資料１

厚生労働省 ひと、くらし、みらいのために Ministry of Health, Labour and Welfare

第２７回保健医療福祉分野における公開鍵基盤認証局の整備と運営に関する専門家会議・専門作業班合同会議

厚生労働省 医政局 参事官（特定医薬品開発支援・医療情報担当）付医療情報室

Ministry of Health, Labour and Welfare of Japan

１．CA鍵更新に関して

審議：HPKIにおける鍵更新の運用方法について

審議：日本医師会認証局 鍵更新の再延期について

審議：認証局ポリシの改定について

２．連絡：次回（第２８回）の予定について

１．CA鍵更新に関して

審議：HPKIにおける鍵更新の運用方法について

審議：日本医師会認証局 鍵更新の再延期について

審議：認証局ポリシの改定について

２．連絡：次回（第２８回）の予定について

保健医療福祉分野PKIにおける鍵更新の運用について

前回（第２６回）における議論での決定事項

- HPKIにおいて、サブCAの鍵更新ではリンク証明書の使用は行わないこととする

現状の課題

１．HPKIポリシにおいてサブCAの鍵更新方法が明確になっていない
　　⇒あるべき形は何か（審議事項１）
２．電子処方箋において一部ベンダーのモジュールが鍵更新に対応をしていない
　　⇒１の議論の前提として運用可能なものが望ましい

審議事項

１．HPKIにおいてサブCAの使用期限満了に伴う鍵更新はHPKIポリシに準拠することをどの形で認めるか

前回議論内容を整理した案（次ページ以降）を中心にHPKIとして望ましい手法についてご議論おねがいします。
下記４パターンを基準に結論を出していただければと思います。

A	**鍵の運用切換え** CAシステムは更新せず、同一のIssuer nameのCAとして、ルート認証局より新たな証明書を発行、鍵の切換えを行い、旧鍵は廃棄、旧CAで発行済みEEへのCRLは新CAより発行
B	**新鍵に合わせてCAの新規構築** 別のCAシステムを新たに構築し（Issuer name も異なる）ルート認証局より新たな証明書を発行、旧CAで発行済みEEの有効期間はCA移行期間としてCRLへの署名のみ旧CAを継続運用する。
C	**上記A,BともにHPKIポリシとして許容可能**
D	**上記A,BともにHPKIポリシとして許容できない** 他の手法を提示（OCSPの利用等）

