CDA 文書暗号化規格

ver. 1.02

日本 HL7 協会

改訂履歴

版	改訂日附	重要度	改訂箇所	内容
1.00	2006 年 3 月 24 日	-	-	初版
1.01	2006 年 4 月 20 日	低	３．５	韓国情報　→　韓国の
1.02	2006 年 5 月 12 日	低	３．５ まえがき ４．２．１	韓国情報　→　韓国の（前版未修正） 10 行目　強度を用いる →　強度を、用いる 6 行目　7.4.2　→　4.4.2

目次

まえがき

昨今、患者あるいは家族から診療の把握、あるいはセカンドオピニオンを得たいという要求が高まってきている。また、医療・介護機関の機能分化と施設間連携を通じた医療・介護の効率向上が求められている。これらの要求を満たすために、適切に電子化された診療に関する情報を提供あるいは交換することが求められ、それらの文書の規格として HL7 CDA が存在する。CDA に準拠した文書は紙などの物理媒体と比べて大量の情報を含めることが可能で、一定の程度の安全性確保をすることがのぞましい。本規格は CDA 文書に安全性確保の目的で暗号化する場合の暗号化の手段を記述するものである。なお、この規格は暗号化を強制するものではなく、暗号化を行うかどうかは他の規格やドメインでの規約などで強制されない限り、CDA 文書を作成し、使用する当事者が自由に決めることができる。また暗号化の強度を、用いる暗号の最高強度以下の任意のレベルに自由に設定できるように、鍵長を 16 octet より短く設定した場合のパディングルールを含めている。

ただし、鍵長を 16 octet より短く設定した場合はパディングルールを公開しているために、この規格が採用している暗号化方式の本来の強度が期待できないことに十分留意して使用する必要がある。

なお、本規格の目標は、以下のとおりである。

(1) 患者ケア情報の提供に重点をおく
(2) システムを低コストで実現できるようにする
(3) 記述された情報は、再利用の可能性があるものとする
(4) さまざまなドキュメント生成アプリケーションで互換性をもつようにする
(5) その文書の転送方式や格納のメカニズムとは独立である
(6) すみやかに設計書を提供する
(7) オープンな標準を使う

1. 適用

本規格は、特定の目的を持つ CDA 文書を可搬電子媒体に記録し使用する場合にデータを暗号化する際の仕様に適用する。暗号化を行うかどうかは本規格では規定しない。

2. 引用規格

本規格は、以下の規格を引用する。日付の付いた引用規格については、その日付以降に発行された修正票または改訂版は参照しない。しかしながら、本仕様書を使用する場合は、各規格の最新版を調べて適用するよう努めなければならない。

ISO/IEC 18033-3 :2005 Information technology -- Security techniques -- Encryption algorithms -- Part 3: Block ciphers

XML Encryption Syntax and Processing (W3C 2002)

患者診療情報提供書規格 V1.00 （日本 HL7 協会 2006）

CDA 文書電子署名規格 V1.00 （日本 HL7 協会 2006）

3. 用語と定義

本規格では、以下の用語と定義が適用される。

3.1 可搬電子媒体

持ち運び可能な記録媒体のこと。CD-R、DVD、MO などを指す

3.2 共通鍵暗号

共通鍵暗号方式(Common key cryptosystem)とは、暗号化と復号に同一の鍵を用いる暗号方式である。秘密鍵暗号方式（secret key cryptosystem）や対称鍵暗号（Symmetric key encryption scheme）とも呼ばれる。

3.3 ブロック暗号

ブロック暗号（Block cipher）とは、共通鍵暗号の一種で、ブロックと呼ばれる固定長のデータを単位として暗号化復号を行う暗号である。これに対して、ビット単位やバイト単位で暗号化を行う暗号はストリーム暗号と呼ばれる。

3.4 AES

AES は、 アメリカの国家新標準暗号規格（Advanced Encryption Standard）で規格

化された共通鍵暗号方式である。 1977 年に発行された暗号規格 DES が技術進歩により時代遅れとなったため、新たな暗号方式の公募を行い、2001 年 3 月に FIPS PUB 197 として公表された。

3.5 SEED

SEED は 1998 年から韓国の Korea Information Security Agency と専門家のグループによって開発された 128 ビットの共通鍵ブロック暗号である。

3.6 Camelia

NTT と三菱電機で開発された共通鍵ブロック暗号である。鍵長は 128 ビット、192 ビット、256 ビットを選択できる。

3.7 ファイル名称

ファイルを可搬電子媒体上で一意に指し示すことができるラベル

3.8 拡張子

ファイルの属性等を示すために使われるファイル名称の部分文字列、JIS X 0606 のファイル拡張名など

4. 暗号化

可搬電子媒体には個人情報、および診療情報が含まれる。媒体は紛失、盗難の恐れがあるため、データの暗号化を行うことが望ましい。可搬電子媒体に含まれる診療情報ファイルを暗号化する場合は個別に暗号化を行う事とする。暗号化する診療情報には CT のシリーズ画像等も含まれる事があり、暗号化する電子ファイルの数が膨大となる可能性があるため IC カードでの暗号化の処理を必須としない。暗号化を行った電子ファイルの情報はデータを暗号化する際にログとして記述し、暗号化したファイルを復号する際に使用する。このファイル名を CRYPTLOG.XML とする。CRYPTLOG.XML は本規格による暗号化を行ってはならない。

