# NEMA Standards Publication PS 3 Supplement 41

*Digital Imaging and Communications in Medicine (DICOM)*
*Digital Signatures*

**Status:** Final Text – 10 Sep 2001

*Prepared by*

**DICOM Standards Committee, Working Group 14**
1300 N. 17th Street, Suite 1847
Rosslyn, Virginia 22209 USA

# Table of Contents

## Foreword

Supplement 31 added features to allow DICOM applications to exchange messages over a Secure Transport Connection. While this protects the data during transit, it does not provide any lifetime integrity checks for DICOM SOP Instances. This supplement adds mechanisms for adding Digital Signatures to DICOM SOP Instances as a first step toward lifetime integrity checks. Digital Signatures, even when used with secure communications channels, do not provide comprehensive security in DICOM environments, since comprehensive security requires site guidelines and policies that are beyond the scope of the DICOM Standard. More comprehensive security might also require other considerations within DICOM that are not covered by this supplement. However, this supplement adds additional measures that applications may use in creating a more comprehensive secure environment within which DICOM could operate. Enhancements to or possibly even replacements for these mechanisms are expected in future versions as clinical needs are identified, as other related standards evolve, and as technology advances.

This supplement only addresses the following aspects of security:

- Authentication – verifying the identity of entities involved in an operation
- Data Integrity – verifying that data within an object has not been altered or removed

Covering other security aspects requires a more comprehensive security policy.

Digital Signatures allow authentication, or verification, of the identity entity that created, authorized, or modified a DICOM Data Set. This supplement adds Attribute sequences to Information Object Definitions that allow creators or modifiers of SOP Instances to certify their identity through Digital Signatures. This authentication is in addition to any authentication done when exchanging messages over a Secure Transport Connection.

The creator of a Digital Signature identifies the Data Elements of a SOP Instance that are included in the calculation of the MAC (Message Authentication Code) used in the Digital Signature. After the creator calculates the MAC, it then encrypts the MAC with a key or the private part of a key pair unique to the creator of the Digital Signature. The key or public part of a key pair is distributed to those recipients who need to verify the signature through means outside the scope of this Standard. Any receiver of the SOP Instance that knows the key or public part of the key pair can then recalculate the MAC and compare it with the MAC recorded in the Digital Signature, taking into account the encryption, in order to verify the integrity of the subset of Data Elements included in the calculation of the MAC for the Digital Signature. If any of the identified Data Elements had been altered or removed, it is extremely unlikely that the MAC calculated by the receiver and the MAC within the Digital Signature would agree. Typically the creator of the Digital Signature would only include Data Elements whose data had been verified in the MAC calculation for the Digital Signature. Note that any Digital Signatures embedded in SOP Instances can be valid for Media Interchange as well as Network Interchange.

This supplement adds modules and Attribute definitions to PS 3.3, PS 3.4, and PS 3.6 for implementing the Digital Signature functions. This supplement adds new Security Profiles to PS 3.15. Implementations may claim conformance to one or more Security Profiles.

## SCOPE

This Supplement allows receivers of an object to authenticate the source of the various pieces of information within the object, and to ascertain that the information has not been altered. The use of these mechanisms is optional.

This Supplement does not address issues of security policies, though clearly adherence to appropriate security policies is necessary for any level of security. This Supplement only provides mechanisms that could be used to implement security policies with regard to the interchange of DICOM objects between Application Entities.

Wherever possible this Supplement utilizes commonly available mechanisms rather than attempt to define DICOM specific mechanisms. By using existing mechanisms and standards this Supplement leverages the knowledge and expertise of those working intimately in the security field. This Supplement primarily selects available mechanisms and dictates how they may be applied within DICOM.

The initial focus of this Supplement is a short-term solution that can be implemented with existing tools. Since security mechanisms are still maturing, these short-term solutions may be replaced by more appropriate solutions in the future.

## ASSUMPTIONS

This Supplement assumes that Application Entities can securely identify local users of the Application Entity, and that user's roles or licenses. Note that users may be persons, or may be abstract entities, such as organizations or pieces of equipment. When Application Entities agree to an exchange of information via DICOM, they may also exchange information about the users of the Application Entity via the Certificates exchanged with the Digital Signatures.

This Supplement also assumes that an Application Entity has secure access to or can securely obtain X.509 key Certificates for the users of the application entity. In addition, this Supplement assumes that an Application Entity has the means to validate an X.509 certificate that it receives. The validation mechanism may use locally administered authorities, publicly available authorities, or some trusted third party.

