【DNSSEC ガイドライン】

DNSSEC による DNS 応答の認証技術ガイドライン

令和 8 年 2 月 25 日版

目 次

序章 想定読者と用語

1. ドメイン名の利用に関する関係者

ドメイン名が利用可能になる、すなわちドメイン名に関連付けられた情報を取り出す名前解決が可能になるまでには、①ドメイン名を登録する、②ドメイン名の情報を権威 DNS サーバーで公開し、その権威 DNS サーバーの情報を登録する、③DNS クライアントから問合わされたドメイン名の情報をフルリゾルバーが権威 DNS サーバー群に問い合わせ、得た結果をクライアントに返すという 3 段階が必要となります。それぞれの段階で、様々な組織が作業に関与します。まず関係者を整理してみましょう。

①ドメイン名を登録する段階

ある組織（または個人）がドメイン名を申請して登録してもらう段階です。例えば、A さんが”example.jp”というドメイン名を申請して、jp ドメイン名として登録してもらいます。この段階では、以下の 2 つの関係者が登場します。

ドメイン名登録者

申請して登録されたドメイン名を維持管理するとともに、そのドメイン名を利用したサービスを提供する組織（または個人）を指します。

ドメイン名登録事業者

トップレベルドメイン（Top Level Domain、TLD）管理組織（レジストリ）とドメイン名登録者の間に入り、ドメイン名登録に関する窓口業務を行う組織です。.com や.net、.org などの gTLD ではレジストラと呼ばれます。また、ドメイン名登録事業者とドメイン名登録者の間に、さらにリセラー（再販事業者・取次事業者）が入る場合もありますが、これも広い意味でドメイン名登録事業者に含まれるものとして扱います。この組織が提供する窓口業務には、DNSSEC に関係するものも含まれることに注意してください。

②ドメイン名の情報を権威 DNS サーバーで公開し、その情報を登録する段階

例えば example.jp というドメイン名の登録後に、www.example.jp という名前の IP アドレスを名前解決できるようにする段階です。この段階では、example.jp ドメイン名の情報を管理する権威 DNS サーバーを構築・運用し、その情報をレジストリに登録する必要があります。

権威 DNS サーバー運用者

ここで、権威 DNS サーバーの運用には様々な形態があることに注意してください。ドメイン名登録者が自前でサーバーを構築する形態、ドメイン名登録事業者が提供する権威 DNS サーバーサービスを利用する形態、権威 DNS サーバーの構築運用を第三者にアウトソーシングする形態などが考えられます。本ガイドラインでは、権威 DNS サーバーを運用しているエンティティ全てを総称して「権威 DNS サーバー運用者」と呼ぶことにします。

また、権威 DNS サーバーによって公開されるデータ（例えば www.example.jp の IP アドレスはこれである、という対応データ）を作成する責任はドメイン名登録者にあることにも注意してください。

③DNS クライアントから問合わされたドメイン名の情報をフルリゾルバーが検索し、権威 DNS サーバーから得た情報をクライアントに返す段階

フルリゾルバー運用者

②で構築した example.jp ドメイン名のデータベースが実際に「使われる」段階です。例えば、www.example.jp の Web ページを閲覧するエンドユーザーは、Web サーバー www.example.jp の IP アドレスを知る必要がありますが、この作業は外部のフルリゾルバーと呼ばれる、別のサーバーに委託しています。フルリゾルバーには通常、エンドユーザーが契約している ISP が運用しているものが設定されます。（Google や Cloudflare などの事業者が提供するパブリックリゾルバーというものも存在しますが、本ガイドラインでは説明を省略します）。フルリゾルバーを運用しているエンティティを本ガイドラインでは「フルリゾルバー運用者」と呼ぶことにします。

2. 本ガイドライン・手引書の想定読者

本ガイドラインは、ドメイン名を様々なインターネット上のサービスに結びつける基盤技術であるドメイン名システム(DNS)の関係者に向けて書かれています。その目的は、ドメイン名に関連付けられた情報を DNS セキュリティ拡張(DNSSEC)によって保護することです。

1 節で記述した関係者をあらためて整理してみましょう。

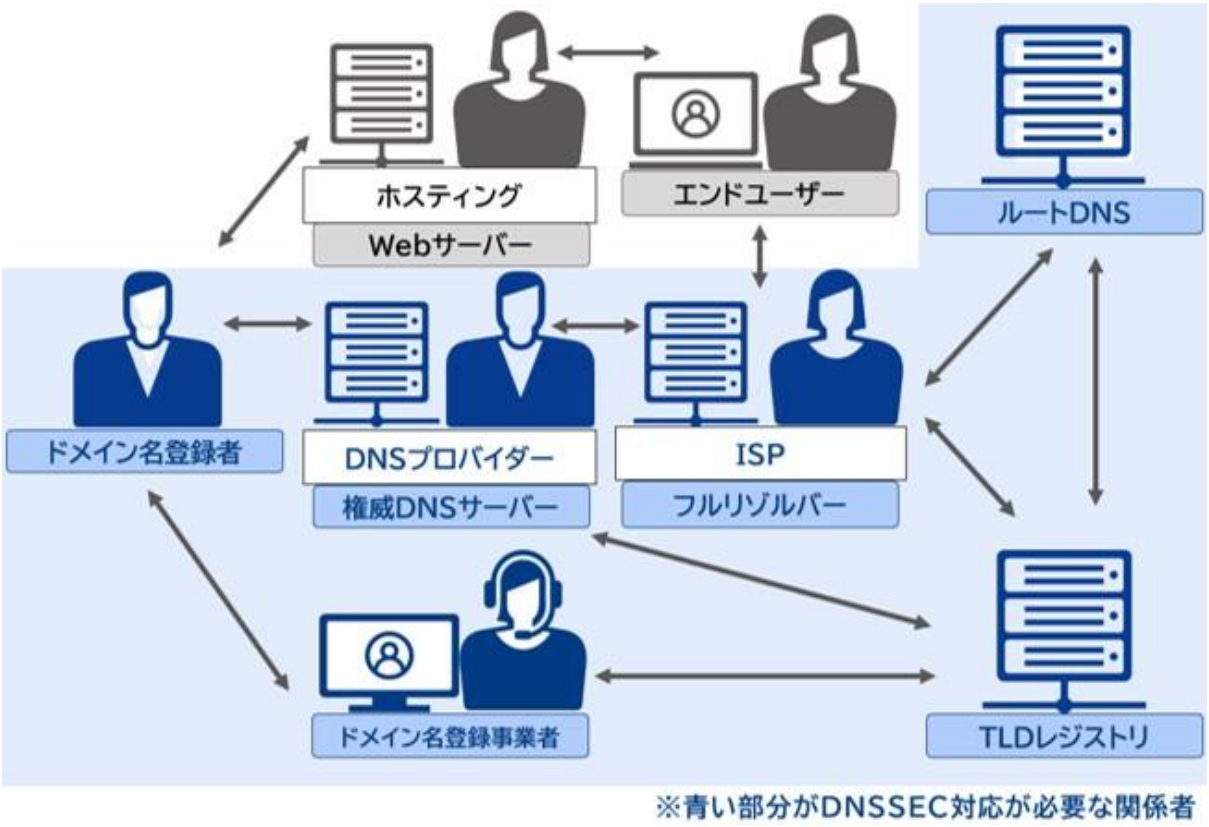

図 1 DNSSEC 対応が必要な関係者

出典:総務省(令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム)

ドメイン名登録者

申請して登録されたドメイン名を維持管理するとともに、そのドメイン名を利用したサービスを提供する組織(または個人)を指します。ドメイン名登録者には、権威 DNS サーバーで公開されるデータを作成する責任があります。つまり、自分のドメイン名を DNSSEC に対応すると決定する責任もドメイン名登録者にあります。

ドメイン名登録事業者

TLD 管理組織(レジストリ)とドメイン名登録者の間に入り、ドメイン名登録に関する窓口業務を行う組織を指します。この組織が提供する窓口業務には、DNSSEC に関係するものが含まれることに注意してください。

権威 DNS サーバー運用者

ドメイン名と関連情報を結びつけるデータベース機能を提供する権威 DNS サーバーを運用する組織を指します。前述したように権威 DNS サーバーの運用形態にはドメイン名登録者自身による運用、ドメイン名登録事業者が提供する権威 DNS サーバーサービス、第三者が提供する権威 DNS サーバーサービスなど、様々なものがありますが、行う作業はいずれも、ドメイン名登録者が作成した DNS データの安定的かつ継続的な公開です。

フルリゾルバー運用者

クライアントからのリクエストに応じて権威 DNS サーバーで公開された情報を検索し、取得・分析して結果をクライアントに返す（名前解決を行う）、フルリゾルバーを運用する組織を指します。

3. 本ガイドラインにおける記載の注意事項

(1)各関係者が読むべき章:ドメイン名登録者への注意

本ガイドラインの各章は、セクション 1 で記述した関係者に向けて書かれています。ほとんどの場合、自分が読むべき章はタイトルから明らかですが、ドメイン名登録者に関しては注意が必要です。

ドメイン名登録者の場合、第 4 章に加え、権威 DNS サーバーを自身で運用していない場合であっても第 3 章を読む必要があります。権威 DNS サーバーで公開されるデータを作成する責任はドメイン名登録者にあることを思い出してください。自身のドメイン名を DNSSEC に対応させる場合、ドメイン名登録者による判断が必要になります。そのため、本ガイドラインではドメイン名登録者が具体的に何をしなければならないのかを理解するため、第 4 章に加え、第 3 章を読むことを推奨します。特に、セクション 3.4.4(鍵の管理に関する注記)は、必ず読むようにしてください。

(2)要求レベルに関する用語

本ガイドラインでは、要求レベルを表現するため、以下の表記法を使用します。

・～しなければならない(MUST)
～してはならない(MUST NOT)
指定された要求項目を実現しない限り、ガイドラインで要求する水準を達成できなくなるという、強い要求レベルを表現するために使用します。

・～すべきである(SHOULD)
本ガイドラインでは RFC 2119 で定義される”SHOULD”ではなく、ISO の表記に近い用語として定義しますのでご注意ください。つまり、指定された要求項目について、「特別な事情がないならそうしなければならない」ではなく「そうすることが望ましい」という、中程度の要求レベルを表現するために使用します。

・～してもよい(MAY)
指定された要求項目について、その項目を実現する方が望ましいが、実現しなくてもガイドラインの水準は満たすことができるという内容を表現するために使用します。例えば「○○という対応を選択してもよい(MAY)」と書かれている場合、○○という対応をした方が望ましいが、対応をしなかったとしてもガイドラインの水準は満たされるという意味になります。

第1章 ドメイン名の重要性とライフサイクルマネージメント

1.1 経営者・代表者の方へ

ドメイン名は、インターネット上でのアイデンティティとして非常に重要であり、組織やサービスのブランド価値やお客様からの信頼と直接結びついています。そのため、ドメイン名の適切な管理と保護は、ご自身の組織のサービスの安全性と信頼性を維持し、インターネット上でのブランド力を持続的に高めるために欠かせない要素となります。そのため、インターネット上でのサービスを提供する際には、Web や電子メールのサーバーのセキュリティ対策と同様に、インターネットでドメイン名を使えるようにするために使われる、DNS の安定運用が非常に重要です。

ドメイン名を乗っ取られたり、不正アクセスを受けたりしないように保護するための対策が強く求められています。DNSSEC は DNS に出自の認証（作成者が設定したデータであること）と完全性の保証（受け取ったデータに改ざんや欠落が見られないこと）を提供する技術であり、DNS データを DNSSEC で保護することで、その信頼性を高めます。

DNSSEC を使用することで、ドメイン名の健全性と完全性を確保し、お客様を情報の改ざんから守ることが可能になります。DNS の内容を保護するためには DNSSEC の導入が必須ですが、その導入には DNS を検索する側と、ドメイン名を DNS に登録する側の双方における、独立した対応が求められます。前者には主に ISP が提供する DNS フルリゾルバーにおける対応が、後者にはドメイン名を登録する組織や、その組織が利用する権威 DNS サーバーを提供する事業者における対応が必要になります。

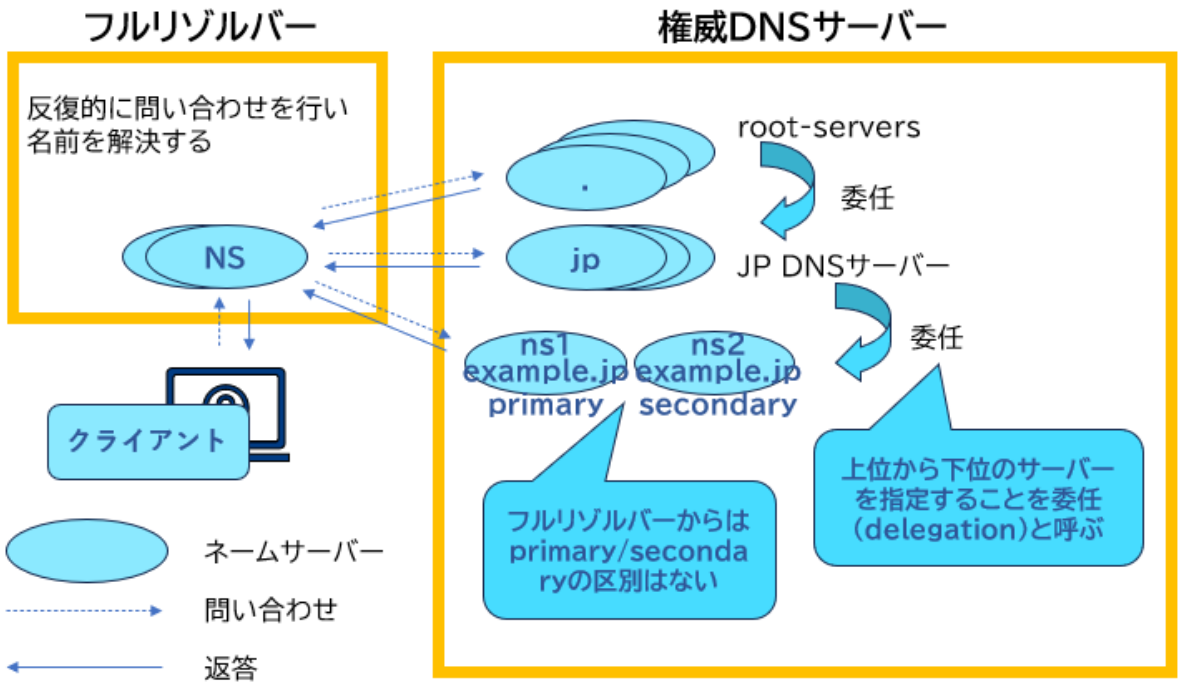

図 2 DNS 構成図

出典：総務省（令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム）

本文書では、第 0 章でフルリゾルバーの DNSSEC 対応、第 3 章で権威 DNS サーバーの DNSSEC 対応、そして第 4 章ではドメイン名登録者およびドメイン名登録事業者を含む、関連事業者の DNSSEC 対応について説明しています。

フルリゾルバーの DNSSEC 対応では従来からのサービスに加え、DNSSEC の署名検証を有効にするための作業が必要となりますが、その負担は大きなものではありません。第 2 章では、フルリゾルバーで DNSSEC の署名検証を有効にすることで信頼の連鎖を構築し、情報が改ざんされていないことを確認することの重要性を強調しています。また、フルリゾルバーで DNSSEC の署名検証を有効にするためには、利用者に提供するすべてのフルリゾルバーを DNSSEC に対応させ、時刻を同期させた上で、IP フラグメンテーションの発生を回避する必要があります。そのため、フルリゾルバーにおいて DNSSEC の署名検証を有効にするにあたっては、DNS サーバーソフトウェアやアプライアンスサーバーなどにおける対応状況を確認した上で、性能を検証・向上させる必要があります。