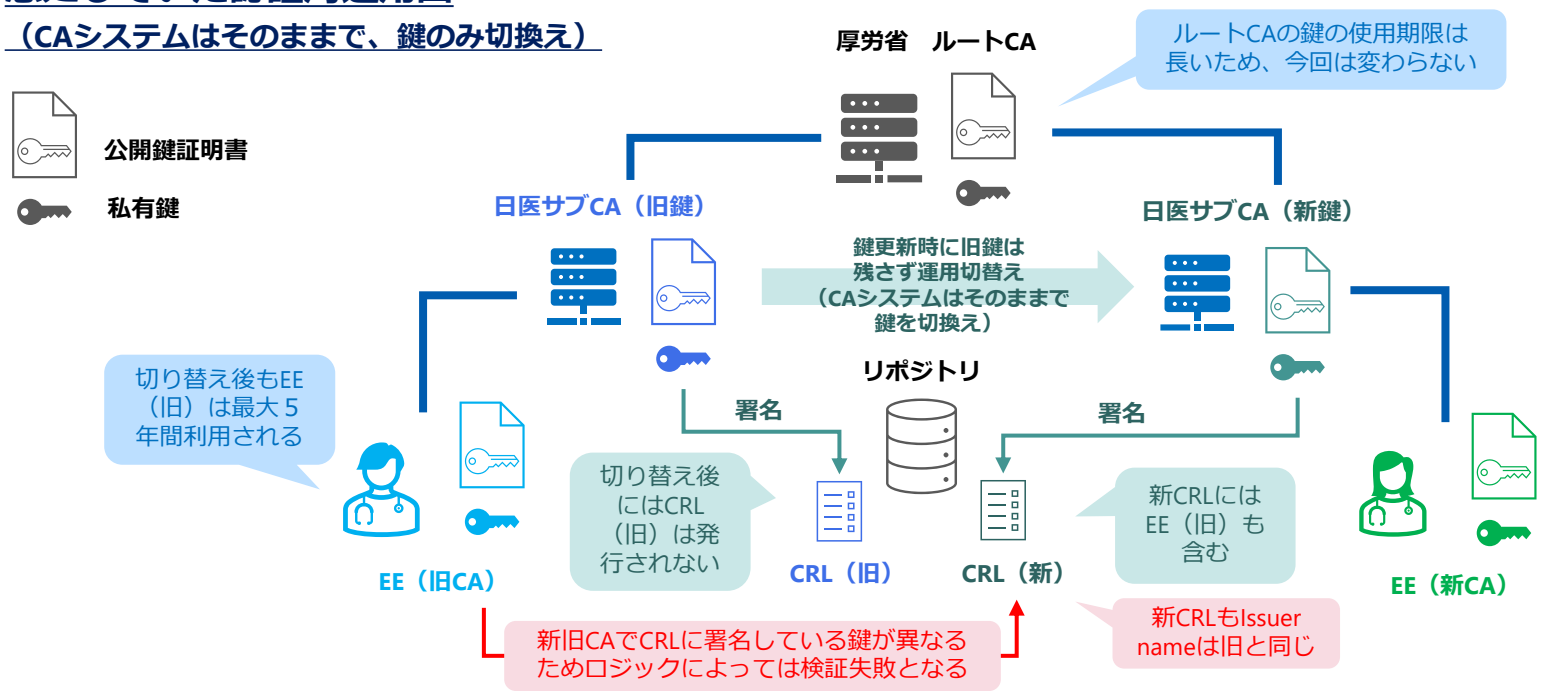
保健医療福祉分野PKIにおける鍵更新の運用について

背景と発生事象

- 日本医師会認証局におけるサブCA鍵更新実施にあたり模擬環境にておいて旧EE（旧サブCAからの発行）の検証がうまくいかないことが発覚　※新EEは問題なし
- 想定していた鍵更新の仕様としては旧EEに対して、新サブCAの稼働に伴い、旧サブCAはCRL発行を停止し、新サブCAより旧EEも含んだCRLも発行する仕様
- 今回の仕様で鍵更新を行った場合、旧EEに対しても新CAで署名されたCRLを発行するため、旧EEの証明書検証を一体的に実施するロジックにしている場合は新旧CAの鍵が異なるため検証処理に失敗してしまう。
- 電子処方箋では多くのベンダーがCryptoAPIなどを使用しており、その場合、鍵更新を意識しない実装では一体的に検証してしまうため、上記挙動で検証に失敗してしまう。

想定していた認証局運用図

（CAシステムはそのままで、鍵のみ切換え）



前回（第26回）の会議で議論となったA案、B案を元にサブCA鍵更新の方法としては下記の２案を議論の土台として整理

【A】鍵の運用切換え

- 旧サブCAは残さず新サブCAに運用切り替え（新サブCAの証明書をルートから発行し切り替え）
- 旧サブCAと新サブCAのDistribution Pointは同一とする
- EE証明書とCRL証明書のissuer nameを比較し同一性を検証する

【B】CAの新規構築

- 新たに新サブCAを立てる（証明書もルートから発行）
- 旧サブCAが発行したEE証明書の期限内は旧サブCAを残しCRLを発行し続ける
  （旧CAでのEE証明書の発行は停止）
- ポリシーの改定により使用期限終了後もCRL発行は認める規定を追加する必要あり
- 旧サブCAと新サブCAのDistribution Pointは異なる値とする
- 旧サブCAと新サブCAのissuer nameは異なる値とする

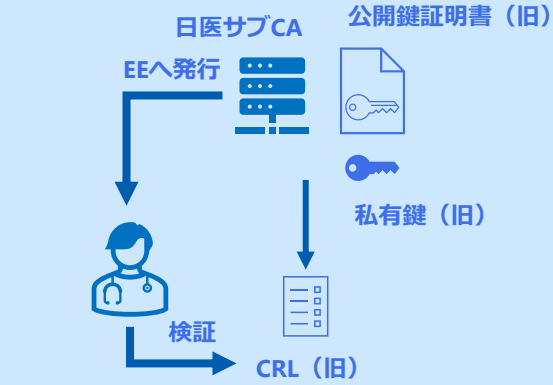
	【A】鍵の運用切換え	【B】CAの新規構築
RFC5280への準拠	RFC5280の解釈による	RFC5280の解釈による
Issuer name	旧CAと新CAで同一	旧CAと新CAは異なる
旧鍵の扱い	運用切り替え後に廃棄	旧EEの有効期限までCRL署名に利用継続 移行期間終了後に廃棄
パス検証	対応不要	新旧EE証明書を検証できるよう新旧サブCA証明書を配布・設定する必要あり
失効リスト検証	EE証明書と一体的に証明書検証しないように設定・改修する必要あり（RFC5280-6.3.3の通り実装）	新旧CRLを検証できるよう新旧サブCA証明書を配布・設定する必要あり
認証局対応	更新時の切替え対応のみ	CA移行期間の負担大
ベンダー（検証者）対応	現行ベンダーのほとんどが改修対応必要の見込み	新旧サブCA証明書を配布・設定する必要あり
ポリシー改定	対応不要と想定（議論内容を踏まえ確認）	必要
対応期間	ベンダ対応に半年程度	新CA立ち上げに１年程度
電子処方箋への影響	更新期限の延長が許容された場合は影響なし	更新期限までに新CAが立ち上がらない場合は新規EEの受付を一時停止
共通事項	検証の考え方や方式がベンダ毎に異なった実装とならないように、署名や検証に係る実装ガイド等を作成する必要がある	