This Supplement cannot assume that all Application Entities preserve bit-for-bit all of the information received via DICOM, implying that such Application Entities may not pass DICOM SOP Instances along without some minor differences in the data. For example, some implementations may strip or add trailing blanks to text fields. However, this Supplement does assume that there are some Application Entities that do preserve and can pass along exact copies of DICOM SOP Instances.

Several sources can contribute to the information content of a DICOM SOP instance. This Supplement proposes a mechanism for embedding multiple digital signatures within DICOM objects. The digital signatures could authenticate the sources of various pieces of information within the object, and could serve as lifetime integrity checks of the information signed. Since not all DICOM implementations are bit preserving, this Supplement provides a mechanism for an application entity to verify the incoming digital signature, then replace it with one that the application entity generates before sending the object on. Further, this Supplement adds a field to extended negotiation for the storage service class by which application entities can determine if their communication partners are bit preserving (and hence digital signature preserving) or not.

# Additions to PS 3.2

*Item 1.2 Add the following to the conformance requirements in Section 7.5*

An implementation shall declare in its Conformance Statement which level of security features it supports, including such things as:

a. The conditions under which the implementation preserves the integrity of Digital Signatures (e.g. is the implementation bit-preserving).
b. The conditions under which the implementation verifies incoming Digital Signatures.
c. The conditions under which the implementation replaces Digital Signatures.

# Additions To PS 3.3

*Item 2.1 Add the following references to Section 2*

ITU-T Recommendation X.509 (03/00) "Information technology - Open Systems Interconnection - The directory: Public-key and attribute certificate frameworks"

Note: ITU-T Recommendation X.509 is similar to ISO/IEC 9594-8 1990. However, the ITU-T recommendation is the more familiar form, and was revised in 1993 and 2000, with two sets of corrections in 2001. ITU-T was formerly known as CCITT.

ISO/IEC 10118-3:1998 Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions (RIPEMD-160 reference)

Note: The draft RIPEMD-160 specification and sample code are also available at ftp://ftp.esat.kuleuven.ac.be/pub/bosselae/ripemd

IETF RFC 2437 PKCS #1 RSA Cryptography Specifications Version 2.0

Note: The RSA Encryption Standard is also defined in informative annex A of ISO/IEC 9796, and in Normative Annex A of the CEN/TC251 European Prestandard prENV 12388:1996.

ISO 7498-2 Information processing systems – Open Systems Interconnection – Basic reference Model – Part 2: Security Architecture

ECMA 235 The ECMA GSS-API Mechanism

FIPS PUB 46 Data Encryption Standard

FIPS PUB 81 DES Modes of Operation

IETF Internet X.509 Public Key Infrastructure; Time Stamp Protocols; March 2000

*item 2.2 Add to following subsections to Section 3*

**3.14 Reference Model Security Architecture Definitions**

This Part of the Standard makes use of the following terms defined in ISO 7498-2:

a. Digital Signature

Note: The definition is “Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of that unit and protect against forgery e.g. by the recipient.”

b. Data Confidentiality

Note: The definition is “the property that information is not made available or disclosed to unauthorized individuals, entities or processes.”

c. Data Origin Authentication

Note: The definition is “the corroboration that the source of data received is as claimed.”

d. Data Integrity

Note: The definition is “the property that data has not been altered or destroyed in an unauthorized manner.”

e. Key Management

Note: The definition is “the generation, storage, distribution, deletion, archiving and application of keys in accordance with a security policy.”

**3.15 Security Definitions**

This Part of the Standard makes use of the following terms defined in ECMA 235:

a. Security Context

Note: The definition is “security information that represents, or will represent a Security Association to an initiator or acceptor that has formed, or is attempting to form such an association.”

*Item 2.3 Add the following subsection to Section 3*

**3.16 DICOM security profiles**

This part of the Standard makes use of the following terms defined in PS 3.15:

a. Message Authentication Code
b. Certificate

*Item 2.4 Add the following abbreviations to Section 4*

MAC Message Authentication Code

ITU-T International Telecommunications Union – Telecommunications Standardization Sector

*Item 2.5 Add the following sub-sections to the definition of the attributes in the SOP Common Module*

### C.12.1.1.3 Digital Signatures Macro

This Macro allows Digital Signatures to be included in a DICOM Data Set for the purpose of insuring the integrity of the Data Set, and to authenticate the sources of the Data Set. Table C.12-5 defines the Attributes needed to embed a Digital Signature in a Data Set. This Macro may appear in individual sequence items as well as in the main Data Set of the SOP Instance.