また、DNSSEC の導入後は、トラストアンカーが最新であるかの確認、適切なログの取得、日々の運用状態の確認が必要になります。第 2 章では日常の監視や確認項目の追加、ログによる異常のチェック、トラブルシューティングの基本や障害対応の方法、運用のノウハウなどについても、詳しく説明しています。

権威 DNS サーバーの DNSSEC 対応では、権威 DNS サーバー運用者とドメイン名登録者における対応の双方が必要になります。権威 DNS サーバー運用者の対応では、DNSSEC 関連情報を適切に設定・公開し、設定された内容を署名する機能の導入が必須です。ドメイン名登録者の対応では、権威 DNS サーバー運用者が提供する署名の機能を用いて自分のドメイン名の情報を署名し、それを権威 DNS サーバーで公開するのと同時に、レジストリの権威 DNS サーバーにドメイン名登録事業者を通じて、DNSSEC に対応するための情報を登録・更新する必要があります。第 3 章では、権威 DNS サーバー運用者が DNSSEC に対応する際の主なポイントとして、DNSSEC で実現される出自の保証とデータの完全性・不在証明の重要性を強調し、これらを実現するために必要な DNSSEC 関連のリソースレコードについて詳しく解説しています。また、時刻の同期、鍵保護の手段の提供など、DNSSEC 対応の要件についても触れており、導入の準備、運用時の監視、トラブルシューティング、運用のノウハウなどに関する具体的なガイドラインを提供しています。特に、署名方法の選定、サーバーの構成、鍵のライフサイクルの管理など、権威 DNS サーバーの運用における、具体的なアドバイスが含まれています。

第 2 章と第 3 章が DNS の運用者向けの内容であるのに対し、第 4 章ではドメイン名登録事業者、およびドメイン名登録者が DNSSEC に対応するために必要な項目について説明しています。2010 年に DNSSEC の正式運用が開始されてから 10 年以上が経過し、国内外において、DNSSEC 対応に関する様々な運用ノウハウが蓄積されています。そうしたノウハウを活用することで、運用開始時に慎重な対応が必要になるものの、DNSSEC 対応が大きな問題を引き起こすことはないことが明らかになっており、また、運用に関わるコストの増加やリソースの追加も大きなものではないことがわかっています。インターネットにおいて自身が管理・運用するドメイン名の安全性を高め、お客様を守るためにも DNSSEC への対応は必須であり、各組織の経営者は、このことに強い意志を持つ必要があります。

1.2 ドメイン名の重要性

ドメイン名は、インターネット上のデバイスやコンピューターを特定するための識別子です。ドメイン名は Web ページの URL やメールアドレスの一部を構成しており、人間にとって覚えやすい名前でコンピューターを指定するために使用されます。ドメイン名は一つのルートから始まる階層構造であり、トップレベルドメイン（TLD）までを含むドメイン名の完全な形式は「FQDN」と呼ばれます。TLD は 3 文字以上の gTLD と 2 文字の ccTLD に大別され、gTLD には多種多様なものを含む一方、ccTLD は国や地域に割り当てられる国別コードで構成されます。ドメイン名のガバナンス（全体管理）には米国の非営利団体である ICANN が関与しており、gTLD や jp などの主な ccTLD ではレジストリ・レジストラモデルに基づく形で、ドメイン名の一元管理と登録者からの申請を処理しています。

ドメイン名は、インターネット上でサービスを提供するための基盤であり、顧客がサービスを利用するための入口となります。そのため、顧客がサービスを安全に利用し続けるためには、顧客が利用するドメイン名を保護し、安全性を高めることが不可欠となります。ドメイン名が危険に晒された場合、サービスの利用を妨害されることになり、顧客の安全な利用が損なわれ、結果として顧客の信頼を失うことになりかねません。

このように、ドメイン名は組織やサービスへの顧客からの入口であり、接点となっています。そのため、ドメイン名は組織やサービスが持つブランド力との強い関係性を有しており、ドメイン名の安全性を高めることで、顧客からの高い信頼の獲得が期待できます。こうしたことから、顧客との接点となっているドメイン名の適切な管理と保護に十分な注意を払い、提供するサービスのレピュテーションを高めることは、インターネット上でサービスを安定的に提供する上での、非常に重要な要因となります。

ドメイン名登録者がインターネット上でサービスを提供するにあたり、CDN 事業者やクラウド事業者にアウトソーシングすることによって、顧客に提供するサービス品質を向上できます。その具体的な項目として、パフォーマンスの向上やセキュリティの強化、DDoS 攻撃をはじめとするサイバー攻撃耐性の向上などが挙げられます。しかし、ドメイン名そのものの保護については、ドメイン名登録者である組織自身が直接関与する必要があります。その内容には、ドメイン名の登録と更新をはじめとする適切なライフサイクルマネージメントの導入やセキュリティ対策、例えば不正アクセスからの保護、ドメイン名の乗っ取りの防止などが挙げられます。

1.3 ドメイン名の保護

顧客がサービスを安全に利用できるようにするためには、顧客が利用するドメイン名を守ることが不可欠です。ドメイン名の安全性を高め、安心して利用できる環境を提供・維持し続けることはサービスの信頼性、ひいては、顧客からの信頼の向上につながります。顧客に関する様々な情報を守り、安全性を担保するためにも、ドメイン名のセキュリティは重要です。言い換えれば、ドメイン名の安全性を向上させ、顧客をオンライン上の脅威から守ることは、事業者の責務の一つであるとも言えます。

インターネット上のサービスを提供しているドメイン名の管理権限を奪う行為（ドメイン名の乗っ取り）は、単なるいたずらや示威行為に留まらず、悪意や害意、あるいは経済的な理由に基づく組織的な犯罪行為に基づくものであるという認識が必要です。ドメイン名の乗っ取りが発生した場合、運用中のサービスに悪影響が及ぶだけではなく、事業継続における大きな脅威ともなり得ます。ドメイン名の乗っ取りにより顧客情報の漏洩が発生する可能性があるほか、乗っ取られたドメイン名を介して、フィッシング詐欺やマルウェアなどの脅威に晒されることになります。結果としてこうした状況は、そのドメイン名を登録していた組織の信頼と評判にも大きなダメージを与えることになります。

つまり、ドメイン名の保護は、ドメイン名登録者の顧客とビジネスの双方をオンライン上の様々な脅威から守るために必要不可欠であり、単に技術的な問題だけではなく、ビジネスの持続可能性と成長に直接関わる、重要な課題です。

1.4 ドメイン名の登録とライフサイクルマネージメント

インターネットにおいてサービスを開始するにあたり、サービス用のドメイン名が必要になる場合があります。その際、そのドメイン名の登録・利用は、サービスを提供する組織のその後のオンラインプレゼンスに大きな影響を及ぼすことになるため、慎重な考慮が必要になります。

ドメイン名はインターネット上でのアイデンティティであり、適切な管理と戦略的な利用が不可欠です。組織が一度登録したドメイン名は知的財産としての価値を持つことになるため、ライフサイクルマネージメントを考慮した、長期的な保護の対象とすべきです。そのため、短期間の使用を目的としたドメイン名の登録・使い捨ては避け、長期的な視点でその価値を高め、効果的な使用ができるようにすることを心がけるべきです。また、ドメイン名の管理にあたってのセキュリティリスクを最小化するため、ドメイン名登録情報に対する適切なロックの実施やドメイン名登録者の認証、そして DNS の安定した運用が必要になります。

ドメイン名は登録されることで登録者に使用権が付与されるものであり、登録者が所有するものではありません。ある時点でのドメイン名登録者は TLD のレジストリやレジストラが公開する WHOIS や RDAP で確認できますが、その登録者はその時点での使用権を有しているに過ぎず、所有しているわけではありません。そのため、登録者が使用権を放棄した場合、そのドメイン名は登録できない状態ではなく、誰でも登録できる状態になります。

ドメイン名の登録は「先願主義」に基づいており、先に申請した者がそのドメイン名を登録できるようになっています。.jp や.com/.net など、主なドメイン名ではその登録プロセスにおいて、申請されたドメイン名の文字列の審査は原則として行われず、仮に既存の社名やブランド、サービス名と同一もしくは類似したものであったとしても、そのドメイン名が未登録であった場合、誰でも登録することができます。

ただし、商標権を侵害する恐れがあり、かつ不正な目的でドメイン名が登録されている場合はドメイン名紛争処理方針（DRP）に従って対応することで、権利者への移転や登録の取り消しが認められることがあります。しかし、ただ単にドメイン名が登録・利用されているというだけでは、その利用をやめさせることはできません。DRP は.com/.net などの gTLD では ICANN が定める UDRP、.jp では JPNIC が定める JP-DRP により、グローバルな基準として運用されています。

ドメイン名は「所有」されるのではなく、「登録」されるものであるということにはドメイン名の登録者からすると違和感を覚えるものではありますが、ドメイン名の登録者はグローバルに定められたこの原則に従って行動する必要があります。そこで重要になってくるのが、ドメイン名は知的財産の一部を構成する資源であるとの前提に基づいた、適切なライフサイクルマネージメントの導入です。具体的には、そもそもドメイン名を新規登録すべきか、新規登録する場合にはどのような文字列を登録すべきか、登録されたドメイン名をどのように利用し、そのための運用を DNS の運用を含め、組織内でいかに行うべきか、定期的に必要になるドメイン名の更新を忘れずに行う仕組みづくり、ドメイン名の登録者情報を最新のものに保ち、仮にそのドメイン名の利用を停止する場合には利用を終了するプロセス（サンセットプロセス）をいかに行うか、などの項目を慎重に検討した上で、ドメイン名の管理を実施することになります。

残念なことに、gTLD の追加や登録料の低価格化などにより、ライフサイクルマネージメントの考慮が十分でない、ドメイン名の安易な登録が増加しています。こうしたドメイン名がポリシーの変更

やコストの削減などを理由として、利用終了後にそのまま廃止された場合、予期せぬトラブルが引き起こされるリスクに晒されることになります。

具体的なリスクとして、再び登録できる状態になる瞬間を狙って第三者がそのドメイン名を素早く再登録する、ドロップキャッチと呼ばれる事例が挙げられます。また、廃止される予定のドメイン名が登録事業者によってドメインパーキングされ、オークションを経て転売に掛けられる事例も発生しています。オークションでは人気のあったドメイン名が競売にかけられ、高額で取り引きされる事例も報告されており、大きな話題になる場合もあります。こうした事例の発生はそのドメイン名を手放した組織にとって、セキュリティ面だけでなく、レピュテーションの面でも大きな影響を及ぼすことになります。

組織におけるドメイン名の選定や管理は慎重に行うべきであり、企業のブランディングやセキュリティポリシー、組織のオンラインでの事業運営方針に沿った、適切なドメイン名ライフサイクルマネージメントの導入が必要になっています。

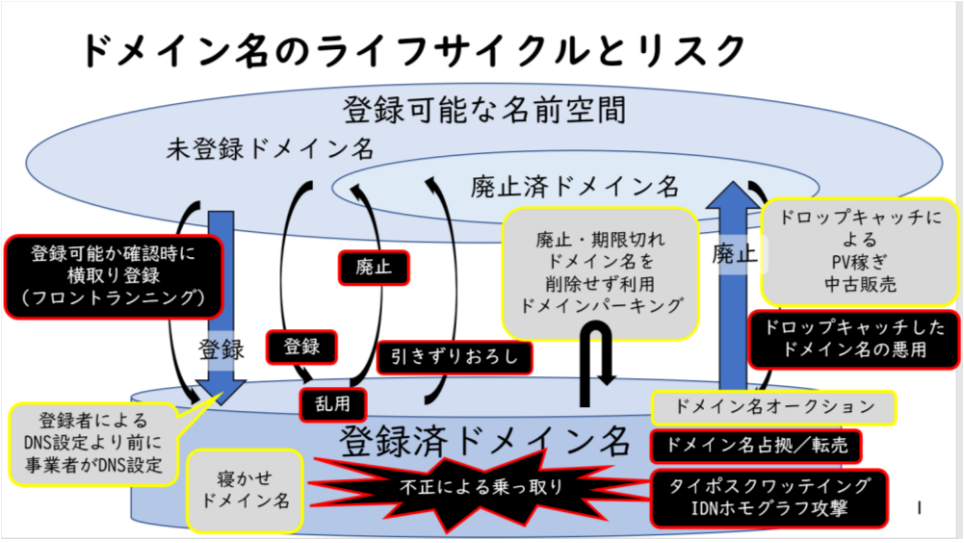

図 3 ドメイン名のライフサイクルとリスク

出典元:一般社団法人日本ネットワークインフォメーションセンター「インターネット 10 分講座 85 号」
https://www.nic.ad.jp/ja/newsletter/No85/NL85_0800.pdf

廃止されたドメイン名を再登録する第三者の主な目的として、以前のウェブサイトが有していたページビューや SEO の価値、そして既存のリンクの獲得が挙げられます。そのため、一般ユーザーからのアクセスが多かった人気のあるドメイン名は、市場で高く評価されることになります。このようなドメイン名は中古市場で高く取引されることになり、中古ドメイン名の再利用を手掛ける専門の大手事業者も存在しています。廃止するドメイン名の再登録に関する決定的な対策は存在せず、ドメイン名を長期にわたって登録し続けるサービスの提供を一部のドメイン名事業者が行っているほか、ドメイン名の失効にあたり利用価値を漸減させるためのサンセットの手順の確立などが検討されている状況にあります。

ドメイン名の登録管理におけるベストプラクティスの一つとして、サービスの開始にあたり、ドメイン名の新規登録が実際に必要かどうかを慎重に検討することが挙げられます。新規登録する場合、

登録されるドメイン名の価値を最大化し、効果的に使用されるようにすべきです。また、年次のイベントに対応する場合、「example2020.jp」のような年毎の新規登録に替え、「2020.example.jp」のように、登録済みのドメイン名のサブドメインを使用することで、前述したドロップキャッチやドメイン名オークションのリスクを回避すべきです。

ドメイン名の登録においては、登録する TLD の評判や、レジストリ・レジストラ・リセラといった、ドメイン名の登録に携わる事業者の信頼性も重要な要素となります。セキュリティ面では、ドメイン名登録情報のロックサービスの提供の有無や登録者の認証プロセス、登録管理業務のインターフェースや TLD の DNS の運用の品質についても、考慮に入れるべきです。

1.5 ドメイン名を守るための DNSSEC

ドメイン名を守ることは、ドメイン名の登録情報を守り不正に書き換えられないようにする（登録情報の保護）、登録情報を最新の正しい情報に更新して維持し続ける（健全性の維持）、受け取り側において情報が正しいものであるかを検証できる仕組みを用意する（完全性の担保）、の 3 つに分類できます。

登録情報の保護では、正当な管理権限を持たない第三者によってドメイン名の登録情報を書き換えられないように注意を払う必要があります。また、ドメイン名の健全性の維持では、ドメイン名と DNS の健全な管理運用が鍵を握ります。サーバー上で動いている DNS ソフトウェアの健全性は言うまでもなく、TLD（親ゾーン）からの委任が正しく行われているか、複数の権威 DNS サーバー間での情報の同期が正しく機能し、かつ保持している情報が一貫性を有しているか、保持しているドメイン名に関する情報に誤りがないかなど、注意を払うべき点が多岐にわたっていることに注意を払う必要があります。

すなわち、DNSSEC は自組織のドメイン名を守るだけでなく、そのドメイン名を利用しようとしている顧客を守るための仕組みでもあることを意識すべきです。悪意を持った第三者が通信の途中で DNS の情報を書き換えた場合、その改ざんを検知して攻撃者の誘導先にアクセスできないようにすることが DNSSEC の存在意義であり、可用性を犠牲にしてでも完全性をまっとうするものとなっています。自らがインターネット上で公開しているサービスが価値あるものであり、かつ、それを顧客に提供している場合、サービス提供者はその顧客を守ることにも十分注意を払うべきであり、その仕組みの一つが DNSSEC であるということを理解する必要があります。

