the encrypted document. A receiving care provider uses the non-encrypted document to determine what to do with the document, and if he is an intended recipient, decrypts the encrypted document and uses the data.

Document encryption can be applied selectively to documents, notably to documents part of an XDM/XDR/XDS submission set.

32.3.2 Detailed Interactions

The Document Encryption process flow follows the basic process flow which is common for IHE content profiles and which is depicted in Figure 32.3.2-1. Document Encryption is typically deployed in an environment with particular transports and relies on functionalities like identity and key management. Figure 32.3.2-2 presents a Document Encryption process flow including grouping with other actors and out-of-band interactions and process steps.

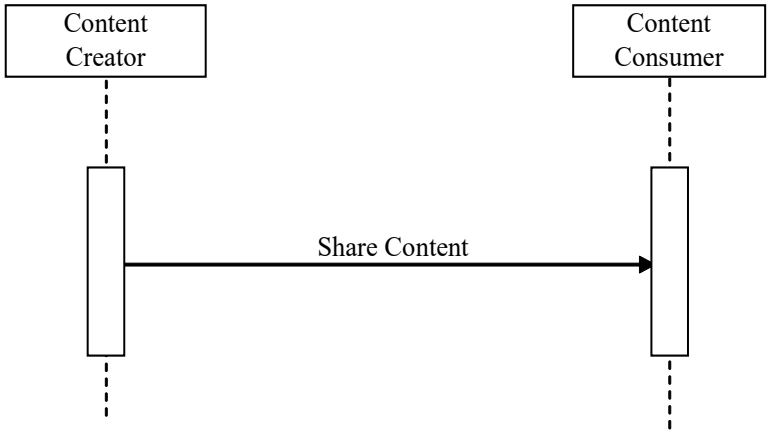

Figure 32.3.2-1: Generic Process Flow for Content Profiles

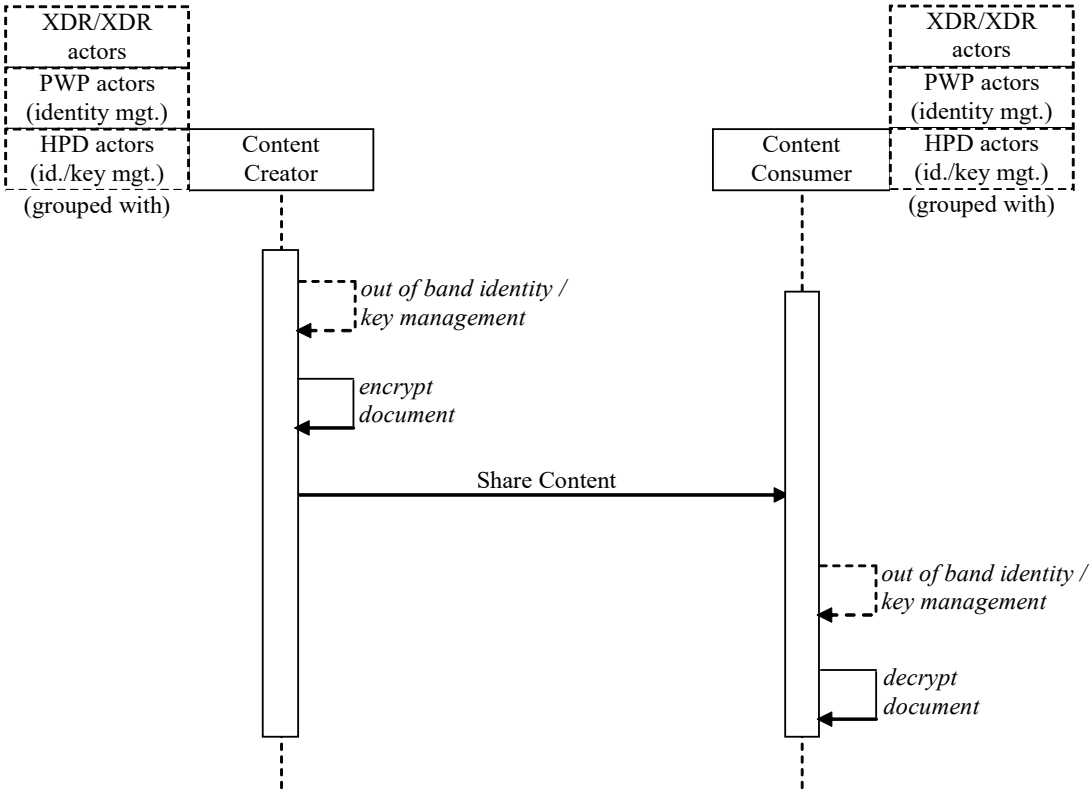

Figure 32.3.2-2: Example Process Flow in Document Encryption Profile

The above flow shows more actors than required in this profile so as to show how other IHE profiles can be used to automate Document Encryption. The IHE PWP and HPD profiles are available solutions for distribution of public certificates.