Note: Each Item of a Sequence of Items is a Data Set. Thus, individual Sequence items may incorporate their own Digital Signatures in addition to any Digital Signatures added to the Data Set in which the Sequence appears.

**Table C.12-5**
**DIGITAL SIGNATURES MACRO ATTRIBUTES**

| Attribute Name | Tag | Type | Attribute Description |
|---|---|---|---|
| MAC Parameters Sequence | (4FFE,0001) | 3 | A sequence of one or more items that describe the parameters used to calculate a MAC for use in Digital Signatures. |
| >MAC ID Number | (0400,0005) | 1 | A number used to identify this MAC Parameters Sequence item. |
| >MAC Calculation Transfer Syntax UID | (0400,0010) | 1 | The Transfer Syntax UID used to encode the values of the Data Elements included in the MAC calculation. Only Transfer Syntaxes that explicitly include the VR and use Little Endian encoding shall be used.<br>Notes: Certain Transfer Syntaxes, particularly those that are used with compressed data, allow the fragmentation of the pixel data to change. If such fragmentation changes, Digital Signatures generated with such Transfer Syntaxes could become invalid. |
| >MAC Algorithm | (0400,0015) | 1 | The algorithm used in generating the MAC to be encrypted to form the Digital Signature.<br>Defined Terms: RIPEMD160<br>MD5<br>SHA1<br>.<br>Note: Digital Signature Security Profiles (see PS 3.15) may require the use of a restricted subset of these terms. |
| >Data Elements Signed | (0400,0020) | 1 | A list of Data Element Tags in the order they appear in the Data Set which identify the Data Elements used in creating the MAC for the Digital Signature. See Section C.12.1.1.3.1.1. |
| Digital Signatures Sequence | (FFFA,FFFA) | 3 | Sequence holding one or more Digital Signatures. |
| >MAC ID Number | (0400,0005) | 1 | A number used to identify which MAC |

| | | | |
|---|---|---|---|
| | | | Parameters Sequence item was used in the calculation of this Digital Signature. |
| >Digital Signature UID | (0400,0100) | 1 | A UID that can be used to uniquely reference this signature. |
| >Digital Signature DateTime | (0400,0105) | 1 | The date and time the Digital Signature was created. The time shall include an offset (i.e., time zone indication) from Coordinated Universal Time.<br>Note: This is not a certified timestamp, and hence is not completely verifiable. An application can compare this date and time with those of other signatures and the validity date of the certificate to gain confidence in the veracity of this date and time. |
| >Certificate Type | (0400,0110) | 1 | The type of certificate used in (0400,0115).<br>Defined Term: X509_1993_SIG<br>Note: Digital Signature Security Profiles (see PS 3.15) may require the use of a restricted subset of these terms. |
| >Certificate of Signer | (0400,0115) | 1 | A certificate that holds the identity of the entity producing this Digital Signature, that entity's public key or key identifier, and the algorithm and associated parameters with which that public key is to be used. Algorithms allowed are specified in Digital Signature Security Profiles (see PS 3.15).<br>Notes: 1. As technology advances, additional encryption algorithms may be allowed in future versions. Implementations should take this possibility into account.<br>2. When symmetric encryption is used, the certificate merely identifies which key was used by which entity, but not the actual key itself. Some other means (e.g., a trusted third party) must be used to obtain the key. |
| >Signature | (0400,0120) | 1 | The MAC generated as described in Section 12.2.1.1 and encrypted using the algorithm, parameters, and private key associated with the Certificate of the Signer (0400,0115). See Section C.12.1.1.3.1.2. |
| >Certified Timestamp Type | (0400,0305) | 1C | The type of certified timestamp used in the Certified Timestamp (0400,0310) Attribute. Required if Certified Timestamp (0400,0310) is present.<br>Defined Terms: CMS_TSP – Internet X.509 Public Key Infrastructure Time Stamp Protocol<br>Note: Digital Signature Security Profiles (see PS 3.15) may require the use of |

| | | | |
|---|---|---|---|
| | | | a restricted subset of these terms. |
| >Certified Timestamp | (0400,0310) | 3 | A certified timestamp of the Digital Signature (0400,0120) Attribute Value, which shall be obtained when the Digital Signature is created. See Section C.12.1.1.3.1.3. |

### C.12.1.1.3.1 Digital Signature Attribute Descriptions

#### C.12.1.1.3.1.1 Data Elements Signed

The Data Elements Signed Attribute shall list the Tags of the Data Elements that are included in the MAC calculation. The Tags listed shall reference Data Elements at the same level as the Mac Parameters Sequence (4FFE,0001) Data Element in which the Data Elements Signed Attribute appears. Tags included in Data Elements Signed shall be listed in the order in which they appear within the Data Set.