第 2 章 フルリゾルバーの DNSSEC 対応

2.1 DNSSEC 対応の基礎

本章の主要な想定読者は「フルリゾルバー運用者」です。したがって、読者が DNSSEC の動作と DNS および DNSSEC に関連する用語を理解していることを前提に書かれています。DNSSEC の基本動作については[2-1]を、関連する用語については[2-2]を参照してください。本セクションでは、本章の記述に関係するフルリゾルバーの動作の概要を説明します。

フルリゾルバーは、エンドユーザーが使用する端末(クライアント)から DNS 問い合わせを受け取り、クライアントに代わって名前解決を行い、その結果をクライアントに返すという役割を持ちます。フルリゾルバーが DNSSEC に対応すると、DNSSEC 署名され、親ゾーンに信頼の連鎖のための情報(DS リソースレコード)が登録されているドメイン名の名前解決手順に、DNSSEC 関連の作業が追加されます。

具体的には、問い合わせを受理すると、フルリゾルバーはルートゾーンから名前解決対象のドメイン名に至るまでの、信頼の連鎖の構築を試みます。信頼の連鎖の構築に成功し、問い合わされた名前に関して得られた情報が信頼できる場合にのみ、クライアントに返される応答に AD(Authentic Data)ビットが設定されます。信頼の連鎖に関する詳細な説明については、セクション 3.1.1 を参照してください。信頼の連鎖を構築でき、応答に追加された署名を検証できた場合、フルリゾルバーが取得した応答は権威 DNS サーバーとフルリゾルバーの間で改ざんされていなかったということを検証できます。つまり AD ビットは DNSSEC 検証の成功、つまり、受け取った情報が真正な権威サーバーから得た情報で、かつ通信途中で改ざんされていないことをクライアントに通知するための印となります。

応答の DNSSEC 検証に失敗した場合、原因の可能性には設定誤りなど様々なものがあり得るにせよ、フルリゾルバーの得た応答に含まれるリソースレコードが悪意ある第三者に改ざんされた可能性を疑う必要があります。その場合、フルリゾルバーはクライアントに SERVFAIL 応答を返し、疑わしい応答をクライアントが利用することを防ぎます。

信頼の連鎖の起点はルートゾーンの KSK に関する情報です。その初期値はフルリゾルバーの設定として与えられるもので、トラストアンカーと呼ばれます。ルートゾーンの KSK は定期的に更新(ロールオーバー)されるため、最新の情報を使用する必要があります。トラストアンカーを自動で更新する仕組みが RFC 5011 で規定されています。

信頼の連鎖の構築は、トラストアンカーを使用してルートゾーンの KSK(DNSKEY リソースレコード)を認証することから開始されます。その後、ルートゾーンの KSK を使用してルートゾーンの ZSK を認証し、ルートゾーンの ZSK を使用して TLD(例えば.jp)の委任の DS を認証するという手順で、ルートゾーンから TLD ゾーンへの信頼の連鎖が構築されます。以下同様にして TLD から SLD(Second Level Domain)への信頼の連鎖が構築され、最終的に目的のドメイン名に到達するまで同じ作業がくり返されます。信頼の連鎖の構築には、DS、DNSKEY、RRSIG などの DNSSEC 関連のリソースレコードが必要になるため、フルリゾルバーが信頼の連鎖の構築を試み

る際には、必要に応じて関連するリソースレコードを取得する作業が発生します。

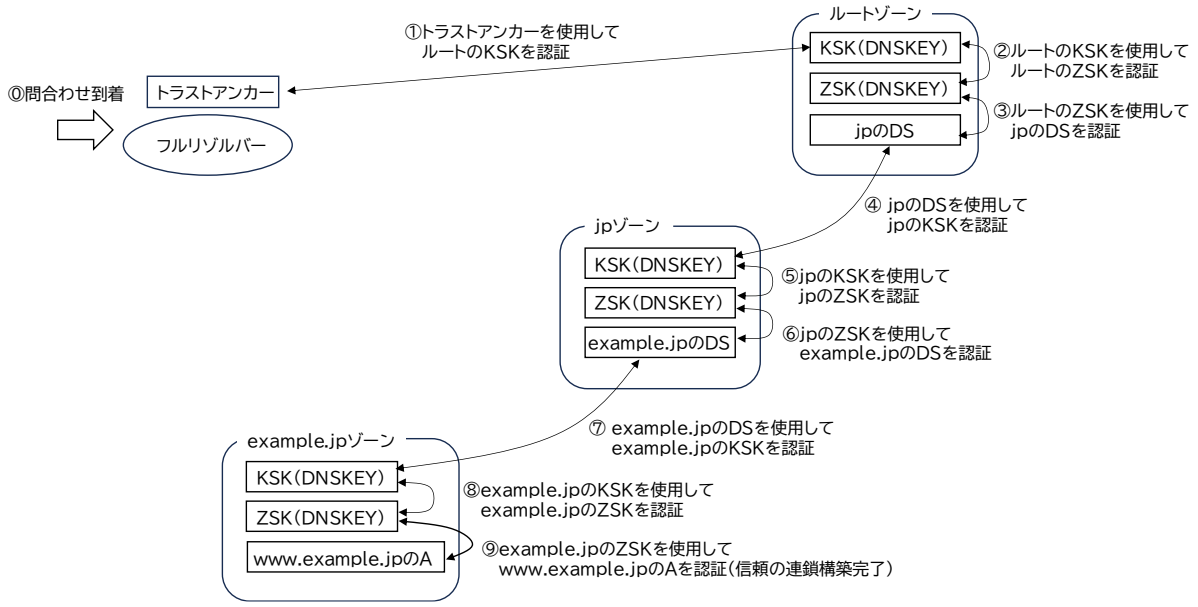

図 4 は、フルリゾルバーが www.example.jp のアドレス(A リソースレコード)を解決する際に、信頼の連鎖がどのように構築され、DNSSEC 検証されるのかを示しています。

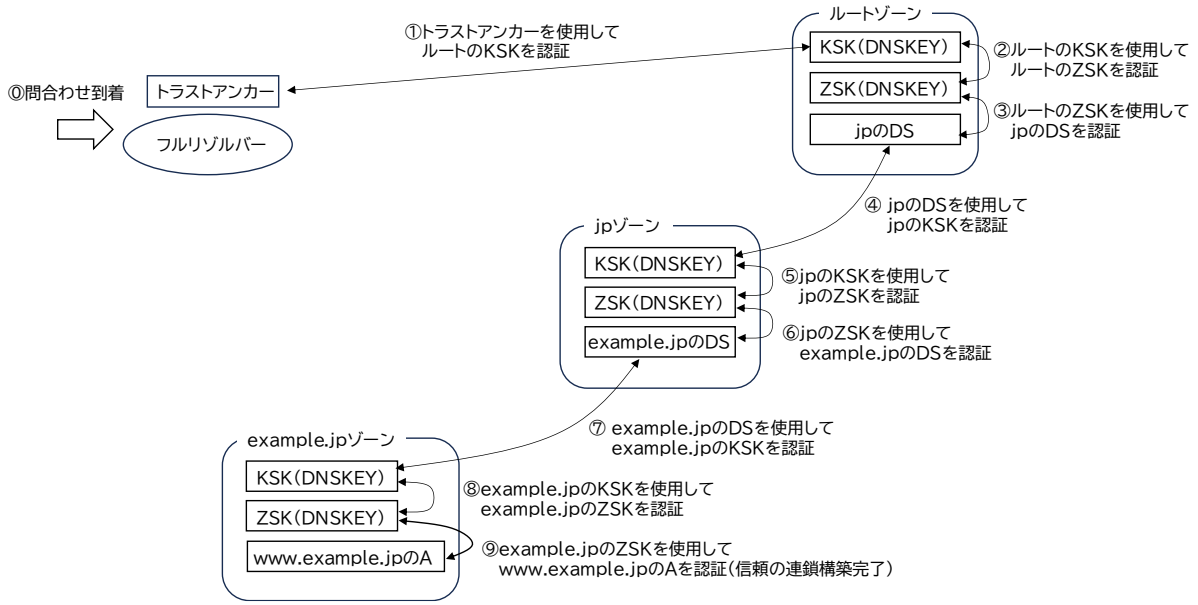

図 4　フルリゾルバーによる信頼の連鎖の構築手順(www.example.jp の A リソースレコードの場合)

出典:総務省(令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム)

ここで、「認証」という作業は、リソースレコードに付属する署名を検証し、リソースレコードが正当な所有者(秘密鍵の保有者)のものであること、リソースレコードに含まれるデータが改ざんされていないことを確認する作業を指します。具体的には、①リソースレコードからメッセージダイジェストを算出し、②署名(RRSIG リソースレコード)を公開鍵(DNSKEY リソースレコード)で検証してメッセージダイジェストが改ざんされていないことを検証し、③①と②を検証するという 3 段階の作業が行われます。署名検証には公開鍵暗号の署名アルゴリズムが使用され、メッセージダイジェスト生成

にはハッシュアルゴリズムが使用されます。図 5 は、

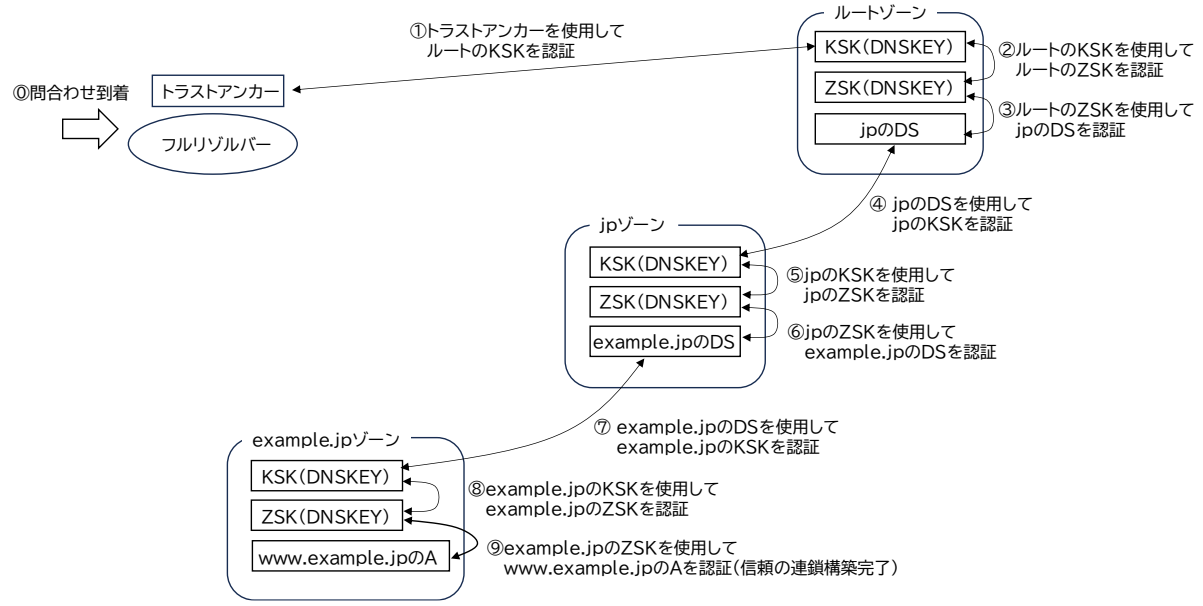

図 4 の⑨の段階、つまり example.jp の ZSK を使用して www.example.jp の A リソースレコードの認証をする例を示しています。

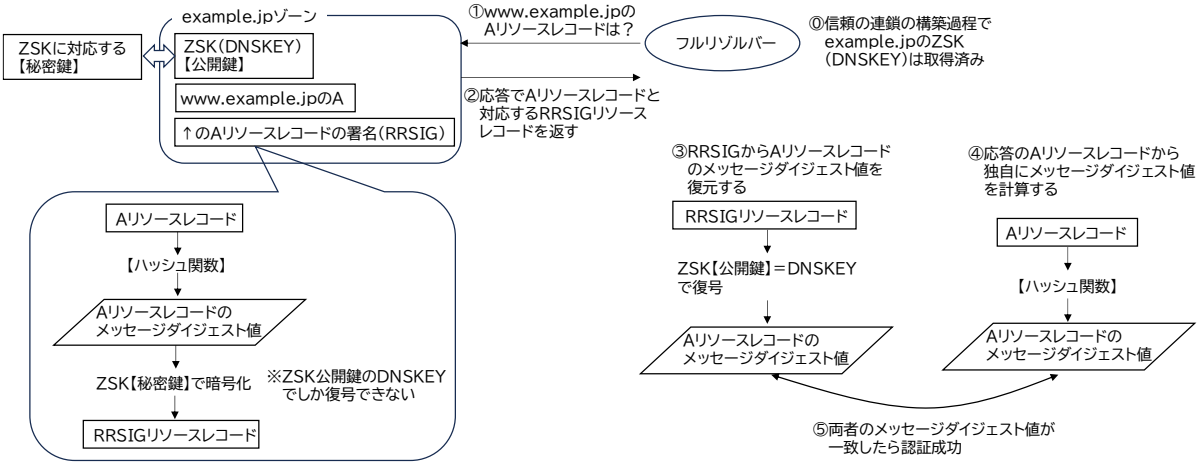

図 5 フルリゾルバーが www.example.jp の A リソースレコードを認証する例(RSA の場合)
出典:総務省(令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム)

署名アルゴリズムとハッシュアルゴリズムはペアで使用されるため、RSASHA(署名アルゴリズムは RSA、ハッシュ方式は SHA-1)や ECDSAP256SHA256(署名アルゴリズムは ECDSA P256、ハッシュ方式は SHA-256)のようなニーモニックが定義されています。フルリゾルバーソフトウェアやアプライアンスによっては全ての署名アルゴリズム/ハッシュアルゴリズムをサポートしていない可能性があることに注意してください。当然ですが、署名アルゴリズムやハッシュアルゴリズムに対応していなければ署名の検証はできません。その場合は RFC 4035 の規定に従って、フルリゾルバーは、未知の署名アルゴリズムやハッシュアルゴリズムを使用するゾーンは未署名(DNSSEC 未対応)であったものとして扱います。

フルリゾルバーはリクエストの内容に応じて権威 DNS サーバーに問い合わせを行いますが、問い合わせ先の権威 DNS サーバーが常に正しく設定されているとは限りません。例えば従来の DNS で Lame delegation のような設定誤りが原因で名前解決に失敗することがあったように、DNSSEC でも設定誤りによる名前解決の失敗は起こり得えます。権威 DNS サーバー側の DNSSEC の設定が誤っている場合、大抵は信頼の連鎖構築に失敗するため、クライアントには応答コード（RCODE）に SERVFAIL が設定されたエラー応答が返されることになります。その場合ユーザーには、ある Web ページにアクセスできなかったりメールの送信に失敗したりするという形で影響が現れます。設定誤りのあるゾーンが影響の大きいものである場合、混乱が生じる可能性があります。そのような場合、一つの方法として、設定誤りのあるゾーンのみ DNSSEC 検証を一時的に無効にして、DNSSEC による保護をあきらめる代わりに利用可能な状態に戻すことが考えられます。RFC 7646 は、その手段としてネガティブトラストアンカーを規定しています。

2.2 DNSSEC 対応の要件

フルリゾルバーで DNSSEC 検証を有効にするためには、満たすべき要件が存在します。それらの項目を以下に提示します。

(1)全てのフルリゾルバーにおける DNSSEC 対応

2-1.	公開しているフルリゾルバーは全て DNSSEC 対応にしなければいけません (MUST)
2-2.	例えば 3 つの異なる IP アドレスで 3 台のフルリゾルバーを運用している場合に、3 台中 2 台を DNSSEC 対応にして 1 台は未対応のまま残すという運用をしてはいけません (MUST NOT)