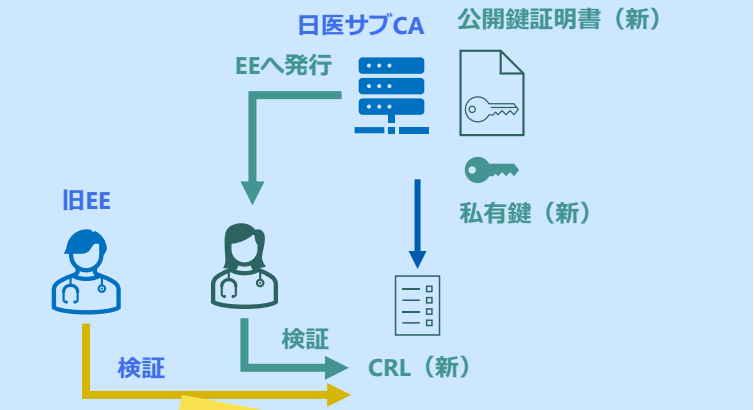
鍵更新前後での構成イメージ

鍵更新前

鍵更新後

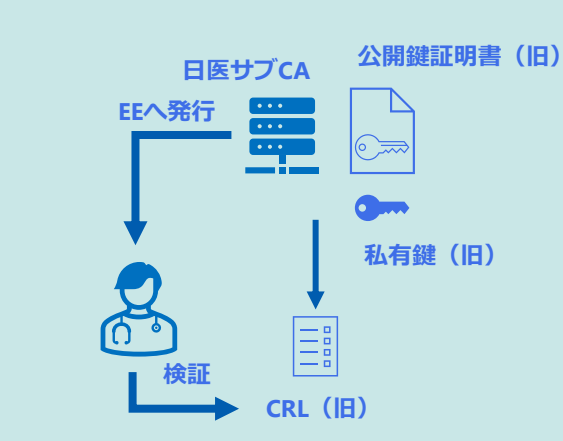
【A】
鍵の運用切換え

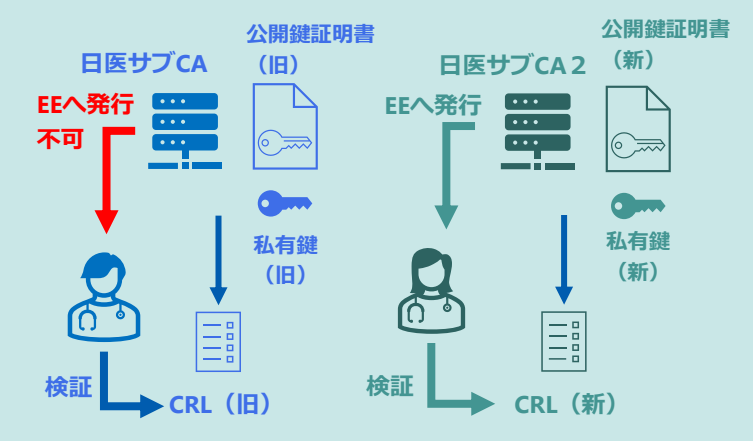




多くの署名モジュールベンダは
改修必要

【B】
CAの新規構築





旧EEの有効期限までCAの
並行運用が必要

【参考情報】暗号方式の移行

HPKIで使用している方式

ハッシュ：SHA-256（128ビットセキュリティ）　2050年まで

暗号：RSA2048（112ビットセキュリティ）　2030年まで

2030年の暗号方式移行の際はCA更新にて対応の見込み

表 5　セキュリティ強度要件の基本設定方針

想定運用終了・廃棄年／利用期間		2022～2030	2031～2040	2041～2050	2051～2060	2061～2070
112 ビット セキュリティ	新規生成 ((a)参照)	移行完遂期間 ((c)参照)	利用不可	利用不可	利用不可	利用不可
	処理 ((b)参照)		許容			
128 ビット セキュリティ	新規生成 ((a)参照)	利用可	利用可	移行完遂期間 ((c)参照)	利用不可	利用不可
	処理 ((b)参照)				許容	
192 ビット セキュリティ	新規生成 ((a)参照)	利用可	利用可	利用可	利用可	利用可
	処理 ((b)参照)					
256 ビット セキュリティ	新規生成 ((a)参照)	利用可	利用可	利用可	利用可	利用可
	処理 ((b)参照)					

Cryptrec 暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準より抜粋

暗号移行時の対応

過去対応

SHA-1からSHA-256への移行時はルート、サブ共にCAを二つ立てて並行運用を実施した。（CAごと更新のパターン）

今後の対応見込み

RSA2048からの移行についても同様にCAを二つ立てて並行運用となる見込み（A案の場合は現CAシステムが新暗号に対応している必要があるが、利用出来ない可能性が高いため）