The following Data Elements shall not be included either implicitly or explicitly in the list of Tags in Data Elements Signed, nor included as part of the MAC calculation:

- The Length to End (0008,0001) or any Tag with an element number of 0000 (i.e., no data set or group lengths may be included in MAC calculations)
- Tags with a group number less than 0008
- Tags associated with Data Elements whose VR is UN
- Tags of Data Elements whose VR is SQ, where any Data Element within that Sequence of Items has a VR of UN recursively
- Tags with a group number of FFFA (e.g. the Digital Signatures Sequence)
- MAC Parameters Sequence (4FFE,0001)
- Data Set Trailing Padding (FFFC,FFFC)
- Item Delimitation Item (FFFE,E00D)

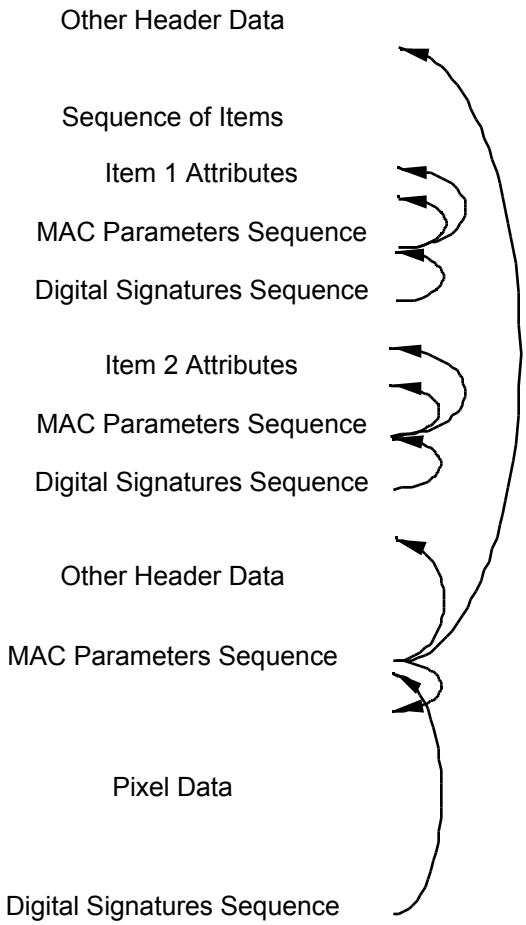
Notes: 1. The Length to End and group lengths can change if non-signed Data Elements change, so it is not appropriate to include them in the MAC calculation.

2. Since the Data Element Tags identifying a sequence and which start each item are included in the MAC calculation, there is no need to include the Item Delimitation Item Tags.

If any of the Data Element Tags in the list refer to a Sequence of Items, then the Tags of all Data Elements within all Items of that Sequence shall be implicitly included in the list of Data Elements Signed, except those disallowed above. This implicit list shall also include the Item Tag (FFFE,E000) Data Elements that separate the Sequence Items and the Sequence Delimitation Item (FFFE,E0DD).

Notes: It is possible to sign individual items within a sequence by including the Digital Signatures Macro in that sequence item. In fact, this is a highly desirable feature, particular when used in the context of reports. The Digital Signatures Macro is applied at the Data Set level, and Sequences of Items are merely Data Sets embedded within a larger Data Set. Essentially, the Digital Signature Macro may be applied recursively.

An example of nesting Digital Signatures within Data Elements is illustrated in the following figure:




In this example, there is main signature covering the pixel data and a few other Data Elements, plus two individually signed items within a sequence.

For Data Elements with a VR OB (e. g. pixel data) that have an undefined length (i.e. the data is encapsulated as described in PS 3.5), the Item Data Element Tags that separate the fragments shall implicitly be included in the list of Data Elements Signed (i.e. a Data Element with a VR of OB is encoded in the same fashion as a Sequence of Items).