公開しているフルリゾルバーは全て DNSSEC 対応にしなければいけません(MUST)。例えば 3 つの異なる IP アドレスで 3 台のフルリゾルバーを運用している場合に、3 台中 2 台を DNSSEC 対応にして 1 台は未対応のまま残すという運用をしてはいけません(MUST NOT)。DNSSEC のセキュリティ機能が正しく機能して署名検証に失敗した場合(例えばキャッシュ汚染攻撃で挿入された偽のリソースレコードを防止した場合)、クライアントには SERVFAIL が返されますが、クライアントは利用可能な別のフルリゾルバーでリトライします。リトライしたフルリゾルバーが DNSSEC に対応していないと、攻撃による偽造リソースレコードによりクライアントが偽サイトに誘導されるリスクが生じます。

(2)時刻の同期

2-3.	DNSSEC の署名情報を保持する RRSIG リソースレコードには、署名の有効期間に関する絶対時刻の情報が含まれます。フルリゾルバーに設定される時刻が合っていない場合、本来有効であるはずの署名を無効と扱う恐れがありますから、NTP 等の手段を使用してフルリゾルバーの時刻を信頼できる時刻ソースと同期しておかなければいけません (MUST)

DNSSEC の署名情報を保持する RRSIG リソースレコードには、署名の有効期間に関する絶対時刻の情報が含まれます。フルリゾルバーに設定される時刻が合っていない場合、本来有効であるはずの署名を無効と扱う恐れがありますから、NTP 等の手段を使用してフルリゾルバーの時刻を信頼できる時刻ソースと同期しておかなければいけません(MUST)。

(3)IP フラグメンテーションの回避

2-4.	DNSSEC を有効にすると、問い合わせに対する応答パケットのサイズが大きくなります。今日では、UDP で IP フラグメンテーションを発生させることは推奨されていない

	ので、EDNS のバッファーサイズを一般的な MTU 値を超えた値に設定してはいけません （MUST NOT）
2-5.	EDNS のバッファーサイズは、リゾルバーソフトウェアのデフォルト値を使用すべき （SHOULD）

DNSSEC を有効にすると、問い合わせに対する応答パケットのサイズが大きくなります。今日では、UDP でフラグメンテーションを発生させるのは推奨されていないので、EDNS のバッファーサイズを一般的な MTU 値を超えた値に設定してはいけません（MUST NOT）。基本的には、リゾルバーソフトウェアのデフォルト値を使用すべき（SHOULD）です。

（4）拡張 DNS エラー導入の検討

2-6.	RFC 8914 で定義される拡張 DNS エラー（EDE: Extended DNS Errors）の導入を検討してもよい （MAY）

DNS の名前解決失敗、DNSSEC の署名検証失敗の原因を詳細に伝える仕組みとして、RFC 8914 で拡張 DNS エラー（EDE: Extended DNS Errors）が規定されています。本ガイドライン執筆時点では主要なフルリゾルバーソフトウェアは対応済みですが、まだ実装から十分な時間を経ていないので、ある程度機能の安定動作が確認されてから導入を検討してもよい（MAY）でしょう。

2.3 導入準備

DNSSEC 検証を有効にする前に、以下の点を確認する必要があります。

(1)ソフトウェア／アプライアンスの対応状況の確認／更新

現在使用中のリゾルバーソフトウェアまたはアプライアンスが DNSSEC 検証に対応しているかどうかを確認する必要があります。ここで「DNSSEC 検証に対応」とは、付録 2 で示される RFC に準拠していることを指します。

対応していない場合には、ソフトウェアまたはアプライアンスの OS 更新などを行って最新の状態にし、可能な限り準拠した状態を保つようにしてください。

また、リゾルバーソフトウェアをパッケージ単位で導入する場合、パッケージのバージョンによって DNSSEC に関連する機能のデフォルト値が異なる可能性がありますので、注意してください。

(2)性能の確認／増強

DNSSEC に対応すると、DNSSEC 関連のリソースレコードを追加で取得したり信頼の連鎖を構築するために署名を検証したりするなどの作業が発生しますから、CPU やメモリの負荷が増大し、スループットが低下することが予想されます。どの程度の影響があるかは、問い合わせに占める署名されたドメイン名の割合やキャッシュのヒット率など、それぞれの使用環境に依存するため、一概に述べることはできません。

参考情報として、DNS 実装 BIND の DNSSEC Guide の中にハードウェア要件に関する記述があります[2-3]。その内容を抜粋すると「CPU 利用率は通常 5%のオーダーである。フルリゾルバー起動後の短い時間は 20%程度に増加するが、キャッシュが満たされると速やかに減少する。一般的な ISP トラフィックと 2022 年半ばのインターネットの状況を考慮すると、キャッシュに使用されるメモリ消費量は約 20%増加する。DNSSEC を有効にしたことによるトラフィック増加量は大抵の場合、測定誤差の範囲内である」となります。

一般的には、現代的なハードウェアを使用するサーバー構成であれば DNSSEC 検証に対応させることの負担は気にしなくてもよいと考えられます。

2.4 導入

導入の際には以下を確認してください。

(1)最新のトラストアンカーの確認／導入

2-7.	リゾルバーに設定として与えられるトラストアンカーには、最新のものを使用しなければいけません (MUST)
2-8.	トラストアンカーの更新は手動による作業でも行えますが、特別な理由がなければパッケージに付属しているトラストアンカーを使用すべき (SHOULD)

フルリゾルバーに設定するトラストアンカーには、最新のものを使用しなければいけません(MUST)。トラストアンカーの更新は手動による作業でも行えますが、特別な理由がなければパッケージに付属しているトラストアンカーを使用すべき(SHOULD)です。パッケージ付属のものを使用する場合には、ソフトウェアのアップデートを怠らないように注意してください。

(2)ロギングの設定変更

2-9.	DNSSEC 署名検証エラーがログ出力できるソフトウェア・アプライアンスの場合は、出力するように設定すべき (SHOULD)
2-10.	応答の RCODE ごとの問い合わせ数を取得して統計的に出力できるように設定すべき (SHOULD)
2-11.	署名検証の成功数、失敗数などのメトリクスが取得できる場合には、それらを取得すべき (SHOULD)

DNSSEC 署名検証エラーがログ出力できるソフトウェア・アプライアンスの場合は、出力するように設定すべき(SHOULD)です。これは、DNS クエリログ(フルリゾルバーが受信した全ての DNS 問い合わせに関する時刻、問い合わせ名、結果などの情報)ではないので注意してください。

また、応答の RCODE ごとの問い合わせ数を取得して統計的に出力できるように設定すべきです(SHOULD)。あわせて、署名検証の成功数、失敗数などのメトリクスが取得できる場合には、それらを取得すべき(SHOULD)です。

(3)稼働状況の確認

キャッシュをクリアした状態で、DNSSEC 署名されたドメイン名が名前解決できることを確認してください。DNSSEC 署名されたドメイン名を自分で維持している場合、それを使用するのが最善です。そのようなドメイン名を準備できない場合は、外部の DNSSEC 署名されたドメイン名を利用してください。

2.5 運用

2.5.1 日常的な監視・確認項目の追加

2-12. 通常のフルリゾルバーの監視項目に、時刻同期状況の確認、DNSSEC 署名された名前が解決できることの確認を加えなければいけません （MUST）

通常のフルリゾルバーの監視項目に、時刻同期状況の確認、DNSSEC 署名された名前が解決できることの確認を加えなければいけません(MUST)。

2.5.2 ログによる異常の有無の確認

2-13. 統計情報を出力し、RCODE として SERVFAIL が返される数または割合の変化を調べる仕組みを構築すべき （SHOULD）

統計情報を出力し、RCODE として SERVFAIL が返される数または割合の変化を調べる仕組みを構築すべき(SHOULD)です。これらが急増している場合、DNSSEC に起因する何らかのトラブルが発生している可能性があります。

2.6 トラブルシューティング

2.6.1 トラブルシューティングの原則

トラブルシューティングを行う際に最も大切なことは、場当たり的かつ属人的な対応を行うのではなく、対応手順を策定し改訂をくり返すことで、体系だった対応が行える体制を維持することです。

これまでの運用経験から、DNSSEC 関連の障害原因はフルリゾルバー側の設定不備か権威 DNS サーバー側の設定不備である場合が多く、キャッシュ汚染攻撃が実際に行われていることによる事例は少ないことがわかっています。したがって、トラブル発生時には、その原因がフルリゾルバー側にあるか権威 DNS サーバー側にあるかを切り分け、フルリゾルバー側にある場合には速やかに対応し、権威 DNS サーバー側にある場合には影響度合いに応じた対応をとることが原則となります。

2.6.2 障害の検出

障害の検出経路は主に 2 つが考えられます。1 つは、セクション 2.5.1 に記述した日常的な監視の結果障害が見つかる場合で、もう 1 つは利用者または顧客からの申告によって障害が発覚する場合です。後者の場合、セクション 2.6.3 以降に記述する技術的な対応に加えて、利用者または顧客への回答が必要となるため、関連部署との連携体制を整えておく必要があります。

2.6.3 障害原因の切り分け

まず、障害が発生している名前に対して CD（Checking Disabled）ビットを設定して名前解決を試みます。名前解決ができた場合には、障害が DNSSEC に起因するものであると切り分けられます。CD ビットを設定した問い合わせを行うツールには、BIND 9 に付属する dig や Unbound に付属する drill などがあります。

次に、信頼の連鎖が正しく構築できるかを確認します。この作業は DNSViz[2-4]などの外部サービスを使用する方が便利でしょう。信頼の連鎖が構築できなければ、原因が権威 DNS サーバー側にあることが判明します。信頼の連鎖が正しく構築できた場合には、障害の原因がフルリゾルバー側である可能性が高いため、ログの解析により原因を調査し対応する必要があります。

2.6.4 障害への対応方法

ここでは、障害原因が権威 DNS サーバー側にある場合を扱います。ここでもまた最も大切なことは、2.6.1 に記述した対応手順の中に①利用者への通知方法、②何らかの対応を取るかどうか、取る場合はその基準は何か、どこまでの対応を取るかなどを明記しておき、手順に従った対応を取ることです。

②について補足すると、対応を取るかどうかの基準として利用者が多いかどうか、TLD など影響範囲が広いかどうか、利用者からの問い合わせ数が一定基準を超えたかどうかなどが考えられます。対応の範囲としては、ネガティブトラストアンカーを使用して一時的にそのドメイン名の DNSSEC 検証を無効化する、DNSViz の結果を証跡として添付し、先方の技術連絡窓口などに対応を促すなどが挙げられます。

2.7 運用ノウハウ

2-14. フルリゾルバーの DNSSEC 対応として以下のレベルを定義し、初期段階における最低限の対応としてレベル 1 への対応、その後の対応としてレベル 2、レベル 3 への対応を推奨すべき （SHOULD）

フルリゾルバーの DNSSEC 対応として以下のレベルを定義し、初期段階における最低限の対応としてレベル 1 への対応、その後の対応としてレベル 2、レベル 3 への対応を推奨すべき（SHOULD）です。本ガイドラインでは以下のような 3 段階を提唱します。

レベル 1: MUST が指定された要件をすべて満たす。
レベル 2: 利用者に影響する署名検証失敗が発生した場合に検知でき、利用者や関係者にその原因を説明できる。
レベル 3: 署名検証失敗が発生した場合に復旧に向けた適切な対応が取れる。

2.7.1 レベル 1

本ガイドラインで MUST が指定された要件を全て満たしている状態です。このレベルにおける最も重要なポイントは、トラストアンカーが適切に管理されている状態を維持することです。

トラストアンカーの更新が必要な状況が発生した場合、対応する KSK のロールオーバーのタイミングで設定変更をすること、自動設定を行っている場合は自動化された仕組みが適切に動作しているかを確認する必要があります。

2.7.2 レベル 2

このレベルの目標は、フルリゾルバーの利用者に影響する署名検証失敗が発生した場合に検知でき、利用者や関係者にその原因を説明できるようになることです。

目標の達成には、統計情報の取得が必要となります。具体的には 2.4（2）および 2.5.2 で SHOULD が指定された要件を満たしてください。以下のサブセクションで、具体的な運用ノウハウも交えて説明していきます。

2.7.2.1 異常の検知

異常の検知は、応答の RCODE ごとの問い合わせ数をはじめとした、フルリゾルバーに関するさまざまな統計情報を検査することで行われます。こうした統計情報の用途は、DNSSEC の異常検知のみにとどまりません。サーバーのリプレースやソフトウェア更新時などの際にも応答の RCODE

ごとの問い合わせ数や割合に変化がないかを調べることで、フルリゾルバーが正常に動作しているかを確認できます。

具体的な統計情報の取得方法は、ソフトウェア実装によって異なります。パケットキャプチャーを介して統計情報を取得する dsc[2-5]などの専用ツールも存在しますが、現在では Prometheus[2-6]で取得するのが一般的です。ただし、dsc は DNS に関連する有用な情報を簡単に得られるという特徴があるので、dsc と Prometheus の双方を設定するのがよいでしょう。

Prometheus を利用する場合は、unbound_exporter[2-7]や bind_exporter[2-8]などの OSS 実装が利用できます。これらは応答の RCODE ごとの問い合わせ数やカウンターも備えているので、可視化や監視にも使いやすいです。

2-15.	本ガイドラインが定めるレベル 1 での運用期間中に統計情報の取得を開始し、ある程度の期間問い合わせの傾向を見てからしきい値を決めるべきです （SHOULD）

応答の RCODE ごとの問い合わせ数が取得できるようになると、SERVFAIL がしきい値を越えていれば何らかの異常を疑ってアラートを発生させることができるようになります。ここではしきい値の設定が重要になりますが、フルリゾルバーの運用ポリシーや利用者の属性などによって異なるため、どの値が適切であるとは一概に言えません。例えば、個人を対象とする場合と法人を対象とする場合では問い合わせの傾向が全く異なりますし、それに応じて想定される SERVFAIL 率や検証失敗数も変わってきます。したがって、レベル 1 での運用期間中に統計情報の取得を開始し、ある程度の期間問い合わせの傾向を見てからしきい値を決めるべきです（SHOULD）。

2.7.2.2 原因の究明

2-16.	原因の究明は、一般にその情報展開（利用者、利用者に説明するサポート部門など）とあわせて考える必要があります。フルリゾルバーが扱うドメイン名の数は膨大なので、全てのドメイン名で生ずる障害情報を共有する必要はありません。その影響度に応じて情報展開をすべきです （SHOULD）

原因の究明は、一般にその情報展開（利用者、利用者に説明するサポート部門など）とあわせて考える必要があります。フルリゾルバーが扱うドメイン名の数は膨大なので、全てのドメイン名で発生する障害情報を共有する必要はありません。その影響度に応じて情報展開をすべきです（SHOULD）。影響度を決める指標としては、問い合わせ総数に対する SERVFAIL 応答数の割合や、サポートセンターへの架電状況の変化などが考えられます。

原因の切り分けはセクション 2.6.3 に従って進めていきます。追加の手段として、RFC 8914 で標準化された拡張 DNS エラー（EDE）対応のフルリゾルバーやパブリックリゾルバーを利用することも選択できるでしょう。この場合、EDE コードが 5～12 の場合には DNSSEC 関連の問題であると切り分けられます。

信頼の連鎖の状況を調査する方法はセクション 2.6.3 で紹介した DNSViz のような外部サービスで構いませんが、注意すべき点が 2 つあります。

2-17.	IP エニーキャストや負荷分散技術を使用して IP アドレスを共有する物理サーバーが複数ある場合に、それを区別する技術として RFC 5001 で NSID(ネームサーバー識別子)が定義されています。自分の AS から調査を行う際には、NSID を表示させるオプションを指定して(例えば dig/delv などを)実行し、応答したインスタンス名を把握しておくべきです (SHOULD)