また、可搬電子媒体に CDA 文書およびそこから参照されている情報以外のデータファイルやビュアソフトなどのプログラムファイルが含まれる場合には本規格による暗号化を行ってはならない。

4.1 要求事項

- 電子診療情報提供書には個人情報および診療情報が含まれるため、電子診療情報提供書に対するセキュリティを確保する必要がある。必要なセキュリティとして個人情報保護に関する法律や関連するガイドラインに適合する必要がある。

・ 紹介元と紹介先の機器のＯＳが異なっていてもデータが性格に伝達されなければならない。そのため、暗号化についてはファイルシステムに依存しない方法をとる。

4.2 データの暗号化

4.2.1 暗号アルゴリズム

暗号化に使用するアルゴリズムは 128 ビット共通鍵ブロック暗号方式とする。、利用できる暗号方式としては、ISO/IEC18933Part-2:2005 で規定されている以下の 3 方式とする。なお、鍵長・ブロック長は 128bit 固定とする。また、ユーザのセキュリティポリシにあわせて、最小4octet(32bit)から最大16octet(128bit)まで1octet 単位で任意の長さの鍵長の「暗証番号」を使用することができるものとする。暗証番号の長さが 16 octet より短い場合は、4.4.2 に述べる方法でパディングし鍵長を 16octet に拡張する。暗証番号はユーザが生成しシステムに入力する場合と、システムがランダムに生成する場合を選択できるように構成することが望まれる。システムによっては、入力または生成する暗証番号を数字のみや英数字と特殊文字などのように限定してもよい。

表4．1 利用可能な暗号

項	暗　号	拡張子(利用可能な場合)
1	AES	AES
2	SEED	SED
3	Camelia	CAM

・ 鍵長: 128bit
・ ブロック長: 128bit

4.2.2 暗証番号のパディング・アルゴリズム

128bit の秘密鍵の LSB から順に 0octet,1octet,……15octet と各オクテットを命名する。ユーザ又はシステムが生成した暗証番号の長さを n octet(4≦n≦16)とする。暗証番号は、

$$8 - 〔n / 2〕 \ (\text{octet}) から \quad 7 + 〔(n + 1) / 2〕 \ (\text{octet})$$

に配置する。

ただし〔〕オペーレータは、小数点以下を切り捨て整数化するオペレーションを意味するものとする。

暗証番号の上位 octet を 0xff でパディングする。また、暗証番号の下位 octet を 0x00 でパディングする。

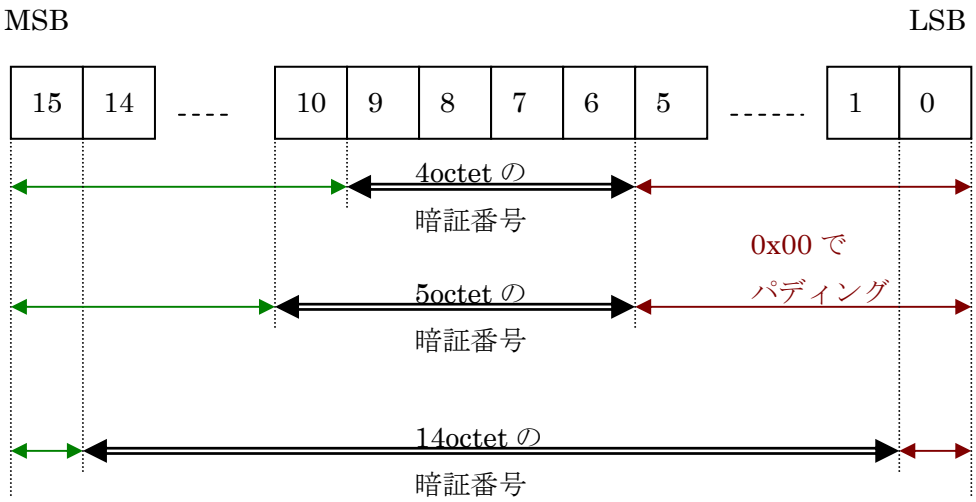

図　4.1 パディングの例

4.2.3　暗号処理後の媒体構成

暗号化処理は CDA 文書およびそれから参照されるファイルを対象とする。暗号化処理を行った後のファイル名称は暗号化処理前のファイル名称と区別できるもとし、CRYPTLOG.XML にファイル名称を記載する。可搬電子媒体に含まれるファイルのディレクトリ構成は暗号処理を施す前のものと同様とする。図４．２に例を示す。