#### C.12.1.1.3.1.2 Signature

To generate the MAC, Data Elements referenced either explicitly or implicitly by the Tags in the Data Elements Signed list shall be encoded using the Transfer Syntax identified by the MAC Calculation Transfer Syntax UID (0400,0010) of the MAC Parameters Sequence item where the Data Elements Signed Attribute appears. Data shall be formed into a byte stream and presented to the MAC Algorithm for computation of the MAC according to the following rules:

For all Data Elements except those with a VR of SQ or with a VR of OB with an undefined length, all Data Element fields, including the Tag, the VR, the reserved field (if any), the Value Length, and the Value, shall be placed into the byte stream in the order encountered.

For Data Elements with a VR of SQ or with a VR of OB with an undefined length, the Tag, the VR, and the reserved field are placed into the byte stream. The Value Length shall not be included. This is followed by each Item Tag in the order encountered, without including the Value Length, followed by the contents of the Value for that item. In the case of an Item within a Data Element whose VR is SQ, these rules are applied recursively to all of the Data Elements within the Value of that Item. After all the Items have been incorporate into the byte stream, a Sequence Delimitation Item Tag (FFFE,E0DD) shall be added to the byte stream presented to the MAC Algorithm, regardless of whether or not it was originally present.

Note: Since the Value Length of Data Elements with a VR of SQ can be either explicit or undefined, the Value Lengths of such Data Elements are left out of the MAC calculation. Similarly, the Value Length of Data Elements with a VR of OB with an undefined length are also left out so that they are handled consistently. If such Data Elements do come with undefined lengths, including the Item Tags that separate the Items or fragments insures that Data Elements cannot be moved between Items or Fragments without compromising the Digital Signature. For those Data Elements with explicit lengths, if the length of an item changes, the added or removed portions would also impact the MAC calculation, so it is not necessary to include explicit lengths in the MAC calculation. It is possible that including the Value Lengths could make cryptoanalysis easier.

After the fields of all the Data Elements in the Data Elements Signed list have been placed into the byte stream presented to the MAC Algorithm according to the above rules, all of the Data Elements within the Digital Signatures Sequence item except the Certificate of Signer (0400,0115), Signature (0400,0120), Certified Timestamp Type (0400,0305), and Certified Timestamp (0400,0310) shall also be encoded according to the above rules, and presented to the MAC algorithm (i.e., the Attributes of the Digital Signature Sequence Item for this particular Digital Signature are also implicitly included in the list of Data Elements Signed, except as noted above).

The resulting MAC code after processing this byte stream by the MAC Algorithm is then encrypted as specified in the Certificate of Signer and placed in the Value of the Signature Data Element.

Notes: 1. The Transfer Syntax used in the MAC calculation may differ from the Transfer Syntax used to exchange the Data Set.

2. Digital Signatures require explicit VR in order to calculate the MAC. An Application Entity which receives a Data Set with an implicit VR Transfer Syntax may not be able to verify Digital Signatures that include Private Data Elements or Data Elements unknown to that Application Entity.This also true of any Data Elements whose VR is UN. Without knowledge of the Value Representation, the receiving Application Entity would be unable to perform proper byte swapping or be able to properly parse sequences in order to generate a MAC.

3. If more than one entity signs, each Digital Signature would appear in its own Digital Signatures Sequence item. The Digital Signatures may or may not share the same MAC Parameters Sequence item.

4. The notion of a notary public (i.e., someone who verifies the identity of the signer) for Digital Signatures is partially filled by the authority that issued the Certificate of Signer.

#### C.12.1.1.3.1.3 Certified Timestamp

To generate a certified timestamp, the Value of the Signature (0400,0120) Attribute is sent to a third party, as specified by the protocol referred to by the Certified Timestamp Type (0400,0305) Attribute. The third party then generates and returns a certified timestamp in the form specified by that protocol. The certified timestamp returned by the third party is encoded as a stream of bytes in the Certified Timestamp Attribute.

Note: The timestamp protocol may be specified by a Profile in PS 3.15.