第 1 に、権威 DNS サーバーは IP エニーキャストを使用して、単一の IP アドレスを複数の拠点で運用している場合があります。そのため、障害内容や攻撃内容によっては、外部のサービスの結果と自分の ISP(AS)から見える状況が一致しない場合があり得ます。IP エニーキャストや負荷分散技術を使用して IP アドレスを共有する物理サーバーが複数ある場合に、それを区別する技術として RFC 5001 で NSID(ネームサーバー識別子)が定義されています。自分の AS から調査を行う際には、NSID を表示させるオプションを指定して(例えば dig/delv などを)実行し、応答したインスタンス名を把握しておくべきです(SHOULD)。

2-18.	外部サービスそのものに障害が発生する可能性もあるので、自分の AS からも信頼の連鎖を調査できるようにしておくべき (SHOULD)

第 2 に、外部サービスそのものに障害が発生する可能性もあるので、自分の AS からも信頼の連鎖を調査できるようにしておくべき(SHOULD)です。使用できるツールとしては、BIND 9 に付属する delv や、DNSViz の CLI 版[2-9]などが挙げられます(DNSViz の CLI 版は導入手順が少し複雑なので、その場合は Docker 版[2-10]を検討するとよいでしょう)。

2.7.3 レベル 3

このレベルの目標は、署名検証に失敗した場合にそれを検知して関係者に説明する(レベル 2)とともに、正しく署名検証できる状態に向けて必要な対応が取れるようになることです。

2.7.3.1 基本的な考え方

フルリゾルバーが扱うドメイン名の数は膨大なので、影響度が少ないと判断できる場合には、復旧に向けた積極的な対応を特段取る必要はありません。復旧を早めたい場合には、権威 DNS サーバーの復旧状況を監視し、復旧後にキャッシュクリアを実行すればよいでしょう。

2-19.	フルリゾルバーが扱うドメイン名の影響度が大きい場合や、積極的な対応を取ってでも復旧をより早めたい場合は、障害が発生しているドメイン名の連絡先(jp ドメインの場合は Whois 検索で表示される「公開連絡窓口」か「技術連絡担当者」)にコンタクトをしてもよい

（MAY）

影響度が大きい場合や、積極的な対応を取ってでも復旧をより早めたい場合は、障害が発生しているドメイン名の連絡先（jp ドメインの場合は Whois 検索で表示される「公開連絡窓口」か「技術連絡担当者」）にコンタクトをしてもよい（MAY）です。コンタクトの際には、DNSViz の出力やコマンドラインツールの出力などの証跡を添付の上で修正を依頼してください。

2-20. 影響度の大きいドメイン名で署名検証の問題が発生している場合、他の ISP でもその異常が観測され、原因の切り分けや修正依頼が既に行われている可能性もあります。日本国内は dnsops.jp[2-11]、海外の情報は dns-oarc[2-12]で交換されているので、常時アンテナを張っておくべき
（SHOULD）

いずれの場合も、コミュニティへの情報提供やコミュニティからの情報収集を行うことで、判断材料を増やすことができます。例えば、影響度の大きいドメイン名で署名検証の問題が発生している場合、他の ISP でもその異常が観測され、原因の切り分けや修正依頼が既に行われている可能性もあります。日本国内は dnsops.jp[2-11]、海外の情報は dns-oarc[2-12]で交換されているので、常時アンテナを張っておくべきです（SHOULD）。

2-21. 署名検証失敗の原因が攻撃ではないと判明し、復旧にある程度時間がかかることが見込まれる場合には、ネガティブトラストアンカーの設定を検討してもよい
（MAY）

上記の作業の結果、署名検証失敗の原因が攻撃ではないと判明し、復旧にある程度時間がかかることが見込まれる場合には、ネガティブトラストアンカーの設定を検討してもよい（MAY）です。次のセクションを参照してください。

2.7.3.2 ネガティブトラストアンカー

ネガティブトラストアンカー（NTA）を設定すると、設定されたドメイン名とそのサブドメインの DNSSEC 署名検証が無効にされますので、利用には細心の注意が必要となります。

署名検証に問題が発生して SERVFAIL 応答が返され続けるのは望ましい状態ではありませんが、署名検証失敗の原因が明らかになる前に独断で NTA を設定するのは危険な行為です。署名検証失敗の理由が攻撃によるものだった場合、署名検証が無効にされた結果、利用者が偽の応答に誘導されて被害が発生するリスクが生じます。

NTA を設定してもよいと考えられるのは、署名検証失敗の原因が攻撃ではないと判断できるようになった後だけです。ただし、フルリゾルバー運用者の視点だけで、攻撃が発生しているか権威 DNS サーバーのリソースレコードに不具合が生じているかを切り分けるのはほぼ不可能で、これを行えるのはドメイン名登録者か権威 DNS サーバー運用者だけです。NTA を設定できるのは、ドメイン登録者が必要なリソースレコードの情報を適切に提供している（間違った情報を提供していない）ことが確認されており、障害の原因が権威 DNS サーバー側にあると切り分けられている場合だけとなります。

権威 DNS サーバー側に問題があることが明らかになっていても、短期間の復旧が期待できる状況であれば NTA は必要ありません。（セクション 2.7.3.1 の記述どおり復旧を監視し、復旧後はキャッシュをクリアすると良いでしょう）。一方で、当該ゾーンの秘密鍵が喪失するなどの事故が発生していた場合には、復旧までに要する時間が長くなる可能性が高くなります。対象のドメイン名が TLD 等の影響が大きいものである場合には、一時的に NTA を設定することになるかもしれません。

2-22. NTA の設定は運用への影響が大きいため、NTA を設定する基準または条件、誰が承認するのか、設定方法および解除方法などは事前に取り決めて手順化しておくべきです（SHOULD）

NTA の設定は運用への影響が大きいため、NTA を設定する基準または条件、誰が承認するのか、設定方法および解除方法などは事前に取り決めて手順化しておくべきです（SHOULD）。

2.8 参考文献

[2-1] DNSSEC の基礎概要
https://www.nic.ad.jp/ja/materials/iw/2012/proceedings/t9/t9-Funato.pdf

[2-2] RFC 8499「DNS の用語」
https://jprs.jp/tech/material/rfc/RFC8499-ja.txt

[2-3] BIND DNSSEC Guide: Getting Started セクションの Hardware Requirements
https://downloads.isc.org/isc/bind9/9.18.19/doc/arm/html/dnssec-guide.html#hardware-requirements

[2-4] DNSViz | A DNS visualization tool
https://dnsviz.net/

[2-5] dsc
https://github.com/DNS-OARC/dsc

[2-6] Prometheus
https://prometheus.io/

[2-7] unbound_exporter
https://github.com/letsencrypt/unbound_exporter

[2-8] bind_exporter
https://github.com/prometheus-community/bind_exporter

[2-9] DNSViz/DNSViz CLI 版
https://github.com/dnsviz/dnsviz

[2-10] DNSViz/DNSViz Docker 版
https://github.com/dnsviz/dnsviz#docker-container

[2-11] dnsops.jp
https://dnsops.jp/

[2-12] dns-oarc
https://www.dns-oarc.net/

第3章 権威 DNS サーバーの DNSSEC 対応

3.1 DNSSEC 対応の基礎

本章の主要な想定読者は「権威 DNS サーバー運用者」ですが、「ドメイン名登録者」も読むべき（SHOULD）内容となっています。ドメイン名登録者には、自分が維持管理するドメイン名の DNS データを作成する責任があります。したがって、DNS データを DNSSEC 対応にするために必要な、幾つかの意思決定を行う必要が出てくるからです。

本章は、読者が DNSSEC の動作と DNS および DNSSEC に関連する用語を理解していることを前提に書かれています。DNSSEC の基本動作については[2-1]を、関連する用語については[2-2]を参照してください。本セクションでは、本章の記述に関係する権威 DNS サーバーの動作の概要を説明します。

権威サーバーが DNSSEC に対応すると、権威を持つゾーンデータの出自と完全性を保証するために必要となる DNSSEC 関連のリソースレコード（DS リソースレコード, DNSKEY リソースレコード, RRSIG リソースレコード）と、データの不在を証明するためのリソースレコード（NSEC リソースレコードまたは NSEC3 リソースレコード、NSEC3 リソースレコードの場合は NSEC3PARAM リソースレコードも）が追加されます。この出自と完全性の保証と不在の証明について、この後の 2 つのサブセクションで説明します。

3.1.1 出自と完全性の保証

「出自と完全性の保証」とは、あるドメイン名に関するリソースレコードがそのドメイン名に対する真正な権威サーバーから得た情報であり、内容が途中で改ざんされていないことを保証することを意味します。

出自と完全性の保証は、信頼の連鎖の構築を介して行われます。実際に信頼の連鎖の構築を行って出自と完全性の保証を行う作業はフルリゾルバーの担当ですが、権威 DNS サーバーはフルリゾルバーが信頼の連鎖を構築できるように、公開鍵暗号に基づいた情報（DNSSEC 関連のリソースレコード）を提供します。

公開鍵暗号と信頼の関係について説明します。公開鍵暗号は、公開鍵と秘密鍵で構成される一対の鍵ペアを使用します。公開鍵はその名のとおり公開されます。DNSSEC では、DNSKEY リソースレコードとしてゾーンに現れます。秘密鍵は秘密に維持されます。公開鍵暗号の特性として、公開鍵で暗号化したデータは秘密鍵で復号でき、秘密鍵で署名したデータは公開鍵で検証できるというものがありますが、DNSSEC では後者を利用します。具体的に、DNS ゾーン内にあるリソースレコード（例えば A リソースレコード）の情報を扱いやすいように固定長のデータに変換し（ハッシュ化と呼びます）、それを秘密鍵で署名し、署名したデータを公開します。DNSSEC では RRSIG リソースレコードとして公開されます。DNSEC に対応したフルリゾルバーは、A リソースレコードと一緒に

A リソースレコードに付随する RRSIG リソースレコードと、そのゾーンの DNSKEY リソースレコードを取得します。次に RRSIG リソースレコードのデータを DNSKEY リソースレコードのデータ(公開鍵)で検証し、A リソースレコードのハッシュ化された値と照合します。その後自分が取得した A リソースレコードを同じ方法で照合できれば、①A リソースレコードは DNSKEY リソースレコード(公開鍵)に対応する秘密鍵の持ち主、つまり正当なゾーン所有者によって提供されたものであり、②そのデータは改ざんされていないことが保証されます。図 6 は、フルリゾルバーが www.example.jp のアドレス(A レコード)を解決する際に、信頼の連鎖がどのように構築されるのかを示しています。

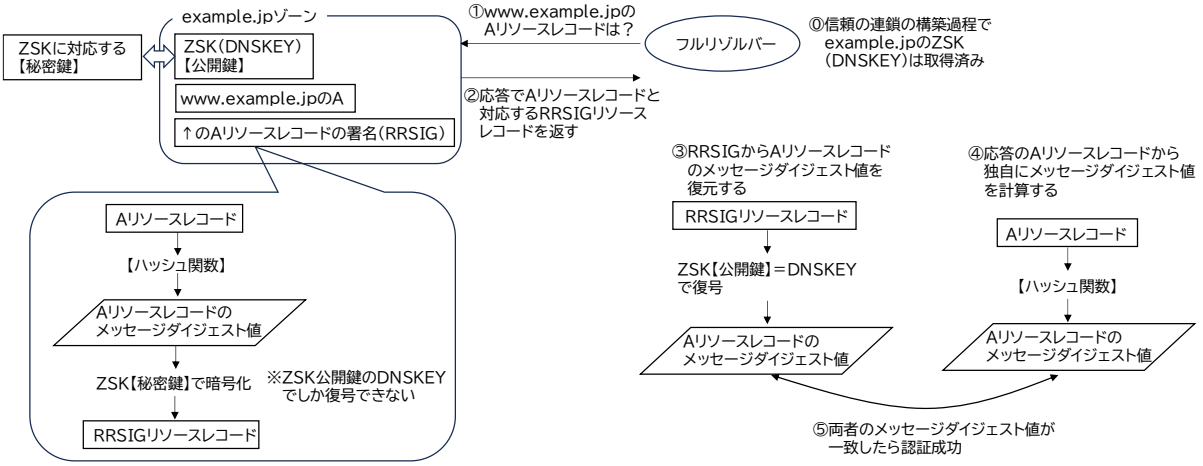

図 6 フルリゾルバーが www.example.jp の A リソースレコードを認証する例(RSA の場合)
出典:総務省(令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム)

次に信頼を「連鎖」させるという意味を説明します。上の説明では、そのデータは確かに DNSKEY リソースレコードに対応する秘密鍵の持ち主からやってきたことは保証されますが、その持ち主が本当にドメイン名(例えば example.jp)の持ち主であるかは保証されません。それを実現する部分が信頼の連鎖に相当します。DNS は、ルートゾーンの権威 DNS サーバーが jp ゾーンの権威 DNS サーバーを委任し、jp ゾーンの権威 DNS サーバーが example.jp ゾーンの権威 DNS サーバーを委任するという階層的な構成となっているため、それをそのまま信頼を保証する仕組みに当てはめます。具体的には、example.jp ゾーンの DNSKEY リソースレコードを親である jp ゾーンが保証し、jp ゾーンの DNSKEY リソースレコードを親であるルートゾーンが保証すればよいのです。これを具体的な仕組みに落とし込む際に、親ゾーンと子ゾーンのやり取りの回数を減らしたいなど、幾つかの運用上の理由から、DNSKEY リソースレコードを実際にゾーンデータに署名する鍵(ZSK)と ZSK は真正だと保証する鍵(KSK)の 2 つに分けた上で、KSK データをハッシュ化した値を DS リソースレコードとして親ゾーンに登録・公開し、親ゾーンの ZSK で署名してそれが真正だと保証するという形態になっています。図 7 は、www.example.jp を例として、信頼が連鎖していく様子を示しています。

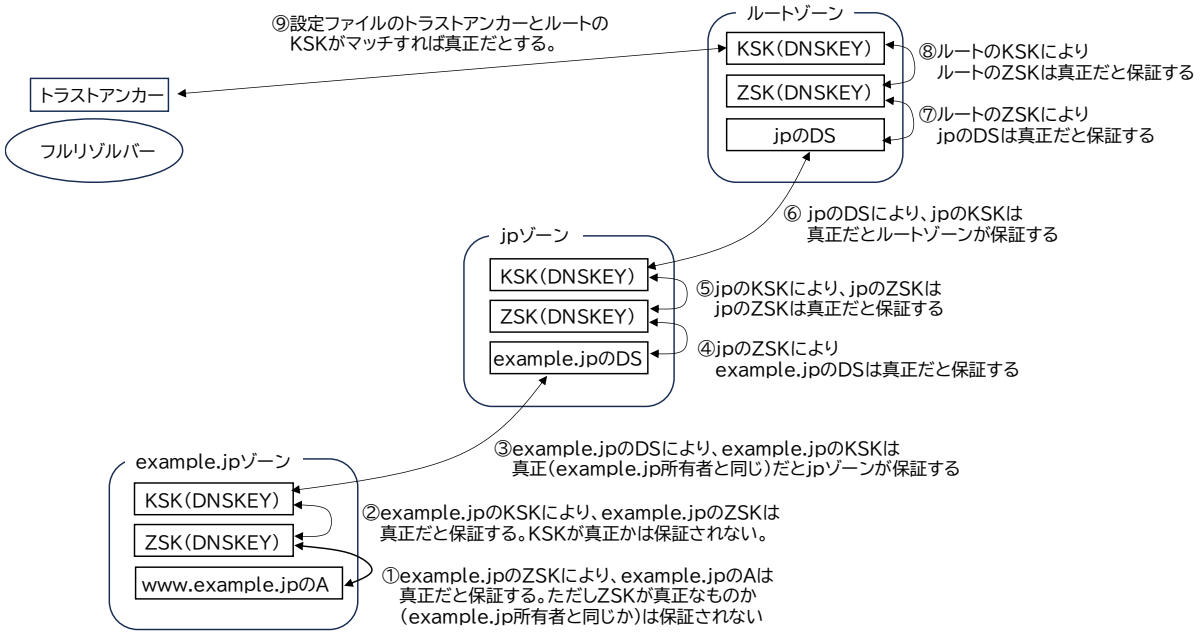

図 7　信頼が「連鎖」する様子(www.example.jp の A リソースレコードの場合)