次期暗号の方式について

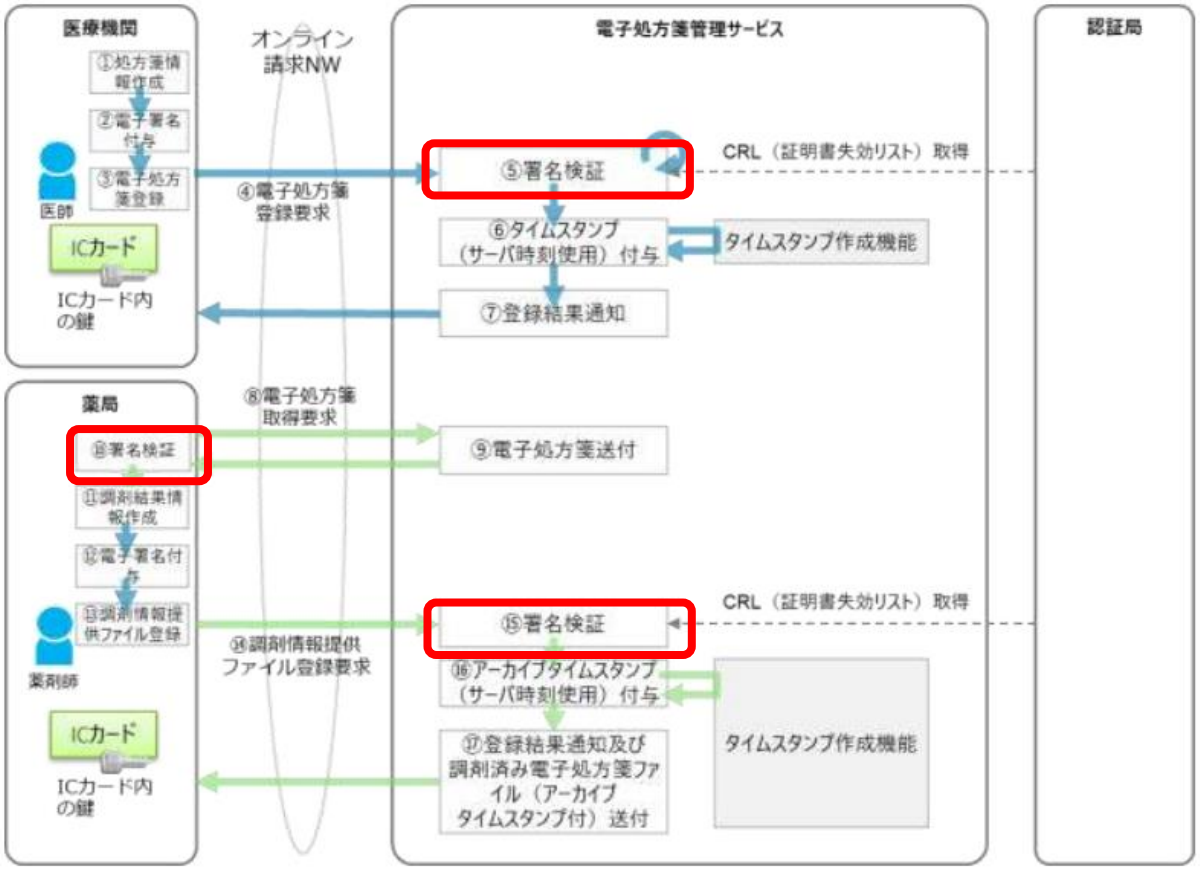
2030年に向けて適宜議論を進めていく

【参考情報】ベンダヒアリング結果

詳細は医薬局より説明させていただきます。

【参考情報】電子処方箋における署名検証

図 5 電子署名の付与及び署名検証のイメージ（ローカル署名）



電子処方箋　技術解説書より抜粋

【A】鍵の運用切り替えの場合、各個別ベンダーが対応必要となるのは薬局における検証時

【審議２】鍵更新期日の延長について

審議事項

２．日本医師会認証局の鍵更新期日の延長可否をご審議お願いします。

A案想定でのベンダヒアリングの結果、最大で年内対応となるベンダが存在している状況です。
B案想定でも期間は必要なため、ポリシ上の最大期間まで再延期をご審議お願いします。

2024年1月16日（当初期限）
⇒　2024年7月16日（延長期限）
⇒　2025年1月16日（ポリシ上の延長期限）

CA名（SHA256)	開始日	公開鍵終了日	私有鍵終了日
厚労省ルートCA	2014年1月17日	2044年1月16日	2029年1月16日
日医署名サブCA	2014年1月17日	2034年1月16日	**2024年1月16日**
日医認証(人)サブCA	2014年1月17日	2034年1月16日	**2024年1月16日**
MEDIS署名サブCA	2015年4月20日	2035年4月19日	**2025年4月19日**
MEDIS認証(人)サブCA	2017年3月17日	2037年3月16日	**2027年3月16日**