32.4 Key management

The Document Encryption Profile allows for multiple methods for identity and key management. This makes it suitable for a rich set of healthcare environments, many of which have a pre-existing key management infrastructure in place. For example, the IHE PWP and HPD profiles include methods for distribution of digital certificates as one possible way to support the PKI key management method. The profile is designed with flexibility in mind as systems should, for example, be able to handle multiple keys for transport (nodes) and document (entities).

Example key management hooks and features include symmetric and asymmetric keys with certificates, passwords, single and multiple recipients, upfront known recipients, recipients that are updated or added during the lifecycle of the encrypted document, and out-of-band key

acquisition. To cater to these different situations the profile contains hooks to different technical key management methods. Examples are the specification of password key derivation and allowing of encryption using keys and certificates that are managed externally.

It is recommended that in case of encryption of multiple documents that are together to be consistent across the multiple documents to use the same algorithms and keys/passwords. This avoids potential confusion for an encrypted document consumer.

32.5 Document Encryption Security Considerations

Implementers of this profile and parties deploying this profile should consider the following security aspects:

- Strength of passwords and keys
  This profile suggests the use of cryptographic algorithms that at the time of writing are approved (e.g., FIPS[4]) and considered secure. However, it is the responsibility of the implementer or deploying party to define a proper password policy, enforce the use of strong passwords, ensure generation of secure keys, etc.
- Expired or Revoke digital certificates
  When digital certificates (PKI) are used for either encryption or signing in this profile, it is up to local policy to mandate verification against Certificate Revocation Lists (CRL). When decryption happens, it is important to determine the state of the digital certificate at the time the encryption happened (given trustable time). Often times a digital certificate will have expired by the time the decryption happens, this is especially true of archives. The fact that the digital certificate is expired at the time of decryption is not a problem and should be expected. The concern is if the digital certificate was expired or revoked at or before the encryption. The underlying reason for key revocation is important, for example disclosure of the private key. This reason needs to be included in a warning to the user. Expired or revoked certificates at the time of encryption will invalidate any signatures but the effect on confidentiality is less obvious. In such cases it is advised that there is a policy that defines for example, if the data may still be decrypted and used. In such case the decryption will still function and produce a decrypted object, but the user should be warned, e.g., that the confidentiality of the document cannot be assured.
- Recovery of encrypted data from broken media
  Those employing Document Encryption are advised to consider that encryption typically increases the impact of errors in the data stream. For example, read errors due to scratched disks at a position in an encrypted file may not only affect that part of the document but the remainder of the document might not be recoverable.
- Security of metadata
  This profile presents how it can be combined with IHE XDR/XDS/XDM exchange protocols. It should be stressed however that this profile only protects the document and not the other transaction-related data. An example of this is the mandatory Document

[4] See NIST FIPS PUB 140-2 Annex A: Approved Security Functions

Sharing metadata, which may contain privacy sensitive data. The impact of this should be assessed separately. Possible action can be to minimize the metadata, but also to use transport security (e.g., IHE ATNA) to protect the transaction including the metadata. Another alternative is to encapsulate the original metadata and document(s) using for example, XDM with Media Encryption Option, and exchanging the encrypted result using a XD* transaction with minimal metadata. Both approaches exclude outsiders from access to the metadata while legitimate intermediaries and end-points can use the metadata for basic handling of the encrypted document.

- Policy aspects
  Policy aspects governing or relating to application of document encryption are out of scope of the profile. Deploying parties are advised to define (e.g., consolidating regulatory and organizational policies) a set of policies outlining when and how encryption should be applied to health documents. Alternatively, this may be derived dynamically from existing policies (e.g., Basic Patient Privacy Consent (BPPC) policies), or in combination.
- Long term storage and availability
  Long term storage of encrypted data puts requirements on key management. An adopter is advised to support long term availability of the data in such situations by proper key management. One such measure may be to have (at least one) institutionally managed or escrow keys instead of (just) keys and passwords bound to individual end-users.

16 Cross-Enterprise Media Interchange (XDM) Integration Profile

Update Section 16.2 and specifically Table 16.2-1

16.2 XDM Integration Profile Options

Options that may be selected for this Integration Profile are listed in Table 16.2-1 along with the actors to which they apply. Dependencies between options when applicable are specified in notes.

Table 16.2-1: XDM - Actors and Options

Actor	Options	Volume & Section
Portable Media Creator	USB (Note 1)	ITI TF-1: 16.2.1
	CD-R (Note 1)	ITI TF-1: 16.2.2
	ZIP over Email (Note 1)	ITI TF-1: 16.2.3
	Basic Patient Privacy Enforcement	ITI TF-2b: 3.32.4.1.4.1
	Zip over Email Response (Note 2)	ITI TF-1: 16.2.4
	Media Encryption (Note 3)	**ITI TF-1 16.2.5**
Portable Media Importer	USB (Note 1)	ITI TF-1: 16.2.1
	CD-R (Note 1)	ITI TF-1: 16.2.2
	ZIP over Email (Note 1)	ITI TF-1: 16.2.3

Actor	Options	Volume & Section
	Basic Patient Privacy Enforcement	ITI TF-2b: 3.32.4.1.4.1
	Zip over Email Response (Note 2)	ITI TF-1: 16.2.4
	Media Encryption (Note 3)	**ITI TF-1 16.2.5**

Note 1: At least one of these options is required for each actor. In order to enable a better interoperability, is highly recommended that the actors support all the options.