出典:総務省(令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム)

整理しましょう。具体的に信頼の連鎖を行うためには①何らかの署名アルゴリズムに基づいて 2 種類の鍵ペア(ZSK と KSK)を作成し、それぞれの公開鍵を DNSKEY リソースレコードで公開する、②DNSKEY リソースレコードを KSK の秘密鍵で署名して公開する、③KSK の公開鍵情報をハッシュ化したものを親ゾーンに送り、DS リソースレコードとして親ゾーンの ZSK で署名して公開する、④自分のゾーンの ZSK の秘密鍵でゾーンデータに署名して公開する(ゾーンを構成する各リソースレコードデータをハッシュ化し秘密鍵で暗号化する)という作業が必要となります。使用可能な署名アルゴリズムやハッシュアルゴリズムに関する推奨事項は RFC 8624 が提供しています。

ここで、自分のゾーンの KSK の公開鍵情報をハッシュ化したものを親ゾーンの DS リソースレコードとして公開してもらうためには、通常、ドメイン名登録事業者に DS リソースレコード情報の取り次ぎを依頼する必要があります。したがって、ドメイン名登録事業者が DS リソースレコード情報の取り次ぎに対応していなければ、権威 DNS サーバー自身を DNSSEC に対応させたとしても正常に動作できないことに注意してください(本ガイドライン執筆時点では、RFC 8078 で CDS/CDNSKEY リソースレコードを使用した DS リソースレコード登録を自動化する仕組みが検討されていますが、実運用に至るまでにはまだ時間を要すると考えられます)。

また、KSK と ZSK は主にセキュリティ上の理由で、今まで使用していたものを新しいものに置き換える作業が発生します(ロールオーバーと呼びます)。ZSK のロールオーバーはゾーン内部に閉じた作業なのでそのゾーンの権威 DNS サーバー単体で実施できますが、KSK のロールオーバーは親ゾーン側での作業(新しい KSK 情報を DS リソースレコードで公開して署名する)が必要になるため、作業手順が異なります。またどちらの場合においても、鍵の更新のために信頼の連鎖が途切れないようにするため、手順とタイミングに注意を払う必要があります。

さらに各署名には、リプレイ攻撃を緩和するために有効期間が設定されています。ですから、鍵やゾーンの内容が更新されていない場合であっても、署名有効期間が満了する前に各リソースレコードへの再署名が必要になります。

3.1.2 不在証明

不在証明は、DNS の NXDomain、NoData で表現される「問い合わされたリソースレコードは存在しない」ことを示す不在情報に署名を追加できるようにすることで、指定された名前とその下の名前にはいかなるリソースレコードも存在しない、または指定された名前には指定されたリソースレコードは存在しないことをクライアントで検証可能にするための仕組みです。DNSSEC において不在を証明する応答（不在応答）には NSEC リソースレコードを使用する方法と NSEC3 リソースレコードを使用する方法の 2 種類があるため、DNSSEC 対応の際にどちらを使用するかを選択する必要があります。

NSEC リソースレコードは、ゾーンに含まれる名前を正規順序でソートし、存在しない名前の前に位置する名前と後に位置する名前を提供することで名前の不在を提示し、その名前に存在するすべてのリソースレコードを提供することでリソースレコードの不在を提示します。そのため、NSEC リソースレコードはその性質上、NSEC リソースレコードのリストをたどることで、そのゾーンのすべてのリソースレコードを取得することができてしまいます。これをゾーンウォーキングと呼びます。

NSEC3 リソースレコードは、NSEC リソースレコードと同様の仕組みをハッシュ化したドメイン名で提供することで、ゾーンウォーキングの問題を緩和します。一方で、問い合わされた名前が存在しない場合、問い合わせ名のハッシュ化や適切な NSEC3 リソースレコードの探索が必要となるため、NSEC リソースレコードに比べると CPU に若干負担がかかります。また一般に応答のサイズが、NSEC リソースレコード使用時よりも大きくなります。

NSEC3 リソースレコードを NSEC リソースレコードと差別化する重要な機能に、オプトアウト機能があります。これは、TLD の権威 DNS サーバーのように、ゾーンに設定される内容の大半が子ゾーンへの委任であるような場合に、DNSSEC で保護された委任（セキュアな委任）を徐々に増やせるようにすることを目的として導入されたものです。NSEC3 リソースレコードのオプトアウトフラグを設定すると、NSEC3 リソースレコードの意味が、NSEC3 リソースレコードがカバーする範囲に特定の名前が存在しないという不在証明から、NSEC3 リソースレコードがカバーする範囲に存在する委任は DNSSEC 非対応であるという証明に変更されます。これにより、NSEC3 リソースレコードリストの再計算や再署名を行うことなく、DNSSEC 非対応な委任を追加・変更・削除できるようになります。

3.2 DNSSEC 対応の要件

権威 DNS サーバーを DNSSEC 対応にするためには、満たすべき要件が幾つか存在します。それを以下に提示します。

(1)時刻の同期

3-1.	権威サーバーに設定される時刻が合っていないと、本来有効であるはずの署名がリゾルバーに無効と扱われる恐れがありますから、NTP 等の手段を使用して権威 DNS サーバーの時刻を信頼できる時刻ソースと同期しておかなければいけません (MUST)

DNSSEC の署名情報を保持する RRSIG リソースレコードには、署名の有効期間に関する絶対時刻の情報が含まれます。権威サーバーに設定される時刻が合っていないと、本来有効であるはずの署名がリゾルバーに無効と扱われる恐れがありますから、NTP 等の手段を使用して権威 DNS サーバーの時刻を信頼できる時刻ソースと同期しておかなければいけません(MUST)。

(2)鍵の保護手段の提供

3-2.	ゾーン内のリソースレコードに署名を行うサーバー上には秘密鍵情報が存在する必要があります。したがって、サーバー上に存在する秘密鍵情報に対する何らかの保護手段を設定しなければいけません (MUST)

リソースレコードへの署名は、DNSKEY リソースレコードで公開されている ZSK の公開鍵に対応する秘密鍵で行われます。秘密鍵が第三者に漏洩すると、署名されたリソースレコードが偽造可能になるため、DNSSEC が提供する出自の認証と完全性の保証は失われてしまいます。一方で、ゾーン内のリソースレコードに署名を行うサーバー上には秘密鍵情報が存在する必要があります。したがって、サーバー上に存在する秘密鍵情報に対する何らかの保護手段を設定しなければいけません(MUST)。詳細については後ほどセクション 3.3.3 に記述します。

3.3 導入の準備

3.3.1 ソフトウェア／アプライアンスの対応状況の確認／更新

現在使用中の権威 DNS サーバーソフトウェアまたはアプライアンスが DNSSEC に対応しているかどうかを確認する必要があります。ここで「DNSSEC に対応」とは、付録 2 で示される RFC に準拠していることを指します。

対応していない場合には、ソフトウェアまたはアプライアンスの OS 更新などを行って最新の状態にし、可能な限り準拠した状態を保つようにしてください。

権威 DNS サーバーソフトウェアをパッケージ単位で導入する場合、パッケージのバージョンによって DNSSEC に関連する機能のデフォルト値が異なる可能性があるので、注意してください。

3.3.2 性能の確認／増強

権威 DNS サーバーを DNSSEC 対応にすると、署名作業、不在応答を作成するための作業が追加で発生するようになり、また DNSSEC 対応前には存在しなかった DNSKEY リソースレコード、RRSIG リソースレコード、DS リソースレコード、NSEC リソースレコードや NSEC3 リソースレコードといった DNSSEC 関連のリソースレコードが追加されるため、ゾーンデータのサイズが肥大化します。

最終的にシステム資源に課される負担は、ゾーンの構造、署名アルゴリズム、鍵の数、NSEC リソースレコードか NSEC3 リソースレコードかの選択、署名された委任の割合、ゾーンファイルのフォーマットなどに依存するので、一概には言えませんが、参考になると考えられる関連情報として、BIND の DNSSEC Guide の中にハードウェア要件に関する記述があります[3-3]。その内容を抜粋すると、先に挙げた要因に応じて変動はしますが、ゾーンファイルの増加量は無視できる程度から 3 倍程度くらいの範囲になり、メモリ消費量は未署名時の 1.5 倍程度になるとされています。

一般的には、現代的なハードウェアを使用するサーバー構成であれば DNSSEC 対応にすることの負担は気にしなくてもよいと考えられます。

3.3.3 構成および鍵の保護手段の検討

3-3.	DNSSEC Signer はその性質上、署名用の秘密鍵(ZSK の秘密鍵)にアクセスできる必要があります。秘密鍵が第三者に漏洩すると DNSSEC のセキュリティ機能が損なわれるため、不特定多数がアクセス可能な公開サーバー上で秘密鍵を保持するのは避けるべきです （SHOULD）

本ガイドラインでは、ゾーンのリソースレコードに署名作業を行うサーバーを DNSSEC Signer と呼ぶことにします。DNSSEC Signer はその性質上、署名用の秘密鍵(ZSK の秘密鍵)にアクセスできる必要があります。秘密鍵が第三者に漏洩すると DNSSEC のセキュリティ機能が損なわれるため、不特定多数がアクセス可能な公開サーバー上で秘密鍵を保持するのは避けるべきです(SHOULD)。考えられる構成例については、セクション 3.6.1 で紹介しています。

3-4.	KSK の秘密鍵は、原則はオフラインにて保管し、必要な場合にだけ DNSSEC Signer にロードして使用し、使用後は削除するという運用にすべきです (SHOULD)

KSK の秘密鍵は、ZSK の更新時に新しい DNSKEY リソースレコードセットに署名する際にしか使用されません。ですから原則はオフラインにて保管し、必要な場合にだけ DNSSEC Signer にロードして使用し、使用後は削除するという運用にすべきです(SHOULD)。

運用上またはセキュリティ上の理由で、DNSSEC Signer 上に置かれる秘密鍵により強力な保全措置が必要とされる場合があります。そのための手段として、HSM(ハードウェアセキュリティモジュール)が挙げられます。HSM は内部のデータ(秘密鍵)を外部に持ち出すことが不可能な設計となっており、金融業界や認証局など、重要な場面で導入されることがあります(DNSSEC ではルートゾーンの鍵管理に導入されています)。もし HSM の導入を検討する場合には、FIPS 140-3[3-1]を調査するとよいでしょう。

3.3.4 署名方法の決定

3-5.	署名作業は、あるレコードに対して一度行ったら終わりではなく、署名の有効期間が終了する前に同じレコードに対して再度署名をし直す必要があります。手動による作業は間違いが起こりやすいので、特別な理由がない限り、DNSSEC に関連する作業には自動化の仕組みを導入すべきです (SHOULD)

署名作業は、サーバーソフトウェアやアプライアンス付属のツールか、独立に開発されたツールを使用して行われます。署名作業は、あるレコードに対して一度行ったら終わりではなく、署名有効期間が終了する前に同じレコードに対して再度署名をし直す必要があります。手動による作業は間違いが起こりやすいので、特別な理由がない限り、DNSSEC に関連する作業には自動化の仕組みを導入すべきです(SHOULD)。

自動化の仕組みを導入する場合、事前に幾つかのパラメーターを決めておく必要があります。ただし、パラメーターは絶対的なものではなく、運用状況に応じて後日変更も可能です。検討すべきパラメーターは以下の 6 つです。これらは広い意味で DNSSEC 運用の方針とも呼べるものなので、DNSSEC ポリシーと呼ばれます。

(1)署名アルゴリズム

署名および署名検証をする際に使用されるハッシュ化方式、公開鍵暗号の方式を選択する必要があります。2022 年時点で、どのようなアルゴリズムが実際に使用されていたのかについて、ICANN が公開している OCTO Publications の OCTO-33 に記述があります。それによると、RSA-SHA-256 が 65.1%、ECDSA-P256-SHA256 が 30.1%使用されていました。また RSA 使用時の鍵長については、KSK は 2048 ビットが 98.6%、ZSK は 1024 ビットが 82.7%で支配的でした。

3-6.	新規に DNSSEC を導入するのであれば、KSK/ZSK ともに暗号アルゴリズムとしてはまず ECDSA-P256-SHA256 を検討すべき （SHOULD）
3-7.	サーバーソフトウェア/アプライアンスやロードバランサー、ファイアウォールが未対応の場合には、広く普及している RSA-SHA-256 でサービスを開始し、問題が解消した時点で ECDSA-P256-SHA256 への移行を検討してもよい （MAY）

2023 年末に Verisign が.com、.net、.edu TLD のリソースレコードに署名する ZSK の暗号アルゴリズムを ECDSA-P256-SHA256 に変更したように、徐々に RSA から ECDSA への移行が始まっています。ECDSA-P-256-SHA256 は RSA-SHA-256（2048 ビット）と同等以上のセキュリティ強度を保ちながら署名サイズを小さくできますので、新規に DNSSEC を導入するのであれば、KSK/ZSK ともに暗号アルゴリズムとしてはまず ECDSA-P256-SHA256 を検討すべきです（SHOULD）。ただし、サーバーソフトウェア/アプライアンスやロードバランサー、ファイアウォールが ECDSA による署名方式に未対応の場合に、広く普及している RSA-SHA-256 でサービスを開始し、問題が解消した時点で ECDSA-P256-SHA256 への移行を検討してもよいです（MAY）。

（2）DNSKEY リソースレコード関連のパラメーター

3-8.	扱うゾーンの規模が大規模でない限りはサーバーソフトウェアやアプライアンスが提供するデフォルトの値を使用すべきです （SHOULD）

DNSSEC の運用開始時に、DNSKEY リソースレコード関連のパラメーターを 3 つ決める必要があります。鍵のロールオーバーに関するものが幾つかあるので、事前に RFC 7583[3-2]を参照しておくと理解の助けになります。一般に、扱うゾーンの規模が大規模でない限りはサーバーソフトウェアやアプライアンスが提供するデフォルトの値を使用すべきです（SHOULD）。

3-9.	DNSKEY リソースレコードの TTL は、フルリゾルバーにキャッシュされるため、極端に短い期間を指定することは避けるべきです （SHOULD）

①DNSKEY リソースレコードの TTL

一般的に、フルリゾルバーにキャッシュされるため、極端に短い TTL を指定することは避けるべきです（SHOULD）。

②DNSKEY リソースレコードの事前公開期間

DNSKEY リソースレコード（ZSK か KSK かは問いません）を権威 DNS サーバーで公開後、実際に使い始めるまでの待機期間を指定します。

③リタイアした DNSKEY リソースレコードの事後公開期間

それまで使用されていた DNSKEY リソースレコード（ZSK か KSK かは問いません）が新しいものに置き換えられた後に、古い DNSKEY リソースレコードをどの程度残しておくかの待機期間を指定します。DNSKEY リソースレコードが更新されても、古い DNSKEY リソースレコードで作成された署名（RRSIG リソースレコード）がキャッシュ DNS サーバーにキャッシュされていますので、それらがキャッシュから削除されるまでは古い RRSIG リソースレコードも検証可能である必要があることからこのパラメーターが必要となります。

（3）署名有効期間

署名有効期間とは、RRSIG リソースレコードのデータ内に存在する「開始時刻」と「終了時刻」の期間を指します。リソースレコードの TTL 値と異なる点は、キャッシュしてもよい期間ではなくリゾルバーが DNSSEC 検証に使える期間を表していることです。設定にあたっては以下の点に注意する必要があります。