【審議３】HPKI認証局ポリシの改訂について

審議事項

３．認証局ポリシの改定（詳細後述）

- 前回（26回）で承認されたリンク証明書利用に係わる箇所の改定
- B案がHPKIとして許容可能な場合は使用期限に関する箇所を改定

認証局ポリシ改定案

１．リンク証明書について

2.2　証明書情報の公開

＜検証者に公開する事項＞に**「リンク証明書（ルートCAのみ対象）」**を追加

5.5.1　アーカイブ記録の種類

認証局がアーカイブする情報に**「リンク証明書（ルートCAのみ対象）」**を追加

5.6　鍵の切替

「ルートCA私有鍵の更新時にはリンク証明書を発行し、公開する。（サブCA私有鍵の更新時は不要）」の文言を追加

２．CAごと鍵更新を行う場合の証明書有効期限について

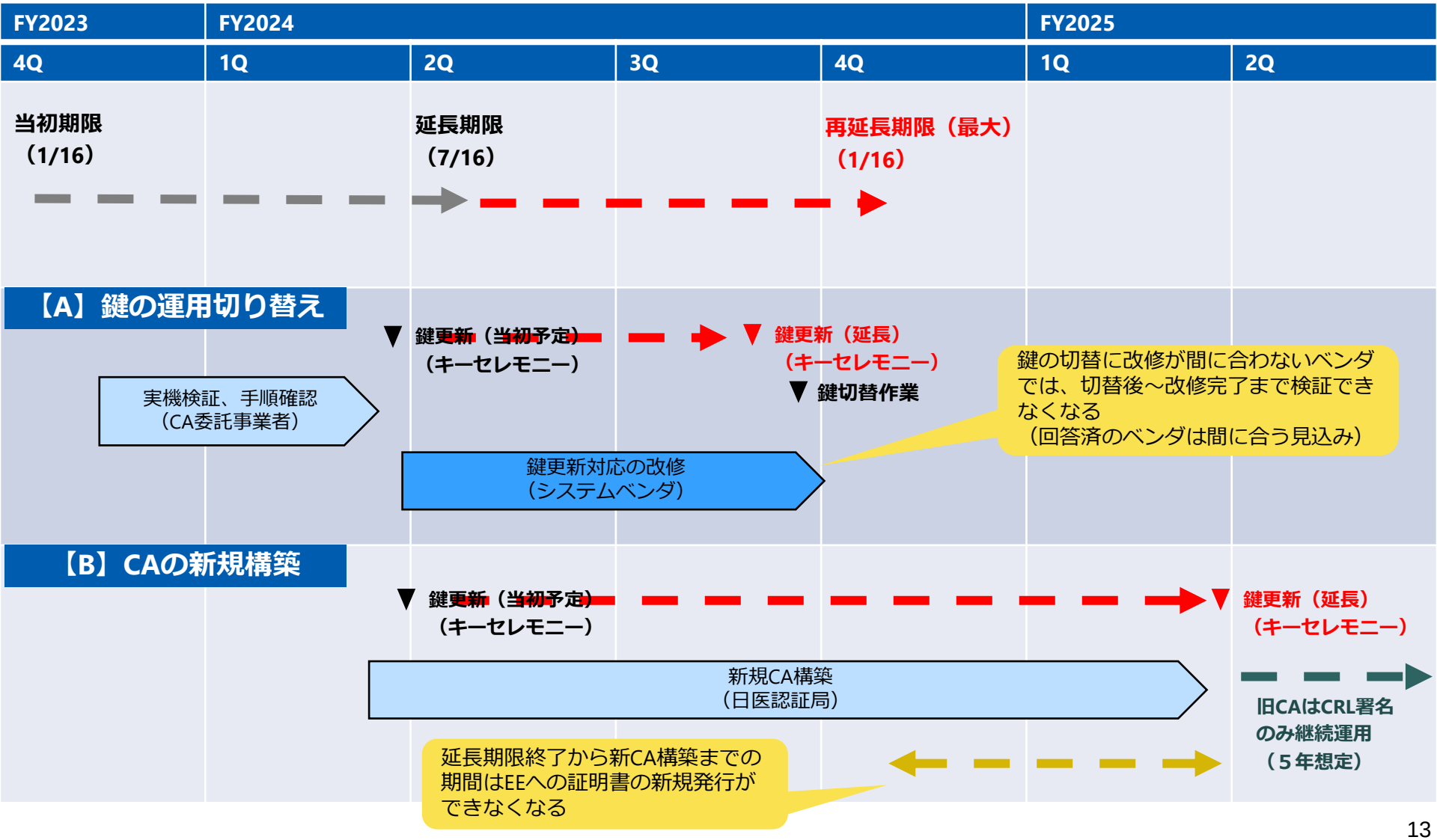
6.3.2　公開鍵証明書の有効期間と鍵ペアの使用期間

（改定案１）CA公開鍵証明書の有効期間は20年を超えないものとし、エンドエンティティの電子証明書発行のための~~その~~私有鍵の使用は10年を超えないものとする。ただし、エンドエンティティの電子証明書発行のための10年の私有鍵の使用期間を延長する必要がある場合にはHPKI認証局専門家会議の承認を得て、最大１年の範囲において必要な期間に限り延長することができる。なお、CRLの署名等、認証局の円滑な運営に必要となる場合は公開鍵の有効期限内に限り私有鍵は利用可能とする。

（改定案２）CA公開鍵証明書の有効期間は20年を超えないものする。また、私有鍵の使用はCA公開鍵証明書の有効期間以内とする。

日本医師会認証局
サブCA鍵更新の再延長について

延長スケジュール案（最大）



【参考】RFC4210 抜粋

サブCAについては対象外となっているが鍵更新について記載されているため参考情報として掲示

RFC4210 4.4 Root CA Key Update（抜粋）

This discussion only applies to CAs that are directly trusted by some end entities. Self-signed CAs SHALL be considered as directly trusted CAs. Recognizing whether a non-self-signed CA is supposed to be directly trusted for some end entities is a matter of CA policy and is thus beyond the scope of this document.

The basis of the procedure described here is that the CA protects its new public key using its previous private key and vice versa. Thus, when a CA updates its key pair it must generate two extra cACertificate attribute values if certificates are made available using an X.500 directory (for a total of four: OldWithOld, OldWithNew, NewWithOld, and NewWithNew).

When a CA changes its key pair, those entities who have acquired the old CA public key via "out-of-band" means are most affected. It is these end entities who will need access to the new CA public key protected with the old CA private key. However, they will only require this for a limited period (until they have acquired the new CA public key via the "out-of-band" mechanism). This will typically be easily achieved when these end entities' certificates expire.

The data structure used to protect the new and old CA public keys is a standard certificate (which may also contain extensions). There are no new data structures required.

Note 1. This scheme does not make use of any of the X.509 v3 extensions as it must be able to work even for version 1 certificates. The presence of the KeyIdentifier extension would make for efficiency improvements.