Note 2: This option requires the ZIP over Email Option.

Note 3: This option requires the USB or CD-R Option.

Add Section 16.2.5

16.2.5 Media Encryption Option

In this option the Portable Media Creator encrypts the XDM media content. The Portable Media Importer decrypts the XDM media content and imports the contained document set. If this option is supported, the USB or CD-R Option shall be supported.

Update Section 16.5

16.5 Security considerations

Update following paragraphs

In the case of physical media, security responsibilities for confidentiality and integrity **may be addressed using the Media Encryption Option. Alternatively, the responsibility ~~are~~ may be** transferred to the patient by providing the media **with unencrypted content** to the patient. In this case it is the patient's responsibility to protect the media, and the patient has the authority to disclose the contents of the media as they choose. They disclose the contents by providing the media.

The **user of the** Portable Media Creator **~~in most cases does not know who the ultimate Importer will be, thus rendering encryption impractical~~ will use her knowledge of the capabilities of Portable Media Importer and the context of the transaction to determine if encryption is necessary and what type of key management would be appropriate.**

In the case of transfer over Email using a ZIP attachment, the transaction is secured by the use of S/MIME.

Please add appendix Q to Vol 1

Appendix Q – IHE encryption overview

Table Q-1 summarizes the alternatives for encryption offered by the IHE profile portfolio for encryption together with an indication when the particular mechanism could be applicable. It is intended to support selection of the right encryption tool for the problem at hand.

Table Q-1: IHE encryption solution overview

Profile	When to use?
IHE ATNA (point-to-point using TLS)	• environment uses networking transactions (e.g., XDS/XDR); or • to be protected data concerns (representation of) XDS/XDR transactions and packages; and • confidentiality need applies between internet hosts (point-to-point)
IHE ATNA (end-to-end using WS-Security)	• environment uses web-services (SOAP); and • to be protected data concerns (representation of) transactions and packages (e.g., XDS/XDR); and • (partial) confidentiality need applies to intermediaries between end-points (end-to-end); and • where encryption between hosts is not sufficient
IHE XDM Email Option (using S/MIME)	• environment uses XDM with exchanges based on Email (SMTP); and • to be protected data concerns (representation of) XDM media content; and • confidentiality need extends from the sender up to the final recipient's Email system (end-to-end)
IHE Document Encryption	• environment uses any means for data exchange, in particular non-XD* means; or • to be protected data concerns (representation of) arbitrary data (documents), in particular non-XD* packages; or • confidentiality need applies between arbitrary end-points (end-to-end), in particular where intermediaries or unanticipated workflows are involved
IHE XDM Media Encryption Option	• environment uses XDM; and • to be protected data concerns (representation of) XDM media content (content and metadata) on physical media; and • confidentiality need matches path from creator to receiver (importer) of media
IHE PDI privacy Option (using CMS)	• environment uses PDI; and • to be protected data concerns (representation of) DICOM data on media; and • confidentiality need matches path from creator to receiver (importer) of media

Notes:

1. Table 1 summarizes the main options for encryption available in IHE. However, there are other non-IHE solutions available that may be more suitable in particular situations.
2. The table provides suggestions in which situation to apply which solution. Nevertheless, it is the responsibility of an implementer or adopter to diligently analyze his situation and select the proper solution.
3. Certain encryption solutions may be used in combination if needed, e.g., host-to-host encryption to protect transactional metadata and entity-to-entity encryption for the actual health information.
4. XDS and XDR and other IHE profiles require grouping with ATNA.

Different IHE encryption alternatives are applicable for different use cases. Table Q-2 presents the applicable use cases for encryption alternatives offered by IHE profiles. The use cases are derived from or taken from the respective IHE use cases including for example IHE ATNA and IHE DEN.

Table Q-2: Use cases for existing and new IHE profiles with encryption
X = applicable (designed-for); (x) = suboptimal

Use case		Document Encryption	XDM Media Encryption Option	ATNA (TLS)	ATNA (WS-Security)	XDM Email Option (S/MIME)	PDI privacy Option (CMS)
1	Point-to-point network exchange between machines	(x)		X	(x)		
2	Network exchange between machines in different trust domains	(x)		X	(x)		
3	Online exchange of documents where partially trusted intermediaries are necessary	X			X		
4	Exchange of medical documents using person-to-person Email	(x)				X	
5	Media data (DICOM) exchange between healthcare enterprises using physical media	(x)	(x)				X
6	Exchange health records using media	X	X				(x)
7	Media to media transfer	X	(x)				
8	File clerk import	X	X				
9	Unanticipated work-flows	X	(x)				
10	Clinical trial	X	X				
11	Multiple recipients of secure document	X	X				
12	Sharing with receivers only partially known a priori, a group or a role	X	X				(x)
13	Partial encrypted XDM submission set	X					