・署名がカバーするリソースレコードの TTL よりも署名有効期間が短い場合、リソースレコードがキャッシュされた際に、一緒にキャッシュされた対応する RRSIG リソースレコードの署名有効期間が満了する可能性があります。したがって、署名有効期間はゾーン内の署名対象となるすべてのリソースレコードの TTL よりも長くなければいけません。

・SOA リソースレコードの Expire（ゾーン情報の更新ができない状態が続いた場合、セカンダリサーバーが持っているデータを継続して利用できる最大時間）よりも署名有効期間が短い場合、セカンダリサーバーがプロトコルで許容される範囲でゾーン情報を公開していても、そこに含まれる RRSIG リソースレコードが有効期間満了になる可能性があります。したがって、署名有効期間は SOA リソースレコードの Expire よりも長くなければいけません。

3-10. 再署名までの間隔が短いと、署名有効期間を過ぎてしまい DNSSEC の署名検証に失敗する可能性が生じます。通常は終了時刻－TTL 値より前に再署名が実行されるべきです （SHOULD）

・署名の有効性を確保するため、再署名までに十分な期間を設定する必要があります。再署名から署名有効期間の終了時刻までの間隔が短いと、署名有効期間を過ぎてしまい DNSSEC の署名検証に失敗する可能性が生じます。通常は終了時刻－TTL 値より前に再署名が実行されるべきです（SHOULD）。

・署名有効期間の上限は、一般に ZSK そのものの寿命と考えられます。現在の鍵の寿命は、インターネット上に暗号データを公開してから解析されるまでの時間という考え方に基づいています。以

上により、署名有効期間は署名から一定期間以上で、ZSK の寿命よりも短い値になると考えられます。

(4)鍵の使用期間または更新間隔

ZSK や KSK をどれくらいの期間使い続けるかを決定する必要があります。これを鍵の使用期間と呼びます。鍵の使用期間が終了するまでに、古い鍵を新しい鍵に置き換える作業が必要となります。これを鍵のロールオーバーと呼びます。ここで、鍵の使用期間とは、この鍵をこれくらいの期間使用すると運用者が決めた期間というだけであり、それを超えても鍵を使用し続けることはできます。プロトコル的に定められた鍵の使用可能期間の上限値はありません。例えば署名の場合には、署名有効期間が終了した後は署名検証に失敗するようになりますが、鍵の場合には、使用期間が終了したとしても、署名を検証するリゾルバーの検証結果には影響しません。

3-11.	署名アルゴリズムとして RSA よりも効率的な ECDSA が RFC 6781 の執筆時よりも普及していることから、ECDSA を採用した場合の使用期間はある程度柔軟に、より長い期間を設定してもよい (MAY)

鍵の使用期間として、これまでは DNSSEC の運用ガイドラインである RFC 6781 にしたがって KSK は 1 年、ZSK は 1 ヶ月から 3 ヶ月という推奨がなされてきました。しかし、署名アルゴリズムとして RSA よりも効率的な ECDSA が RFC 6781 の執筆時よりも普及していることから、ECDSA を採用した場合の使用期間はある程度柔軟に、より長い期間を設定してもよい(MAY)と考えられます。

(5)鍵のロールオーバー方式

ZSK と KSK それぞれについて、ロールオーバー方式を決定する必要があります。選択可能なロールオーバー方式と、ロールオーバーのタイミングに関する詳細な説明については RFC 7583[3-2]を参照してください。

3-12.	KSK は CDS/CDNSKEY リソースレコードを使用した DS リソースレコードの更新方法が規定されていますが、本ガイドライン執筆時点では広く普及していないことから、運用者の手作業が発生せざるを得ません。その場合、二重 KSK 法(または二重署名法)を使用すべき (SHOULD)

ZSK については、使用する自動化ツールが使用する方式に従えば問題ありません。KSK は CDS/CDNSKEY リソースレコードを使用した DS リソースレコードの更新方法が規定されていますが、本ガイドライン執筆時点では広く普及していないことから、運用者の手作業が発生せざるを得ません。その場合、二重 KSK 法(または二重署名法)を使用すべき(SHOULD)です。

(6)不在証明の実現方法

3-13. ゾーンウォーキングが問題になるのは TLD など大量の委任を持つ特殊なゾーンに限られますので、特別な理由がなければ NSEC を選択すべきです （SHOULD）

不在証明を NSEC リソースレコードで実現するか NSEC3 リソースレコードで実現するかを選択する必要があります。NSEC3 パラメーターの設定に関するガイダンスを定めた RFC 9276 は、オプトアウト機能を使用する必要があるか、ゾーンウォーキングのリスクを低減する必要がある場合には NSEC3 リソースレコードを使用し、そうでないなら NSEC リソースレコードを優先して使用すべきであると提言しています。一般に、ゾーンウォーキングが問題になるのは TLD など大量の委任を持つ特殊なゾーンに限られますので、特別な理由がなければ NSEC を選択すべきです（SHOULD）。

3-14. NSEC3 リソースレコードを使用する場合、RFC 9276 の規定に従い、イテレーションは 0 を使用しなければいけません （MUST）
3-15. NSEC3 リソースレコードを使用する場合、ソルトは使用してはいけません （MUST NOT）

NSEC3 リソースレコードを使用する場合、RFC 9276 の規定に従い、イテレーションは 0 を使用しなければいけません(MUST)。またソルトは使用してはいけません(MUST NOT)。

3.4 運用

3.4.1 日常的な監視と確認

現在でも、運用中の権威 DNS サーバーに対して DNS 応答や生存確認などの監視は実施していると思われますが、ここでは特に、DNSSEC を有効化した後に追加すべき監視のポイントを説明します。

(1)時刻同期状況の確認

3-16. 権威 DNS サーバーに設定される時刻がずれると提供する署名の検証に失敗する可能性が高まるので、時刻同期状況は必ず確認しなければいけません (MUST)

権威 DNS サーバーの時刻が合っていない場合、提供する署名の検証に失敗する可能性が高まるので、時刻同期状況は必ず確認しなければいけません(MUST)。

(2)権威 DNS サーバーが提供するデータが署名検証できるか

3-17. 権威 DNS サーバーが提供するデータまでの信頼の連鎖が構築可能であることを確認しなければいけません (MUST)

管理対象の権威 DNS サーバーが提供するゾーンデータまでの信頼の連鎖が構築可能であることを確認しなければいけません(MUST)。具体的には、親ゾーンの DS リソースレコードと自ゾーンの DNSKEY リソースレコードの対応関係や、必要とされる DNSSEC 関連のリソースレコード全て(DNSKEY リソースレコード、RRSIG リソースレコード、NSEC または NSEC3 リソースレコード)がゾーンに存在することを確認する必要があります。

これらの監視を行うためには DNSSEC 署名を検証可能なツールを用意するか、DNSSEC 対応のフルリゾルバーを利用して認証済みを示す AD ビット付きの応答を確認するなどの方法が考えられます。

(3)RRSIG リソースレコードの確認

3-18. DNSSEC 検証エラーが発生するリスクを軽減するため、RRSIG リソースレコードの署名有効期間、特に終了時刻が想定よりも切迫していないかを監視すべきです (SHOULD)

DNSSEC 検証エラーが発生するリスクを軽減するため、RRSIG リソースレコードの署名有効期間、特に終了時刻が想定よりも切迫していないかを監視すべきです（SHOULD）。

自動化ツールを導入している場合、ポリシーに従いゾーンの再署名が一定期間毎に自動的に行われますから、終了時刻が迫る事態は起こり得ないはずです。しかし、自動化ツールといえども万能ではなく、誤動作や異常動作が発生する可能性は否定できません。そのような状況になった場合、本来行われるはずだった再署名が行われずに署名有効期間が満了してしまう可能性もあります。

この監視を行う場合、（2）の監視で署名検証ができることは確認済みのはずですから、RRSIG リソースレコードの終了時刻のみを監視すればよいことになります。

（4）鍵のロールオーバーの進行状況および正常性の確認

3-19. 鍵のロールオーバーが進行中の場合、RFC 7583 を参照しながら想定されたとおりに作業が進行していることを確認すべきです （SHOULD）

鍵のロールオーバーが進行中の場合、ロールオーバーのタイミングに関する考慮事項を記述した RFC 7583 を参照しながら、想定されたとおりに作業が進行していることを確認すべきです（SHOULD）。

3.4.2 ログによる異常の有無の確認

3-20. 日常的なゾーン情報のロード・更新に関する情報や権威サーバーへのアクセスログに加え、自動化されたプロセス（署名の更新やロールオーバー）に関するエラー情報を確認するようにすべきです （SHOULD）

日常的なゾーン情報のロード・更新に関する情報や権威サーバーへのアクセスログに加え、自動化されたプロセス（署名の更新やロールオーバー）に関するエラー情報を確認するようにすべきです（SHOULD）。

3.4.3 鍵の管理に関する注記

DNSSEC では ZSK、KSK という 2 種類の鍵ペアが必要となりますが、これらの鍵は生成、運用に加えてバックアップが必要となります。先に述べたように、秘密鍵は署名作業に必要なため、DNSSEC Signer 上に置かれますが、ハードウェアトラブル、ソフトウェアのバグ、手続き上の不備などの理由で消失する可能性があります。そのような状況から速やかな復旧をするためには、鍵情報のバックアップは欠かせません。

一般に、鍵情報のバックアップは権威 DNS サーバー運用者が行うと考えられますが、権威 DNS サーバー運用者とドメイン名登録者の契約内容次第ではドメイン名登録者が責任を負う場合があります。したがって、鍵の管理においては権威 DNS サーバー運用者と、ドメイン名登録者においては管理責任がどちらにあるのかを事前に契約約款などで確認しておくべきです。

3.4.4 DNSSEC に関連する協調作業への対応

権威 DNS サーバーやその運用者の変更、運用者が異なる複数の権威 DNS サーバーを利用する場合、DNSSEC に関連する協調作業が求められる場合があります。行うべき作業は移転元で DNSSEC を使用しているか、移転先で DNSSEC を使用するかによって異なります。詳細については[3-3]を参照してください。

3-21. DNSSEC 署名を維持したまま権威 DNS サーバー運用サービスを変更することを可能にする手段として、RFC 8901 で Multi-Signer DNSSEC モデルが規定されています。将来このモデルへの対応が必要となることがあるかもしれませんが、本ガイドライン執筆時点では、直ちに対応しなくてもよい （MAY）

DNSSEC 署名を維持したまま権威 DNS サーバー運用サービスを変更することを可能にする手段として、RFC 8901 で Multi-Signer DNSSEC モデルが規定されています。将来このモデルへの対応が必要となることがあるかもしれませんが、本ガイドライン執筆時点では、直ちに対応しなくてもよい(MAY)と考えられます。

3.4.5 鍵のロールオーバーについて

ここで、鍵のロールオーバーについて整理しておきます。DNSSEC では、ZSK/KSK という 2 種類の鍵ペアを使用するため、それぞれについてロールオーバーが必要です。

3-22. ZSK のロールオーバーについては、セクション 3.3.4 で説明した DNSSEC ポリシーに従い、自動化ツールで処理を行うべきです(セクション 4.2 に記述したようにロールオーバー中の監視はすべきです （SHOULD）

ZSK のロールオーバーについては、セクション 3.3.4 で説明した DNSSEC ポリシーに従い、自動化ツールで処理を行うべきです(セクション 4.2 に記述したようにロールオーバー中の監視はすべきです(SHOULD))。

KSK のロールオーバーについては、現時点で自動化の仕組みの開発・普及が始まった段階にありますから、自動化までにはもうしばらく時間がかかります。したがって人手によるコマンド入力作業や、親ゾーン側の状態確認作業などが必要となります。

3-23. DNSSEC の運用開始から KSK の第 1 回ロールオーバーまでには少なく見積もっても 1 年、ECDSA などの強力な暗号鍵を使用している場合にはもう少し後でもよい

（MAY）

ただし、DNSSEC の運用開始から KSK の第 1 回ロールオーバーまでには少なく見積もっても 1 年、ECDSA などの強力な暗号鍵を使用している場合にはもう少し後でもよい（MAY）と考えられますから、その間に時間をかけて情報を集め、準備を進めてください。

なお、これらの定期的なロールオーバーとは別に、緊急の鍵のロールオーバーが発生する可能性があります。これは、使用している鍵が外部に漏洩しもはや安全ではなくなった場合に緊急に鍵を置き換える作業を指します。こちらについてはセクション 3.5.3 項を参照してください。

3.5 トラブルシューティング

ここでは、権威 DNS サーバーに起こりえる DNSSEC のよくあるトラブルについて、事象と対策を交えながら解説します。

3.5.1 原因の切り分けと対応

基本的には従来の(DNSSEC 未対応時の)権威 DNS サーバーのトラブルシューティングに DNSSEC に関係する要素が追加される形になります。

したがって、まず全ての権威 DNS サーバーへの UDP/TCP 経由での到達性を確認してください。DNS の TCP 対応は RFC 7766 で既に必須(RFC で MUST 指定)になっていますが、中間にあるファイアウォールやロードバランサーの設定変更時に TCP ポートが閉じられてしまうトラブルが時折見受けられます。次いで、全ての公開権威 DNS サーバーが同じデータを保持していることを確認し(プライマリからのゾーン転送が原因でないことを切り分け)、問い合わされた応答に対して(DNSSEC 関連のリソースレコードの正しさはさておき)何かしらの応答が返されることを確認してください。

ここからが DNSSEC の問題の切り分け段階となります。DNSSEC の異常は、大きく分けて以下の 3 つに分類できます。

(1)信頼の連鎖の破綻

権威 DNS サーバーが提供する DNSKEY リソースレコード(KSK)と、親ゾーンに登録された DS リソースレコードが対応しない場合に発生します。セクション 4.2 の日常的な監視で発見される場合がほとんどであると考えられます。監視ツールの出力に加えて、DNSViz[2-6]のような外部ツールや、RFC 8914 で標準化された拡張 DNS エラー(EDE)対応のフルリゾルバーやパブリックリゾルバーを活用して裏付けを取ると良いでしょう。

3-24. 親ゾーンの DS リソースレコードと自ゾーンの DNSKEY リソースレコードを照合することになります。照合の際には鍵タグの照合だけにとどめず、ダイジェスト値も照合すべき(SHOULD)

事象としては①親ゾーンの DS リソースレコードが誤っている(または存在しない)、②権威 DNS サーバー側に親ゾーンの DS リソースレコードに対応する DNSKEY(KSK)リソースレコードが存在しない(または誤っている)、のどちらかになりますので、DS リソースレコードと DNSKEY リソースレコードを照合することになります。照合の際には鍵タグの照合だけにとどめず、ダイジェスト値も照合すべきです(SHOULD)。ダイジェスト値の導出には、例えば BIND 9 に付属の dnssec-dsfromkey コマンドなどが使用できます。

なお、DS リソースレコードや DNSKEY リソースレコードに付随する署名(RRSIG リソースレコード)が誤っているまたは存在しないという理由で信頼の連鎖が構築できない場合がありますが、これは署名検証の異常に分類されるため(2)を参照してください。

(2)DNS データの不備による署名検証の失敗

権威 DNS サーバーが提供する DNS データに何らかの不備が生じて、署名検証に必要な情報が欠落している場合に発生します。セクション 4.2 の日常的な監視で発見される場合もあるかもしれませんが、ランダムサンプリングのような形で検査している場合には外部からの申告で発覚する場合もあります。この場合も、DNSViz[2-6]のような外部ツールや、RFC 8914 で標準化された拡張 DNS エラー(EDE)対応のフルリゾルバーやパブリックリゾルバーが裏付けに活用できます。

全 DNS データに影響する不備であれば日常的な監視で発見できるはずです。親ゾーンとの信頼の連鎖の維持が確認されているのであれば、そこから先に原因があります。RRSIG リソースレコードと DNSKEY(ZSK)リソースレコードが対応しているか、RRSIG リソースレコードの署名有効期間が過ぎていないかなどを順番に検査してください。