Note 2. While the scheme could be generalized to cover cases where the CA updates its key pair more than once during the validity period of one of its end entities' certificates, this generalization seems of dubious value. Not having this generalization simply means that the validity periods of certificates issued with the old CA key pair cannot exceed the end of the OldWithNew validity period.

Note 3. This scheme ensures that end entities will acquire the new CA public key, at the latest by the expiry of the last certificate they owned that was signed with the old CA private key (via the "out-of-band" means). Certificate and/or key update operations occurring at other times do not necessarily require this (depending on the end entity's equipment).

この議論は、一部の最終エンティティから直接信頼されるCAにのみ適用されます。自己署名CAは、直接信頼できるCAと見なされます。非自己署名のCAが一部の最終エンティティに対して直接信頼されることになっているかどうかを認識することは、CAポリシーの問題であり、したがって、このドキュメントの範囲を超えています。

ここで説明する手順の基礎は、CAが以前の秘密鍵を使用して新しい公開キーを保護し、その逆も同様です。したがって、CAがキーペアを更新する場合、X.500ディレクトリを使用して証明書が利用可能になった場合、2つの追加のCACERTIFICATE属性値を生成する必要があります（OldWithold、OldWithNew、NewWithold、およびNewWithNew）。

CAがキーペアを変更すると、「帯域外」平均を介して古いCAの公開キーを取得したエンティティが最も影響を受けます。古いCAの秘密鍵で保護されている新しいCA公開キーにアクセスする必要があるのは、これらの最終エンティティです。ただし、限られた期間のみこれを必要とします（「バンド外」メカニズムを介して新しいCA公開キーを取得するまで）。これは通常、これらの最終エンティティの証明書が期限切れになると簡単に達成されます。

新しいCAパブリックキーを保護するために使用されるデータ構造は、標準証明書です（拡張機能も含まれている場合があります）。新しいデータ構造は必要ありません。

注1.このスキームは、バージョン1の証明書でも機能する必要があるため、X.509 V3拡張機能のいずれも使用していません。KeyIdentifier拡張機能の存在により、効率の改善が可能になります。

注2.スキームは、CAがその最終エンティティの証明書の1つの有効期間中にキーペアを複数回更新するケースをカバーするために一般化できますが、この一般化は疑わしい価値のようです。この一般化を持たないということは、単に古いCAキーペアで発行された証明書の有効期間が、古いものの有効性期間の終わりを超えることができないことを意味します。

注3.このスキームは、エンディティが新しいCAの公開キーを取得することを保証します。最新の状態で、所有した最後の証明書の有効期限があります。。他の場合に発生する証明書および/または主要な更新操作は、必ずしもこれを必要とするわけではありません（エンティティのエンティティの機器に応じて）。

【参考】RFC4210 抜粋

サブCAについては対象外となっているが鍵更新について記載されているため参考情報として掲示

RFC4210 4.4.1 CA Operator Actions（抜粋）

To change the key of the CA, the CA operator does the following:

1. Generate a new key pair;

2. Create a certificate containing the old CA public key signed with the new private key (the "old with new" certificate);

3. Create a certificate containing the new CA public key signed with the old private key (the "new with old" certificate);

4. Create a certificate containing the new CA public key signed with the new private key (the "new with new" certificate);

5. Publish these new certificates via the repository and/or other means (perhaps using a CAKeyUpdAnn message);

6. Export the new CA public key so that end entities may acquire it using the "out-of-band" mechanism (if required).

The old CA private key is then no longer required. However, the old CA public key will remain in use for some time. The old CA public key is no longer required (other than for non-repudiation) when all end entities of this CA have securely acquired the new CA public key.

The "old with new" certificate must have a validity period starting at the generation time of the old key pair and ending at the expiry date of the old public key.

The "new with old" certificate must have a validity period starting at the generation time of the new key pair and ending at the time by which all end entities of this CA will securely possess the new CA public key (at the latest, the expiry date of the old public key).

The "new with new" certificate must have a validity period starting at the generation time of the new key pair and ending at or before the time by which the CA will next update its key pair.

CAのキーを変更するには、CAオペレーターが次のことを行います。

1. 新しいキーペアを生成します。

2. 新しい秘密鍵（「新しい」証明書を持つ古い証明書）で署名された古いCA公開キーを含む証明書を作成します。

3. 古い秘密鍵（「古い」証明書を「新しく」証明書）で署名した新しいCA公開キーを含む証明書を作成します。

4. 新しい秘密鍵（「新しい」証明書を持つ「新しい」証明書）で署名された新しいCA公開キーを含む証明書を作成します。

5. これらの新しい証明書をリポジトリおよび/またはその他の手段から公開します（おそらくCakeYupDannメッセージを使用）。

6. 新しいCA公開キーをエクスポートして、エンティティが「帯域外」メカニズム（必要に応じて）を使用してそれを取得できるようにします。

古いCAの秘密鍵は不要になりました。ただし、古いCAの公開キーはしばらくの間使用され続けます。このCAのすべての最終エンティティが新しいCAの公開キーを安全に取得した場合、古いCAの公開キーは（非repudiation以外）要求されなくなりました。