Notes:

1. Scenarios often have more candidate solutions; the exact circumstances determine the most applicable solution
2. Some profiles may be used in combination

Volume 2 – Transactions

Update Section 3.32

3.32 Distribute Document Set on Media

This section corresponds to Transaction ITI-32 of the IHE IT Infrastructure Technical Framework. Transaction ITI-32 is used by the Portable Media Creator to create the **XDM** media content **on media** and by **the** Portable Media Importer to read the **XDM** media content **from media**.

Update Section 3.32.3

3.32.3 Referenced Standards

DICOM PS3.10 Media Storage and File Format for Data Interchange (DICOM file format). http://dicom.nema.org/

DICOM PS3.12 Media Formats and Physical Media for Data Interchange, Annex F - 120mm CD-R media, Annex R - USB Connected Removable Devices, Annex V - ZIP File Over Media, and Annex W - Email Media. http://dicom.nema.org/

DICOM PS3.15 Security and System Management Profiles, Annex B - Secure Transport Connection Profiles. http://dicom.nema.org/

XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition). A Reformulation of HTML 4 in XML 1.0. W3C Recommendation 26 January 2000, revised 1 August 2002. http://www.w3.org/TR/xhtml1.

XHTML™ Basic. W3C Recommendation 19 December 2000. http://www.w3.org/TR/xhtm-basic.

MDN: RFC3798 Message Disposition Notification. http://www.rfc-editor.org/rfc/rfc3798.txt

Cryptographic Message Syntax (CMS), RFC5652, September 2009

Password-based Encryption for CMS, RFC3211, December 2001

Cryptographic Message Syntax (CMS) Algorithms", RFC3370, August 2002

"Use of the Advanced Encryption Standard (AES) Encryption Algorithm in Cryptographic Message Syntax (CMS)", RFC3565, July 2003

Create Section 3.32.4.1.2.4

3.32.4.1.2 Message Semantics

[…]

3.32.4.1.2.4 Media Encryption Option

In the case of physical media, e.g., CD-R or USB, the Portable Media Creator supporting the Media Encryption Option may encrypt the XDM media content. To encrypt, the Portable Media Creator shall ZIP the XDM media content (in line with DICOM Part 12 Annex V.2). Then the Portable Media Creator shall apply CMS encryption, as defined by Section 3.32.4.1.6, to the ZIP-ed media content. Finally, the Portable Media Creator shall assign the filename starting with "XDMME", followed by zero to three numerical characters, and with ".pk7" file extension to the resulting file on the media.

Create Section 3.32.4.1.2.4

3.32.4.1.4 Expected Actions

[…]

3.32.4.1.2.4 Media Encryption Option

In the case of physical media, e.g., CD-R or USB, the Portable Media Importer supporting the Media Encryption Option shall identify encrypted XDM media content by scanning the media file-system for files starting with "XDMME" and with a file extension equal to ".pk7". The Portable Media Importer shall decrypt encrypted XDM media content using the CMS standard as defined in Section 3.32.4.1.6. The Portable Media Importer shall un-ZIP the decrypted media content to obtain the XDM media content (in line with DICOM Part 12 Annex V.2).

Update Section 3.32.4.1.5

3.32.4.1.5 Security considerations

~~**In the case of physical media, encryption of the CD-R or USB shall not be used.**~~

~~**In the case the media used is the ZIP file over Email, the transaction shall be secured by S/MIME (see IHE ATNA) and comply with the security process as defined in the DICOM Part 15 Appendix (Secure Use of ZIP File Media over Email). The security process requires the use of S/MIME to both encrypt and sign the message. The encryption is used to maintain confidentiality during the transport. The signature is used to maintain integrity during transport and indicates that the sender is authorized to send the message.**~~

Portable Media Creators that create media shall generate one or more ATNA "Export" events into the audit trail to describe the media creation event. These events shall describe each submission set and/or study that is exported. **When the Media Encryption Option is used or when S/MIME is used for Email transport, the ATNA audit event shall set the ParticipantObject.Encrypted attribute to True.**

Portable Media Importers that import media shall generate one or more ATNA "Import" events into the audit trail to describe the media import event. These events shall describe each submission set and/or study that is imported. **When the Media Encryption Option is used or**

when S/MIME is used for Email transport, the ATNA audit event shall set the ParticipantObject.Encrypted attribute to True.

Note: It is easy to build a partial implementation of actors in the XDM Profile that lack the auditing capability. For example, a person can manually create media that comply with the requirements of the XDM media. It is possible that the manual process omits the generation of audit records for their activity. This would not be a compliant or complete implementation of the actors, but it is easy to make this kind of mistake.