*Item 2.6 Add the following rows to Table C.12-1 sop common module attributes*

| Attribute Name | Tag | Type | Attribute Description |
|---|---|---|---|
| | | | |
| | | | |
| *Include 'Digital Signatures Macro' Table C.12-5* | | | |

## Additions to PS 3.4

*Item 3.1 Replace the third entry in Table B.3-1 with the following*

| Item Bytes | Field Name | Description of Field |
|---|---|---|
| 3 | Level of Digital Signature support | A Level 2 SCP may further define its behavior in this byte field.<br>0 – The signature level is unspecified, the AE is an SCU only, or the AE is not a level 2 SCP<br>1 – signature level 1<br>2 – signature level 2<br>3 – signature level 3<br>If extended negotiation is not supported, the default shall have a value of 0. |

*Item 3.2 Replace the third entry in Table B.3-2 with the following*

| Item Bytes | Field Name | Description of Field |
|---|---|---|
| 3 | Level of Digital Signature support | A Level 2 SCP may further define its behavior in this byte field.<br>0 – The signature level is unspecified, the AE is an SCU only, or the AE is not a level 2 SCP<br>1 – signature level 1<br>2 – signature level 2<br>3 – signature level 3<br>If extended negotiation is not supported, no assumptions shall be made by the Association-requester about the capabilities of the Association-acceptor based upon this extended negotiation. |

*Item 3.3 Add the following to the end of section B.4.1*

Three levels of Digital Signature support are defined for an SCP which claims conformance to Level 2 (Full) storage support:

Signature Level 1. SCP may not preserve Digital Signatures and does not replace them.

Signature Level 2. SCP does not preserve the integrity of incoming Digital Signatures, but does validate the signatures of SOP Instances being stored, takes implementation-specific measures for insuring the integrity of data stored, and will add replacement Digital Signatures before sending SOP Instances elsewhere.

Signature Level 3. SCP does preserve the integrity of incoming Digital Signatures (i.e. is bit-preserving and stores and retrieves all Attributes regardless of whether they are defined in the IOD).

*Item 3.4 Add the following after the first bullet item of Section B.4.3.2*

The level of Digital Signature support, as defined by Section B.4.1, shall be stated.

## Additions to PS 3.6

*Item 4.1 Add the following rows to the table in section 6*

| Tag | Name | VR | VM |
|---|---|---|---|
| (0400,0005) | MAC ID number | US | 1 |
| (0400,0010) | MAC Calculation Transfer Syntax UID | UI | 1 |
| (0400,0015) | MAC Algorithm | CS | 1 |
| (0400,0020) | Data Elements Signed | AT | 1-n |

| | | | |
|---|---|---|---|
| (0400,0100) | Digital Signature UID | UI | 1 |
| (0400,0105) | Digital Signature DateTime | DT | 1 |
| (0400,0110) | Certificate Type | CS | 1 |
| (0400,0115) | Certificate of Signer | OB | 1 |
| (0400,0120) | Signature | OB | 1 |
| (0400,0305) | Certified Timestamp Type | CS | 1 |
| (0400,0310) | Certified Timestamp | OB | 1 |
| (4FFE,0001) | MAC Parameters Sequence | SQ | 1 |
| (FFFA,FFFA) | Digital Signatures Sequence | SQ | 1 |

## Additions to PS 3.15

*Item 7.1A Correct the following reference in Section 2*

ECMA 23~~3~~5, The ECMA GSS-API Mechanism

*Item 7.1B Add the following references to Section 2*

ITU-T Recommendation X.509 (03/00) "Information technology - Open Systems Interconnection - The directory: Public-key and attribute certificate frameworks"

Note: ITU-T Recommendation X.509 is similar to ISO/IEC 9594-8 1990. However, the ITU-T recommendation is the more familiar form, and was revised in 1993 and 2000, with two sets of corrections in 2001. ITU-T was formerly known as CCITT.

ISO/IEC 10118-:1998 Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions (RIPEMD-160 reference)

Note: The draft RIPEMD-160 specification and sample code are also available at ftp://ftp.esat.kuleuven.ac.be/pub/bosselae/ripemd

IETF RFC 2437 PKCS #1 RSA Cryptography Specifications Version 2.0

Note: The RSA Encryption Standard is also defined in informative annex A of ISO/IEC 9796, and in Normative Annex A of the CEN/TC251 European Prestandard prENV 12388:1996.

FIPS PUB 46 Data Encryption Standard

FIPS PUB 81 DES Modes of Operation

IETF Internet X.509 Public Key Infrastructure; Time Stamp Protocols; March 2000

*Item 7.2 Add the following definition to Section 3.2 (definitions from ISO 7498-2)*

e. Digital Signature

Note: The definition is “Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of that unit and protect against forgery e.g. by the recipient.”