「古い」証明書は、古いキーペアの生成時に始まり、古い公開キーの有効期限で終了する有効期間を持たなければなりません。

「古い」証明書は、新しいキーペアの生成時に始まり、このCAのすべてのエンティティが新しいCA公開キーを安全に所有する時点で終了する有効期間を持たなければなりません（最新の場合、有効期限は有効です古い公開鍵の日付）。

「新しい」証明書は、新しいキーペアの生成時間から始まり、CAが次にキーペアを更新する時期以前に終了するまでの有効期間を持たなければなりません。

【参考】RFC5280 抜粋

RFC5280 4.2.1.4 Issuer（抜粋）

The issuer field identifies the entity that has signed and issued the certificate. The issuer field MUST contain a non-empty distinguished name (DN). The issuer field is defined as the X.501 type Name [X.501]. Name is defined by the following ASN.1 structures:

発行者フィールドは、証明書に署名して発行したエンティティを識別します。発行者フィールドには、空でない識別名(DN)を含める必要があります。発行者フィールドは、X.501タイプの名前[X.501]として定義されています。名前は、次のASN.1構造によって定義されます。

RFC5280 4.2.1.13 CRL Distribution Points（抜粋）

The cRLDistributionPoints extension is a SEQUENCE of DistributionPoint. A DistributionPoint consists of three fields, each of which is optional: distributionPoint, reasons, and cRLIssuer. While each of these fields is optional, a DistributionPoint MUST NOT consist of only the reasons field; either distributionPoint or cRLIssuer MUST be present. If the certificate issuer is not the CRL issuer, then the cRLIssuer field MUST be present and contain the Name of the CRL issuer. If the certificate issuer is also the CRL issuer, then conforming CAs MUST omit the cRLIssuer field and MUST include the distributionPoint field.

cRLDistributionPoints拡張は、DistributionPointのシーケンスです。 DistributionPointは3つのフィールドで構成され、各フィールドはオプションです: distributionPoint、reasons、およびcRLIssuer。これらの各フィールドはオプションですが、DistributionPointは理由フィールドのみで構成してはなりません(MUST NOT)。distributionPointまたはcRLIssuerのいずれかが存在する必要があります。証明書発行者がCRL発行者でない場合、cRLIssuerフィールドが存在し、CRL発行者の名前が含まれている必要があります。証明書の発行者がCRLの発行者でもある場合、準拠するCAはcRLIssuerフィールドを省略し、distributionPointフィールドを含める必要があります。

【参考】RFC5280 抜粋

RFC5280 6.1.1 Inputs　（抜粋）

(d) trust anchor information, describing a CA that serves as a trust anchor for the certification path. The trust anchor information includes:
(1)　the trusted issuer name,
(2) the trusted public key algorithm,
(3) the trusted public key, and
(4) optionally, the trusted public key parameters associated with the public key.

The trust anchor information may be provided to the path processing procedure in the form of a self-signed certificate. When the trust anchor information is provided in the form of a certificate, the name in the subject field is used as the trusted issuer name and the contents of the subjectPublicKeyInfo field is used as the source of the trusted public key algorithm and the trusted public key. The trust anchor information is trusted because it was delivered to the path processing procedure by some trustworthy out-of-band procedure. If the trusted public key algorithm requires parameters, then the parameters are provided along with the trusted public key.

(d) トラストアンカー情報。認証パスのトラストアンカーとして機能するCAを記述します。トラストアンカー情報は次のとおりです。
(1)信頼される発行者名、
(2)信頼される公開鍵アルゴリズム、
(3)信頼される公開鍵、および
(4)オプションとして、公開鍵に関連付けられた信頼される公開鍵パラメータ。
トラストアンカー情報は、自己署名証明書の形でパス処理手順に提供することができます。トラストアンカー情報が証明書の形式で提供される場合、主体者名フィールドの名前が信頼される発行者名として使用され、subjectPublicKeyInfoフィールドの内容が信頼される公開鍵アルゴリズムと信頼される公開鍵のソースとして使用されます。トラストアンカー情報は、信頼できるアウトオブバンドの手順を通じてパス処理手順に配信されるため、信頼されているものとします。信頼される公開鍵アルゴリズムにパラメータが必要な場合、パラメータを信頼される公開鍵とともに提供します。

【参考】RFC5280 抜粋

RFC5280 6.3. CRL Varidation　（抜粋）

This section describes the steps necessary to determine if a certificate is revoked when CRLs are the revocation mechanism used by the certificate issuer. Conforming implementations that support CRLs are not required to implement this algorithm, but they MUST be functionally equivalent to the external behavior resulting from this procedure when processing CRLs that are issued in conformance with this profile. Any algorithm may be used by a particular implementation so long as it derives the correct result.

This algorithm assumes that all of the needed CRLs are available in a local cache. Further, if the next update time of a CRL has passed, the algorithm assumes a mechanism to fetch a current CRL and place it in the local CRL cache.

This algorithm defines a set of inputs, a set of state variables, and processing steps that are performed for each certificate in the path. The algorithm output is the revocation status of the certificate.