The Portable Media Importer shall check the hash value and size as found in the Document Sharing metadata to detect corruption within the metadata or media. The Portable Media Importer shall notify the user if any errors are detected.

Create Section 3.32.4.1.6

3.32.4.1.6 Media Encryption Option specification

This section describes the requirements and constraints for the CMS encryption that apply when the XDM Media Encryption Option is used:

- The Portable Media Creator shall encrypt the ZIP-ed XDM media content as defined by the CMS specification [RFC5652] and the CMS options and constraints as specified in the following subsections.
- The Portable Media Importer shall decrypt the ZIP-ed XDM media content according to the CMS specification [RFC5652] and the CMS options and constraints as specified in the following subsections.

Table 3.32.4.1.6-1 presents the cryptographic algorithms referenced in the following subsections and/or CMS specification. The Portable Media Creator and Portable Media Importer actors shall support these cryptographic algorithms. The use of other cryptographic algorithms is outside the scope of this profile. The OIDs are taken from their respective sources.

Table 3.32.4.1.6-1: CMS cryptographic algorithms

Algorithm ID	OID	Function	
id-aes128-CBC	2.16.840.1.101.3.4.1.2	confidentiality	content encryption
id-aes192-CBC	2.16.840.1.101.3.4.1.22		
id-aes256-CBC	2.16.840.1.101.3.4.1.42		
id-aes128-wrap	2.16.840.1.101.3.4.1.5		password and symmetric key management method
id-aes192-wrap	2.16.840.1.101.3.4.1.25		
id-aes256-wrap	2.16.840.1.101.3.4.1.45		
id-PBKDF2	1.2.840.113549.1.5.12		password key management method
id-hmac-sha1	1.3.6.1.5.5.8.1.2		
rsaEncryption	1.2.840.113549.1.1.1		PKI key management method
		integrity	signature

Algorithm ID	OID	Function	
sha1WithRSAEncryption	1.2.840.113549.1.1.5		
sha256WithRSAEncryption	1.2.840.113549.1.1.11		
id-sha1	1.3.14.3.2.26		digest
id-sha256	2.16.840.1.101.3.4.2.1		

3.32.4.1.6.1 Content Type

The Portable Media Creator shall use the CMS enveloped-data content type.

The Portable Media Importer shall support the enveloped-data content type. The enveloped-data content type allows for encryption of content with support for various key management methods.

3.32.4.1.6.2 Content encryption

Portable Media Creator shall support encrypting content with AES-128 CBC, AES-196 CBC, or AES-256 CBC. The algorithm used is identified in CMS through the ContentEncryptionAlgorithmIdentifier [RFC3565].

The Portable Media Importer shall support AES-128 CBC, AES-196 CBC and AES-256 CBC [RFC3565] to decrypt the encrypted content. This permits a Portable Media Creator to determine the appropriate key length with the assurance that the Portable Media Importer can decrypt it regardless of which key length is chosen. The key used to encrypt the content is referred to as content encryption key.

3.32.4.1.6.3 Content integrity

Content integrity protection is used to enable the Portable Media Importer to validate that the decryption succeeded. For this purpose a digest or signature is added to the data before encryption.

The Portable Media Creator shall create a CMS digested-data or signed-data structure, which encapsulates the content. The resulting structure is encapsulated by the CMS enveloped-data structure. For both digested-data and signed-data the Portable Media Creator shall use SHA-256 as digest algorithm. In case of the signed-data the RSA algorithm shall be used [RFC3370].

The Portable Media Importer shall support the digested-data and signed-data content types. The Portable Media Importer, in order to assure that the decryption succeeded, shall verify a digest and may verify a signature. The ability to verify a signature will depend on the technical and trust infrastructure of the Portable Media Importer. The Portable Media Importer shall support the SHA-256 as well as SHA-1 digest algorithms.

3.32.4.1.6.4 Key management

The Portable Media Creator encrypts the content encryption key for one or more recipients. The Portable Media Creator and Portable Media Importer actors shall support the key management

methods listed below so as to enable the widest possible interoperability. For each recipient the Portable Media Creator shall apply one or more of these key encryption methods:

- PKI
- shared symmetric key
- password

There is no obligation to use all three methods in a deployment as this depends on the environment with e.g., availability of keys, key management infrastructure, work-flow, etc.

The following sections provide further requirements for each of the key management methods. Specifically, it discusses the CMS RecipientInfoType and KeyEncryptionAlgorithmIdentifier structures.

3.32.4.1.6.4.1 PKI

The PKI key management method applies asymmetric encryption to the symmetric key that encrypts the payload. It requires that the Portable Media Creator obtains the recipient's certificate as this contains the recipient's public key. The management of such certificate is out-of-scope of this transaction, but implementers can for example use the IHE PWP or HPD Profile to obtain certificates.