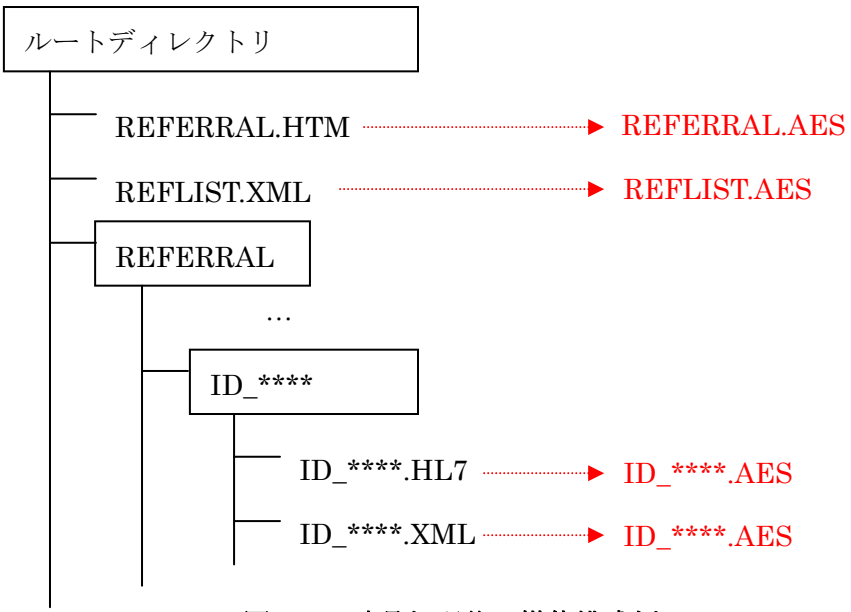

図　4.2 暗号処理後の媒体構成例

4.2.4 暗号化ログファイル

暗号化を行ったファイルの情報はデータを暗号化する際にログとして、ファイル名とその属性(ファイルサイズ、タイムスタンプ、暗号アルゴリズム等)を CRYPTLOG.XML ファイルに記述する。このファイルは全ての暗号処理が終了した時点で書き込み、復号化の際に使用する。ファイルに記述する項目については表 4.2 を参照のこと。

なお、CRYPTLOG.XML はできる限りトップディレクトリに置くことが望ましいが、媒体ファイル仕様により最上位ディレクトリに置かない場合は、別途各媒体ファイル構造規格で規定されたい。(CRYPTLOG.XML の XML スキーマについては表4．3を参照)

表 4.2　CRYPTLOG.XML の内容

項目	内容
データ暗号日時	暗号処理を開始した日時 （yyyy/mm/dd:HH:MM:SS）
暗号化ファイル数	媒体に含まれる暗号化処理を行ったファイルの数
暗号アルゴリズム	暗号処理に使用したアルゴリズム名称
オリジナルファイル名称	暗号処理前のファイル名称（媒体のルートフォルダからのパス情報も含む）
暗号後のファイル名称	暗号処理後のファイル名称（媒体のルートフォルダからのパス情報も含む）
暗号処理のステータス	暗号処理の成否

4.2.5 暗号化の解除方法について

暗号化された媒体はどのような環境でも暗号化解除できることが望まれる。そのため、復号化の処理に使用する暗証番号は容易に取得でき、かつ安全に保管されなければならない。暗証番号は診療情報を保存する媒体とは別の媒体－紙媒体等のに暗号化の際に記録する。暗証番号を記録する媒体は診療情報媒体とは別の場所に保管され、復号処理の際にのみ使用される事が望まれる。暗証番号を記録する媒体は紛失・破壊の恐れがあるため、媒体に記録すると同時に必ずバックアップを保存しておくこと。

暗号化解除を行った後にデータを書き込む媒体は、暗号化された媒体とは別の CD-R(DVD-R)、FD 等の媒体、又は復号処理を行う PC のローカルハードディスクで提供される。復号化の際に媒体を別途用意する場合は媒体に、別媒体で用意する場合は別媒体に、暗号化に関する情報、および暗号化を解除するための手順書を格納あるいは記載されることが望まれる。また、暗号化に関する免責を記載することが望まれる。

※媒体を上書きするためには、ライティングソフト等が必要になることが想定される。PC によってはライティングソフトが利用できないことも考えられるため、暗号化した情報をローカルハードディスクに展開して参照するといった運用も可能とする。

表 4.3 CRYPTOLOG.XML ファイルーXML スキーマ

```
<<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <xsd:element name="EncryptInfo" type="EncryptInfo_type"/>
        <xsd:element name="FileInfo" type="FileInfo_type"/>
        <xsd:element name="File" type="File_type"/>
        <xsd:complexType name="EncryptInfo_type">
                <xsd:sequence>
                        <xsd:element name="Date" type="xsd:dateTime"/>
                        <xsd:element name="FileCount" type="xsd:integer"/>
                        <xsd:element name="Algorithm" type="xsd:string"/>
                </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="FileInfo_type">
                <xsd:sequence>
                        <xsd:element ref="File" minOccurs="0" maxOccurs="unbounded"/>
                </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="File_type">
                <xsd:sequence>
                        <xsd:element name="OriginalFileName" type="xsd:string"/>
                        <xsd:element name="EncryptFileName" type="xsd:string"/>
                        <xsd:element name="EncryptStatus" type="xsd:string"/>
                </xsd:sequence>
        </xsd:complexType>
</xsd:schema>
```