このセクションでは、CRLが証明書発行者が使用する失効メカニズムである場合に、証明書が失効しているかどうかを判断するために必要な手順について説明します。CRLをサポートする準拠実装は、このアルゴリズムを実装する必要はありませんが、このプロファイルに準拠して発行されたCRLを処理する場合、この手順から生じる外部動作と機能的に同等でなければなりません。正しい結果が得られる限り、特定の実装で任意のアルゴリズムを使用できます。

このアルゴリズムは、必要なすべてのCRLがローカルキャッシュで使用できることを前提としています。さらに、CRLの次の更新時間が経過した場合、アルゴリズムは、現在のCRLをフェッチしてローカルCRLキャッシュに配置するメカニズムを想定しています。

このアルゴリズムは、パスの各証明書に対して実行される一連の入力、一連の状態変数、および処理ステップを定義します。アルゴリズムの出力は、証明書の失効ステータスです。

RFC5280 6.3.1. Revocation Inputs　（抜粋）

To support revocation processing, the algorithm requires two inputs:

(a) certificate: The algorithm requires the certificate serial number and issuer name to determine whether a certificate is on a particular CRL. The basicConstraints extension is used to determine whether the supplied certificate is associated with a CA or an end entity. If present, the algorithm uses the cRLDistributionPoints and freshestCRL extensions to determine revocation status.

(b) use-deltas: This boolean input determines whether delta CRLs are applied to CRLs.

失効処理をサポートするには、アルゴリズムに2つの入力が必要です。

(a)証明書: アルゴリズムは、証明書が特定のCRLにあるかどうかを判別するために、証明書のシリアル番号と発行者名を必要とします。 basicConstraints拡張は、提供された証明書がCAまたはエンドエンティティに関連付けられているかどうかを判断するために使用されます。存在する場合、アルゴリズムはcRLDistributionPointsおよびfreshestCRL拡張を使用して失効ステータスを決定します。

(b)use-deltas: このブール入力は、 Delta CRLがCRLに適用されるかどうかを決定します。

【参考】RFC5280 抜粋

RFC5280 6.3.3 CRL Processing　（抜粋）

This algorithm begins by assuming that the certificate is not revoked. The algorithm checks one or more CRLs until either the certificate status is determined to be revoked or sufficient CRLs have been checked to cover all reason codes.

(b) Verify the issuer and scope of the complete CRL as follows:
(1) If the DP includes cRLIssuer, then verify that the issuer field in the complete CRL matches cRLIssuer in the DP and that the complete CRL contains an issuing distribution point extension with the indirectCRL boolean asserted. Otherwise, verify that the CRL issuer matches the certificate issuer.

(f) Obtain and validate the certification path for the issuer of the complete CRL. The trust anchor for the certification path MUST be the same as the trust anchor used to validate the target certificate. If a key usage extension is present in the CRL issuer's certificate, verify that the cRLSign bit is set.

(g) Validate the signature on the complete CRL using the public key validated in step (f).

If the revocation status has not been determined, repeat the process above with any available CRLs not specified in a distribution point but issued by the certificate issuer. For the processing of such a CRL, assume a DP with both the reasons and the cRLIssuer fields omitted and a distribution point name of the certificate issuer. That is, the sequence of names in fullName is generated from the certificate issuer field as well as the certificate issuerAltName extension. After processing such CRLs, if the revocation status has still not been determined, then return the cert_status UNDETERMINED.

このアルゴリズムは、証明書が取り消されていないと想定することから始まります。アルゴリズムは、証明書のステータスが取り消されたと判断されるか、すべての理由コードをカバーするのに十分なCRLが確認されるまで、1つ以上のCRLを確認します。

(b)次のように、完全なCRLの発行者と範囲を確認します。
(1)DPにcRLIssuerが含まれる場合、完全なCRLの発行者フィールドがDPのcRLIssuerと一致し、完全なCRLに、indirectCRLブール値がアサートされた発行配布ポイント拡張が含まれていることを確認します。それ以外の場合は、CRL発行者が証明書発行者と一致することを確認します。

(f)完全なCRLの発行者の認定パスを取得して検証します。証明書パスのトラストアンカーは、ターゲット証明書の検証に使用されるトラストアンカーと同じである必要があります。鍵使用目的拡張がCRL発行者の証明書に存在する場合は、cRLSignビットが設定されていることを確認します。

(g)ステップ(f)で検証された公開鍵を使用して、完全なCRLの署名を検証します。

失効ステータスが決定されていない場合は、配布ポイントで指定されていないが証明書発行者によって発行された使用可能なCRLを使用して、上記のプロセスを繰り返します。このようなCRLの処理については、理由とcRLIssuerフィールドの両方が省略されたDP、および証明書発行者の配布ポイント名を想定します。つまり、fullName内の名前のシーケンスは、証明書発行者フィールドと証明書発行者AltName拡張から生成されます。そのようなCRLの処理後、失効ステータスがまだ決定されていない場合は、cert_status UNDETERMINEDを返します。

１．CA鍵更新に関して

審議：HPKIにおける鍵更新の運用方法について

審議：日本医師会認証局 鍵更新の再延期について

審議：認証局ポリシの改定について

２．連絡：次回（第２８回）の予定について

第２８回HPKI専門家会議　予定

開催日時：6月２０日（木）　１０：００～１２：００
場所：AP虎ノ門　Bルーム　（オンライン同時開催）

<u>議題予定</u>

- ルート認証局　準拠性審査結果報告
- 日本医師会認証局　準拠性審査結果報告
- 日本薬剤師会認証局　準拠性審査結果報告
- HPKIリモート署名サービスの利用用途限定の扱いについて