The PKI key management method uses key transport (KeyTransRecipientInfo) as CMS RecipientInfoType. The PKI key management method does not mandate the Portable Media Creator to support a particular encryption algorithm or related parameters such as key sizes. To use the PKI method the Portable Media Creator uses the algorithm and parameters as key size belonging to the recipient as specified by the recipient's certificate. A Portable Media Creator determines from the certificate content the algorithms and related parameters to use. CMS defines algorithm identifiers for e.g., RSA Encryption [RFC3370].

3.32.4.1.6.4.2 Shared symmetric key

The shared symmetric key method applies symmetric encryption to deliver the content encryption key to a recipient. The symmetric key can be pre-shared or involve key retrieval, both of which are out-of-scope of this transaction. Actors that use this method are assumed to have some kind of key management infrastructure in place supporting symmetric keys.

The shared symmetric key method uses symmetric key-encryption keys (KEKRecipientInfo) as CMS RecipientInfoType. Portable Media Creator and Portable Media Importer actors shall support AES key wrap algorithms (see Table 3.32.4.1.6-1). CMS mandates that the key length for the key encryption key minimally has the length of the content encryption key.

3.32.4.1.6.4.3 Password

The password key management method applies symmetric encryption to deliver the content encryption key to a recipient where the symmetric key is derived from a password. The Portable Media Creator uses a password known to the recipient or uses some means to make the password available to a recipient.

The password-based method uses password (PasswordRecipientInfo) as CMS RecipientInfoType. Portable Media Creator and Portable Media Importer shall support AES key wrap algorithms (see Table 3.32.4.1.6-1). CMS mandates that the key length for the key encryption key minimally have the length of the content encryption key.

The Portable Media Creator and Portable Media Importer Actors shall use PBKDF2 as key derivation algorithm [RFC3211]. Both Portable Media Creator and Portable Media Importer Actors shall support HMAC-SHA1 in the key derivation process. The properties of SHA1 reduce the key search space to 160 bits, which may be less than the 192 or 256 bit keys used for content encryption, but still more than the typical (effective) length of common passwords or pass phrases.

Passwords are defined as an octet string of arbitrary length whose interpretation as a text string is unspecified character encoding. It is recommended that use of characters is limited to the ASCII character set. This addresses environments in which character encoding cannot not explicitly identified e.g., when written down.

Volume 3 – Cross-Transaction Specifications and Content Specifications

Update Section 4.1 (specifically Table 4.1-5)

4.1 XDS Metadata

[…]

4.1.7 Document Definition Metadata

[…]

Table 4.1-5: Document Metadata Attribute Definition

XDSDocumentEntry Attribute	Definition	Source	Constraints
eventCodeListDisplay Name	The list of names to be displayed for communicating to human reader the meaning of the eventCode. If present, shall have a single value corresponding to each value in eventCodeList. See eventCodeList for an example.	O[3]	XDS Affinity Domain specific
formatCode	Code globally uniquely specifying the format of the document. Along with the typeCode, it should provide sufficient information to allow any potential XDS Document Consumer to know if it will be able to process the document. The formatCode shall be sufficiently specific to ensure processing/display by identifying a document encoding, structure and template (e.g., for a CDA Document, the fact that it complies with a CDA schema, possibly a template and the choice of a content-specific style sheet). **When the mimeType attribute indicates that the document is encrypted (a value such as application/pkcs7-mime) then the formatCode shall reflect the contents of the document before encryption.** Shall have a single value. Format codes may be specified by multiple organizations. Format codes defined by ITI shall have names with the prefix urn:ihe:iti: Format codes defined by other IHE domains shall have names with the prefix urn:ihe:"domain initials": Format codes defined by the Affinity Domain shall have names with the prefix urn:ad:"name of affinity domain": Affinity Domains shall be unique. The prefixes described here are not assumed to be exhaustive. […]	R	XDS Affinity Domain specific

XDSDocumentEntry Attribute	Definition	Source	Constraints
hash	[…]	[…]	[…]

Add Section 5.3

5.3 Document Encryption

5.3.1 References

- [RFC5652] Cryptographic Message Syntax (CMS), RFC5652, September 2009
- [RFC3211] Password-based Encryption for CMS, RFC3211, December 2001
- [RFC3370] Cryptographic Message Syntax (CMS) Algorithms", RFC3370, August 2002
- [RFC3565] "Use of the Advanced Encryption Standard (AES) Encryption Algorithm in Cryptographic Message Syntax (CMS)", RFC3565, July 2003
- [RFC2045] Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies, RFC2045, November 1996
- [RFC2183] Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field, August 1997

5.3.2 Document Encryption specification

The Document Encryption Profile specifies the operation of per-file document encryption. This section specifies the structure of encrypted documents according to the MIME and CMS specifications. It also presents the responsibilities of the Content Creator and Content Consumer Actors which are responsible for encryption and decryption respectively.

The encrypted document takes the form of a CMS ASN.1 encoded file, which contains CMS metadata and encapsulates a MIME entity, which encapsulates the original document, with an encryption transform applied. The following subsections specify applicable MIME and CMS options and constraints. The sections thereafter follow with guidelines for Document Sharing and transport bindings for encrypted documents.