*Item 7.3 Extend the following subsection in Section 3*

### 3.10 DICOM SECURITY PROFILES DEFINITIONS

**Message Authentication Code:** A digest or hash code derived from a subset of Data Elements.

**Certificate:** An electronic document that identifies a party and that party’s public encryption algorithm, parameters, and key. The Certificate also includes, among other things, the identity and a digital signature from the entity that created the certificate. The content and format of a Certificate are defined by ITU-T Recommendation X.509.

*Item 7.4 Add the following subsection to Section 6*

### 6.3 DIGITAL SIGNATURE PROFILE

An implementation may claim conformance to one or more Digital Signature Profiles.

A Digital Signature profile consists of the following information:

a. The role that the Digital Signature plays, including:
    1. Who or what entity the Digital Signature represents.
    2. A description of the purpose of the Digital Signature.
    3. The conditions under which the Digital Signature is included in the Data Set.
b. A list of Attributes that shall be included in the Digital Signature.
c. The mechanisms that shall be used to generate or verify the Digital Signature, including:
    1. The algorithm and relevant parameters that shall be used to create the MAC or hash code, including the Value to be used for the MAC Algorithm (0400,0015) Attribute.
    2. The encryption algorithm and relevant parameters that shall be used to encrypt the MAC or hash code in forming the Digital Signature.
    3. The certificate type or key distribution mechanism that shall be used, including the Value to be used for the Certificate Type (0400,0110) Attribute.
    4. Any requirements for the Certified Timestamp Type (0400,305) and Certified Timestamp (0400,310) Attributes.
d. Any special requirements for identifying the signatory.
e. The relationship with other Digital Signatures, if any.
f. Any other factors needed to create, verify, or interpret the Digital Signature

Digital Signature Profiles are specified in Annex C.

*Item 7.5 Add the following sections to Annex A*

### A.2 BASIC DIGITAL SIGNATURES SECURE USE PROFILE

An implementation that validates and generates Digital Signatures may claim conformance to the Basic Digital Signatures Secure Use Profile. Any implementation that claims conformance to this Security Profile shall obey the following rules in handling Digital Signatures:

a. The implementation shall store any SOP Instances that it receives in such a way that it guards against any unauthorized tampering of the SOP Instance.

b. Wherever possible, the implementation shall validate the Digital Signatures within any SOP Instance that it receives.

c. If the implementation sends the SOP Instance to another Application Entity, it shall do the following:

1. remove any Digital Signatures that may have become invalid due to any allowed variations to the format of Attribute Values (e.g. trimming of padding, alternate representations of numbers),

2. generate one or more new Digital Signatures covering the Data Elements that the implementation was able to verify when the SOP Instance was received.

### A.3 BIT-PRESERVING DIGITAL SIGNATURES SECURE USE PROFILE

An implementation that stores and forwards SOP Instances may claim conformance to the Bit-Preserving Digital Signatures Secure Use Profile. Any implementation that claims conformance to this Security Profile shall obey the following rules in handling Digital Signatures:

a. The implementation shall store any SOP Instances that it receives in such a way that when the SOP instance is forwarded to another Application Entity, the Value fields of all Attributes are bit-for-bit duplicates of the fields originally received.

b. The implementation shall not change the order of Items in a Sequence.

c. The implementation shall not remove or change any Data Element of any SOP Instance that it receives when sending that SOP Instance on to another Application Entity via DICOM. This includes any Digital Signatures received.

Note: Implementations may add new Data Elements that do not alter any existing Digital Signatures.

d. The implementation shall utilize an explicit VR Transfer Syntax.

Note: Implementations that cannot use an explicit VR Transfer Syntax cannot conform to this Secure Use Profile, since it may not be able to verify Digital Signatures that are received with an implicit VR Transfer Syntax.

e. The implementation shall not change the VR of any Data Element that it receives when it transmits that object to another Application Entity.

*Item 7.6 Add the following Annex*

# Annex C DIGITAL SIGNATURE PROFILES (Normative)

### C.1 BASE RSA DIGITAL SIGNATURE PROFILE

The Base RSA Digital Signature Profile outlines the use of RSA encryption of a MAC to generate a Digital Signature. This Profile does not specify any particular set of Data Elements to sign. Other Digital Signature profiles may refer to this profile, adding specifications of which Data Elements to sign or other customizations.