5.3.2.1 MIME header

The document is encapsulated in a MIME wrapper. The resulting MIME entity is the CMS content to be encrypted. The following requirements apply to the Content Creator:

- The Content Creator shall prepare a MIME entity by applying a MIME wrapper to the document. [RFC2045].
- The Content Creator shall set the “Content-Type” MIME header [RFC2045] such that it equals the mime-type for the document if known.

- The Content Creator should use binary encoding and correspondingly set the "Content-transfer-encoding" MIME header to "binary" [RFC2045].
- The Content Creator shall set the "Content-Disposition" MIME header filename parameter to the filename of the document if known [RFC2183].
- The Content Creator shall create the MIME entity in canonical form.

The following requirements apply to the Content Consumer:

- The Content Consumer shall process the MIME entity according to [RFC2045].
- The Content Consumer may extract the "Content-Type" from the MIME headers to determine the content type or mime-type when present but may also use other information.
- The Content Consumer may use "Content-Disposition" filename parameter to determine the original filename of the document if present.

5.3.2.2 CMS processing

The Content Creator shall encrypt the MIME entity encapsulating the document as defined by the CMS specification [RFC5652] and the CMS options and constraints as specified in the following subsections.

The Content Consumer shall decrypt the encrypted document according to the CMS specification [RFC5652] and constraints as specified in the following subsections.

Table 5.3.2.2-1 presents the cryptographic algorithms referenced in the following subsections and/or CMS specification. The Content Creator and Content Consumer Actors shall support these cryptographic algorithms. The use of other cryptographic algorithms is outside the scope of this profile. The OIDs are taken from their respective sources.

Table 5.3.2.2-1: CMS cryptographic algorithms

Algorithm ID	OID	Function	
id-aes128-CBC	2.16.840.1.101.3.4.1.2	confidentiality	content encryption
id-aes192-CBC	2.16.840.1.101.3.4.1.22		
id-aes256-CBC	2.16.840.1.101.3.4.1.42		
id-aes128-wrap	2.16.840.1.101.3.4.1.5		password and symmetric key management method
id-aes192-wrap	2.16.840.1.101.3.4.1.25		
id-aes256-wrap	2.16.840.1.101.3.4.1.45		
id-PBKDF2	1.2.840.113549.1.5.12		password key management method
id-hmac-sha1	1.3.6.1.5.5.8.1.2		
rsaEncryption	1.2.840.113549.1.1.1		PKI key management method
		integrity	signature
sha1WithRSAEncryption	1.2.840.113549.1.1.5		

Algorithm ID	OID	Function	
sha256WithRSAEncryption	1.2.840.113549.1.1.11		
id-sha1	1.3.14.3.2.26		digest
id-sha256	2.16.840.1.101.3.4.2.1		

5.3.2.2.1 Content Type

The Content Creator shall use the CMS enveloped-data content type.

The Content Consumer shall support the enveloped-data content type. The enveloped-data content type allows for encryption of content with support for various key management methods.

5.3.2.2.2 Content encryption

Content Creator shall support encrypting content with AES-128 CBC, AES-196 CBC, or AES-256 CBC. The encryption algorithm used is identified in CMS through the ContentEncryptionAlgorithmIdentifier [RFC3565].

The Content Consumer shall support AES-128 CBC, AES-196 CBC and AES-256 CBC [RFC3565] to decrypt the encrypted content. This permits a Content Creator to determine the appropriate key length with the assurance that the Content Consumer can decrypt it regardless of which key length is chosen. The key used to encrypt the content is referred to as content encryption key.

5.3.2.2.3 Content integrity

Content integrity protection is used to enable the Content Consumer to validate that the decryption succeeded. For this purpose a digest or signature is added to the data before encryption.

The Content Creator shall create a CMS digested-data or signed-data structure, which encapsulates the content. The resulting structure is encapsulated by the CMS enveloped-data structure. For both digested-data and signed-data the Document Creator shall use SHA-256 as digest algorithm. In case of the signed-data the RSA algorithm shall be used [RFC3370].

The Content Consumer shall support the digested-data and signed-data content types. The Content Consumer, in order to assure that the decryption succeeded, shall verify a digest and may verify a signature. The ability to verify signature will depend on the technical and trust infrastructure of the Portable Media Importer. The Content Consumer shall support the SHA-256 as well as SHA-1 digest algorithms.

5.3.2.2.4 Key management

The Content Creator encrypts the content encryption key for one or more recipients. The Content Consumer and Content Creator Actors shall support the key management methods listed below so as to enable the widest possible interoperability. For each recipient the Content Creator shall apply one or more of these key encryption methods:

- PKI
- shared symmetric key
- password

There is no obligation to use all three methods in a deployment as this depends on the environment with e.g., availability of keys, key management infrastructure, work-flow, etc.

The following sections provide further requirements for each of the key management methods. Specifically, it discusses the CMS RecipientInfoType and KeyEncryptionAlgorithmIdentifier structures.

5.3.2.2.4.1 PKI

The PKI key management method applies asymmetric encryption to the symmetric key that encrypts the payload. It requires that the Content Creator obtains the recipient's certificate as this contains the recipient's public key. The management of such certificate is out-of-scope of this profile, but implementers can for example use the IHE PWP or HPD Profile to obtain certificates.

The PKI key management method uses key transport (KeyTransRecipientInfo) as CMS RecipientInfoType. The PKI key management method does not mandate the Content Creator to support a particular encryption algorithm or related parameters such as key sizes. To use the PKI method the Content Creator uses the algorithm and parameters as key size belonging to the recipient as specified by the recipient's certificate. A Content Creator determines from the certificate content the algorithms and related parameters to use. CMS defines algorithm identifiers for e.g., RSA Encryption [RFC3370].

5.3.2.2.4.2 Shared symmetric key

The shared symmetric key method applies symmetric encryption to deliver the content encryption key to a recipient. The symmetric key can be pre-shared or involve key retrieval, both of which are out-of-scope of this profile. Actors that use this method are assumed to have some kind of key management infrastructure in place supporting symmetric keys.

The shared symmetric key method uses symmetric key-encryption keys (KEKRecipientInfo) as CMS RecipientInfoType. Content Creator and Content Consumer Actors shall support AES key wrap algorithms (see Table 5.3.2.2-1). CMS mandates that the key length for the key encryption key minimally has the length of the content encryption key.

5.3.2.2.4.3 Password

The password key management method applies symmetric encryption to deliver the content encryption key to a recipient where the symmetric key is derived from a password. The Content Creator uses a password known to the recipient or uses some means to make the password available to a recipient.

The password-based method uses password (PasswordRecipientInfo) as CMS RecipientInfoType. Content Creator and Content Consumer Actors shall support AES key wrap

algorithms (see Table 5.3.2.2-1). CMS mandates that the key length for the key encryption key minimally have the length of the content encryption key. The Content Creator and Content Consumer Actors shall use PBKDF2 as key derivation algorithm [RFC3211]. Both Content Creator and Content Consumer Actors shall support HMAC-SHA1 in the key derivation process. The properties of SHA1 reduce the key search space to 160 bits, which may be less than the 192 or 256 bit keys used for content encryption, but still more than the typical (effective) length of common passwords or pass phrases.

Passwords are defined as an octet string of arbitrary length whose interpretation as a text string is unspecified character encoding. It is recommended that use of characters is limited to the ASCII character set. This addresses environments in which character encoding cannot not explicitly identified e.g., when written down.

5.3.3 Document Sharing Metadata

The Content Creator and Content Consumer Actors shall prepare and interpret respectively the Document Sharing Metadata for CMS encrypted documents as defined by ITI TF-3 and as further restricted by Table 5.3.3-1.

Table 5.3.3-1: XDSDocumentEntry metadata

XDSDocumentEntry Attribute	Document Encryption Requirement	Remark
hash	No special requirement for Document Encryption	Hash is calculated over encrypted document
mimeType	Set to “application/pkcs7-mime”	Identifies document encrypted according to Document Encryption Profile
size	No special requirement for Document Encryption	Size of encrypted document

Other attributes are not affected by this profile and may have a value as if this profile was not applied.

5.3.4 Transport bindings

Document Encryption is defined independent of content types and transport mechanisms. To facilitate integration with certain transport mechanisms this section discusses the bindings with certain transports.

Main purpose of the transport binding is to provide a means to determine that the content is encrypted according to the Document Encryption Profile and to determine the nature or type of the encrypted content.

In case multiple documents are transported in the same or related transactions, a Content Creator may decide per document to encrypt it according to the Document Encryption Profile or not.

5.3.4.1 XDM

For IHE XDM [ITI-32] transaction, the Content Creator may use ".p7m" as file extension for CMS encrypted files in the XDM file structure on media. The Content Creator should leave the base file name unchanged. For example, the file name "doc0001.xml" would become "doc0001.p7m". The original filename may be preserved in and retrieved from the MIME header within the CMS encapsulation.

5.3.4.2 Non-XD* transports

For many transports (media file systems, HTTP, etc.) only a limited amount of metadata is available to signal the nature of the content. Most common are the filename including file extension and mime-type. Recommendations include:

- A Content Creator may use ".p7m" as file extension for CMS encrypted content.
- A Content Creator may use "application/pkcs7-mime" as mime-type for CMS encrypted content.
- A Content Consumer may use above metadata to determine if this was data encrypted according to this Document Encryption Profile. The Content Consumer is advised to have a robust type determination method, because the content with the ".p7m" extension can also concern S/MIME protected content.

5.3.5 Security Considerations

The ATNA audit event that records the Creation of this document shall set the ParticipantObject.Encrypted attribute to True.

Once the document is decrypted, the decrypted content must be handled with care e.g., to prevent disclosure to unauthorized parties.