The creator of a digital signature shall use one of the RIPEMD-160, MD5, or SHA-1 hashing functions to generate a MAC, which is then encrypted using a private RSA key. All validators of digital signatures shall be capable of using a MAC generated by any of three hashing functions specified (RIPEMD-160, MD5, or SHA-1).

Note: The use of MD5 is not recommended by its inventors, RSA. See: ftp://ftp.rsasecurity.com/pub/pdfs/bulletn4.pdf

The MAC to be signed shall be padded to a block size matching the RSA key size, as directed in RFC 2437 (PKCS #1). The Value of MAC Algorithm (0400,0015) shall be set to either "RIPEMD160", “MD5”, or “SHA1”. The public key associated with the private key as well as the identity of the Application Entity or equipment manufacturer that owns the RSA key pair shall be transmitted in an X.509 (1993) signature certificate. The Value of the Certificate Type (0400,0110) Attribute shall be set to "X509_1993_SIG". A site-specific policy determines how the X.509 certificates are generated, authenticated, and distributed. A site may issue and distribute X.509 certificates directly, may utilize the services of a Certificate Authority, or use any reasonable method for certificate generation and verification.

If an implementation utilizes timestamps, it shall use a Certified Timestamp Type (0400,0305) of “CMS_TSP”. The Certified Timestamp (0400,0310) shall be generated as described in “Internet X.509 Public Key Infrastructure; Time Stamp Protocols; March 2000”.

## C.2 CREATOR RSA DIGITAL SIGNATURE PROFILE

The creator of a DICOM SOP Instance may generate signatures using the Creator RSA Digital Signature Profile. The Digital Signature produced by this Profile serves as a lifetime data integrity check that can be used to verify that the pixel data in the SOP instance has not been altered since its initial creation. An implementation that supports the Creator RSA Digital Signature Profile may include a Creator RSA Digital Signature with every SOP Instance that it creates; however, the implementation is not required to do so.

As a minimum, an implementation shall include the following attributes in generating the Creator RSA Digital Signature:

a. the SOP Class and Instance UIDs
b. the SOP Creation Date and Time, if present
c. the Study and Series Instance UIDs
d. any attributes of the General Equipment module that are present
e. any attributes of the Overlay Plane, Curve or Graphic Annotation modules that are present
f. any attributes of the General Image and Image Pixel modules that are present
g. any attributes of the SR Document General and SR Document Content modules that are present
h. any attributes of the Waveform and Waveform Annotation modules that are present

The Digital Signature shall be created using the methodology described in the Base RSA Digital Signature Profile. Typically the certificate and associated private key used to produce Creator RSA Digital Signatures are configuration parameters of the Application Entity set by service or installation engineers.

Creator RSA Digital Signatures bear no direct relationship to other Digital Signatures. However, other Digital Signatures, such as the Authorization Digital Signature, may be used to collaborate the timestamp of a Creator RSA Digital Signature.

## C.3 AUTHORIZATION RSA DIGITAL SIGNATURE PROFILE

The technician or physician who approves a DICOM SOP Instance for use may request the Application Entity to generate a signature using the Authorization RSA Digital Signature Profile. The Digital Signature produced serves as a lifetime data integrity check that can be used to verify that the pixel data in the SOP instance is the same that the technician or physician saw when they made the approval.

As a minimum, an implementation shall include the following attributes in generating the Authorization RSA Digital Signature:

a. the SOP Class and Instance UIDs
b. the Study and Series Instance UIDs
c. any attributes whose Values are verifiable by the technician or physician (e.g., their Values are displayed to the technician or physician)
d. any attributes of the Overlay Plane, Curve or Graphic Annotation modules that are present
e. any attributes of the General Image and Image Pixel modules that are present
f. any attributes of the SR Document General and SR Document Content modules that are present
g. any attributes of the Waveform and Waveform Annotation modules that are present

The Digital Signature shall be created using the methodology described in the Base RSA Digital Signature Profile. The Application Entity shall determine the identity of the technician or physician and obtain their certificate through a site-specific procedure such as a login mechanism or a smart card.

Authorization RSA Digital Signatures bear no direct relationship to other Digital Signatures. However, other Digital Signatures, such as the Creator RSA Digital Signature, may be used to collaborate the timestamp of an Authorization RSA Digital Signature.