次に外部からの申告で、一部のリソースレコードだけが署名検証に失敗するという場合を取り上げます。自動署名ツールを使用している場合、設定によっては各リソースレコードの署名更新時期を少しずつずらしながら再署名を行います。すると、あるタイミングで自動署名ツールが誤動作した場合に一部の署名だけがおかしいということが発生し得ます。その場合は自動署名ツールのログを確認の上で必要な対処をしていくことになります。なお、一部のリソースレコードだけが署名検証に失敗するという申告を受けて調査しても権威サーバー側で何も問題が見つけられない場合もあり得ます。その場合については(3)を参照してください。

(3)リソースレコードデータの改ざんによる署名検証の失敗

これは外部のフルリゾルバーがキャッシュ汚染攻撃などのために偽造されたリソースレコードデータを受信した場合に発生する事象なので、日常的な監視では発見できません。外部からの申告受理後に(1)(2)の検査を経て全て異常が見つからなかった(外部ツールで裏付けを強化してもよいでしょう)場合に、このタイプだと切り分けられます。

この場合、フルリゾルバーで署名検証が失敗することは DNSSEC として正しい動作なので(署名検証に失敗しないと偽造データが正当なものとして使用されてしまいます)、権威 DNS サーバー側ではこれ以上の対処はできません。外部ツールの証跡などを添付し、キャッシュ汚染攻撃の可能性を申告者に通知することになります。

3.5.2 緊急時に DNSSEC を無効化すべきか

DNSSEC に関連する異常が発生した場合、問題解決を短縮化するため、親ゾーンの DS リソースレコードを一旦削除して DNSSEC を無効にしたいと考えることがあるかもしれません。
DNSSEC の無効化は一見有効な手段に見えるかもしれませんが、幾つかのリスクがあることを認識しておく必要があります。

まず DNSSEC を無効化すると、当然ですが DNSSEC が提供していた保護もなくなりますので、偽造された DNS データであっても受理されるようになります。これは、DNS を介して提供して

いる様々なサービス、例えば Web、メールなどのサービスの全利用者が不正なサイトに誘導される可能性がある危険な状態になることを意味します。

また、DNSSEC の無効化は直ちに実現するわけではありません。DS リソースレコードの登録・更新や削除はドメイン名登録事業者経由で行われて、そこからの作業になります。例えば、ドメイン名登録事業者が日中の営業時間帯にしか申請を処理しない場合、深夜帯に発生したトラブルの対策として DNSSEC の無効化を試みても翌営業時間開始までは何もしてもらえません。加えて、そこから親ゾーン側の作業になりますから、DS リソースレコードの削除まで想定外の時間がかかる可能性があります。

さらに、DS リソースレコードが削除されても、直ちに DNSSEC が無効になるわけではありません。フルリゾルバーには、削除された DS リソースレコードがキャッシュされている可能性があるからです。

3-25. DNSSEC の無効化には幾つかのリスクがありますから、緊急時の対応手段の最初に検討するものではありません。別の手段での対策をまず検討し、最終手段として無効化を選択するという位置づけにすべきです （SHOULD）

以上のように、DNSSEC の無効化には幾つかのリスクがありますから、緊急時の対応手段の最初に検討するものではありません。別の手段での対策をまず検討し、最終手段として無効化を選択するという位置づけにすべきです（SHOULD）。

3.5.3 秘密鍵が漏洩・消失した場合の対応

使用している秘密鍵が外部に漏洩すると、攻撃者は署名検証に成功するデータを偽造できるようになります。そうなると、DNSSEC が提供するセキュリティは保証されなくなってしまいますから、緊急に鍵を置き換える作業が必要となります。

あるいは、（特に KSK の）秘密鍵が障害やトラブルで失われてしまい、バックアップデータからの復旧ができなかった場合もまた、鍵の置き換えが必要となります。

鍵の緊急ロールオーバーに関しては RFC 6781 に記述があります。秘密鍵が外部に漏洩した場合、信頼の連鎖を維持したままロールオーバーを行うこともできますし、信頼の連鎖を一旦壊してロールオーバーを行うこともできます。状況にもよりますが、セクション 5.2 の記述に従い、可能な限り信頼の連鎖を維持した方がよいでしょう。秘密鍵が失われてしまった場合、もはや KSK を使用した再署名ができないので、信頼の連鎖を一旦壊す以外の選択肢はありません。

3.5.3.1 KSK の緊急ロールオーバー手順：信頼の連鎖を維持する場合

1. 新しい KSK を DNSKEY リソースレコードセットに追加します。この段階では、漏洩した KSK はまだ DNSKEY リソースレコードセット内に残されます。DNSKEY の TTL を短く設定し直し、DNSKEY リソースレコードセットがキャッシュから短期間に削除されるようにしておきます。

2. 新たに追加した KSK と、漏洩した KSK の両方で DNSKEY リソースレコードセットに署名します。この際に、署名有効期間の値設定に工夫が必要となります。具体的には、親ゾーンで新しい DS リソースレコードが公開されて古い DS リソースレコードがキャッシュから消去されるまでの期間よりも少しだけ長めに設定してください。

3. 新しい KSK に対応する DS リソースレコードを、ドメイン名登録事業者を介して登録申請します。

4. 二重 KSK 法によるロールオーバーを行います。具体的には、親ゾーンで DS リソースレコードが公開された後、古い DS リソースレコードのキャッシュが破棄されるのを（当該 DS リソースレコードの TTL の期間だけ）待ちます。

5. 漏洩した古い KSK を DNSKEY リソースレコードセットから削除し、新しい KSK だけで DNSKEY リソースレコードセットに署名します。この際に、DNSKEY リソースレコードの TTL や、署名有効期間を本来の値に設定し直します。

3.5.3.2 KSK の緊急ロールオーバー手順：信頼の連鎖を一旦壊す場合

1. ドメイン名登録事業者を介して、親ゾーンの DS リソースレコードを削除する申請をします。削除までには一定の時間を要することに注意してください。

2. DS リソースレコードが削除されると、署名を検証するリゾルバーからは DNSSEC の状態が Insecure（DS リソースレコードが無く信頼の連鎖が途切れている）に見えるようになります。

3. 削除された DS リソースレコードのキャッシュが破棄されるのを（当該 DS リソースレコードの TTL の期間だけ）待ちます。この期間はゾーンファイルを一切変更してはいけません。この段階は非常に大切で、これを省略すると、DNSSEC の状態が Bogus になり事態が悪化するリスクが発生します。

4. DNSKEY リソースレコードセットから古い KSK を削除して、新しい KSK を追加します。DNSKEY リソースレコードセットは新しい KSK で署名します。

5. 新しい KSK に対応する DS リソースレコードを、ドメイン名登録事業者を介して登録申請します。

6. 親ゾーンに DS リソースレコードが登録された時点で信頼の連鎖が復活します。

3.5.3.3 ZSK の緊急ロールオーバーについて

ZSK が漏洩/紛失した場合、親ゾーンとのやりとり（DS リソースレコードの追加や削除など）は必要ないため、権威 DNS サーバー単体で問題を解決できます。

基本的には、自動化ツールが定期的に自動で行っている更新処理を、直ちに手動で再現するという形になります。

3.6 運用ノウハウ

本セクションでは、これまでに記述してきたガイドラインを実装する場合に採り得る選択肢や、運用上知っておくと有用な情報などを提供します。

3.6.1 サーバー構成

DNSSEC Signer 上にある秘密鍵を保護する一つの方法として、DNSSEC Signer をゾーンの NS リソースレコードとして設定されない隠れプライマリ(hidden primary)として設定し、公開用 DNS サーバーをそのセカンダリサーバーとして設定する構成が考えられます。そのような構成であれば、隠れプライマリへのアクセスをセカンダリサーバーだけに制限することで ZSK の秘密鍵の安全を確保できます。

権威 DNS サーバーが管理するゾーン数が少ない場合には、同一機器上で未署名ゾーンをロードして署名を行っても問題ありませんが、ISP のように多数のゾーンを運用している場合、ゾーンの管理作業と署名作業を分離したいという要望があるかもしれません。そのような場合、ゾーンの管理専用のサーバー(ゾーンは未署名のまま追加、削除、変更が管理される)と署名用の DNSSEC Signer を分離することもできます。具体的には、ゾーン管理用のサーバーから DNSSEC Signer に未署名のゾーンを転送し、そこで署名を行った上で公開用サーバーに署名済みのゾーンを転送する構成となります。

3.6.2 鍵のライフサイクルの共有

DNSSEC では、ZSK、KSK それぞれが作成され、使用され、破棄されるという一連の状態の変化をたどります。これを鍵のライフサイクルと呼びます。現在使用している鍵がライフサイクルのどの状態にあるのかを運用者が把握しお互いに意思疎通を図るために、鍵がどのようなライフサイクルをたどるのかを定義し共有しておくと便利です。

鍵の状態は幾つかの文書でそれぞれ少しずつ異なるものが定義されていますが、ここではロールオーバーのタイミングに関する考慮事項を記述した、RFC 7583 に記述されるものを紹介します。

Generated

鍵が作成されてまだ使用されていない(DNSKEY リソースレコードとしてゾーンに登録されてもいない)状態。

Published

ZSK の場合、鍵が DNSKEY リソースレコードとしてゾーンに登録され、その時に Active 状態にある KSK で署名された状態。

KSK の場合、鍵が DNSKEY リソースレコードとしてゾーンに登録され、その時に Active 状態にある KSK(古い鍵)と登録した DNSKEY リソースレコード自身で署名された状態。

Ready

DNSKEY リソースレコードが公開されてから、置き換えられようとしている古い鍵(またはその鍵で作成された署名など)のコピーがリゾルバーのキャッシュから破棄されていることが保証される程度に十分に長い時間が経過した状態。

Active

ZSK の場合、鍵がリソースレコードセットへの署名で使用されており、鍵と作成した RRSIG リソースレコードの両方がゾーンに現れている状態。

KSK の場合、鍵に対応する DS リソースレコードが親ゾーンに存在し、DNSKEY リソースレコードセットの署名検証ができる状態。

Retired

ZSK の場合、鍵がリソースレコードセットの署名に使われなくなった状態。

KSK の場合、後継の DNSKEY リソースレコードと DS リソースレコードが存在している状態。

Dead

鍵はまだゾーン内に存在するが、その鍵で作成した署名や対応する DS リソースレコードの情報がリゾルバーのキャッシュ内から破棄されていることが保証される程度に十分に長い時間が経過した状態。

Removed

鍵が削除された状態。

Revoked

鍵が RFC 5011 方式でトラストアンカーとして設定されていることがわかっている場合に、DNSKEY に”失効(revoke)”ビットを設定して一定期間公開することで、その鍵をトラストアンカーとして使用しているリゾルバーにその鍵がゾーンから削除されようとしていることを通知している状態

図 8 で、ロールオーバー進行状況とともに状態が変化する様子の概略を示します(ZSK は事前公開法、KSK は二重 KSK 法の使用を前提としています)。各状態の期間の長さについては RFC 7583 を参照してください。特に KSK のロールオーバーの場合、Ready 状態には親側の DS 反映までの時間が含まれますので注意してください。

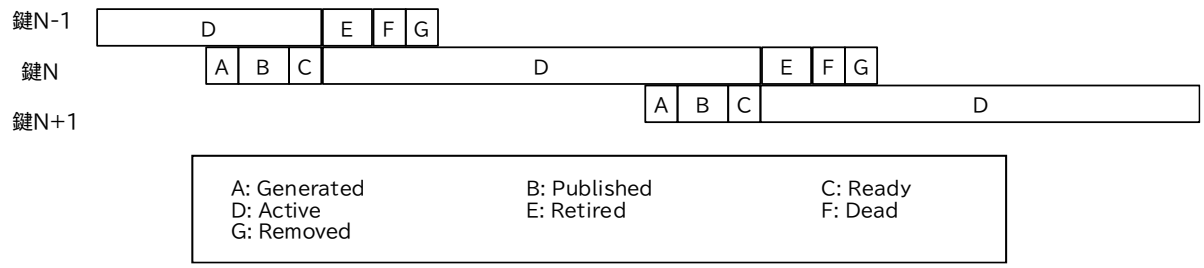

図 8 鍵のライフサイクル

出典:総務省(令和 5 年度 ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査の請負事業 DNSSEC ガイドライン作成チーム)

3.7 参考文献

[3-1] FIPS 140-3
https://doi.org/10.6028/NIST.FIPS.140-3

[3-2] RFC 7583: DNSSEC における鍵のロールオーバーのタイミングに関する考慮点
https://jprs.jp/tech/material/rfc/RFC7583-ja.txt

[3-3] DNSSEC レジストラ移転ガイドライン
https://dnssec.jp/wp-content/uploads/2010/11/20101122-techwg-registrar-transfer-guideline.pdf

第4章 ドメイン名登録・登録管理関係者

本章の主要な想定読者はドメイン名登録事業者、及びドメイン名登録者となります。gTLD ではレジストラが、.jp では指定事業者がドメイン名登録事業者の役割を務めています。4.1 ではドメイン名登録事業者が対応すべき内容について、4.2 ではドメイン名登録者が対応すべき内容について扱います。

DNSSEC の有効化において、ドメイン名登録事業者とドメイン名登録者は重要な役割を果たします。ドメイン名登録事業者はドメイン名登録者が DNSSEC を有効にするための機能を提供する必要があり、ドメイン名登録者は DNSSEC の有効化における最終的な責任者となります。

本章では、ドメイン名登録者が DNSSEC の有効化を選択出来るようにするために必要な情報についてまとめています。用語の補足として、本章に登場する「権威 DNS サーバー」はそのドメイン名の登録情報における「ネームサーバー（Name Server）」となります。

4.1 ドメイン名登録事業者

本節ではドメイン名登録事業者が対応すべき内容について取り扱います。なお、ドメイン名登録者がリセラ（再販事業者・取次事業者）を介してドメイン名を登録している場合、リセラにおける対応にも適用されます。

4.1.1 DS リソースレコードの取り次ぎ

4-1.	ドメイン名登録者が維持管理するドメイン名において DNSSEC を有効にするために、ドメイン名登録事業者はレジストリへの DS リソースレコードの取り次ぎ機能を登録管理業務として実装し、ドメイン名登録者に提供すべきです。レジストリへの DS リソースレコードの取り次ぎ機能は DNSSEC 対応における基礎となるため、ドメイン名登録事業者における実装・提供を特に強く推奨します （SHOULD）

ドメイン名登録者が維持管理するドメイン名において DNSSEC を有効にするために、ドメイン名登録事業者はレジストリへの DS リソースレコードの取り次ぎ機能を登録管理業務として実装し、ドメイン名登録者に提供すべきです。レジストリへの DS リソースレコードの取り次ぎ機能は DNSSEC 対応における基礎となるため、ドメイン名登録事業者における実装・提供を特に強く推奨します（SHOULD）。

DS リソースレコードの取り次ぎ機能を提供する事により、ドメイン名登録者は DNSSEC を適切に設定・管理することが出来ます。もし DS リソースレコードの取り次ぎ機能を提供していない場合は、次節の注意事項について考慮する必要があります。

4.1.2 ドメイン名の移管(指定事業者の変更)における注意事項

ドメイン名登録者はドメイン名の移管手続きを行うことでドメイン名登録事業者を変更出来ますが、ドメイン名登録事業者はドメイン名登録者が当該ドメイン名の DNSSEC が有効になっている状態で移管手続きをしていることに気づいていない可能性があることに注意してください。

DNSSEC が有効になっているドメイン名には親ゾーンであるレジストリに DNSSEC を有効にするための情報、具体的には第 3 章で説明した、DS リソースレコードの情報が登録・設定されています。この状態でネームサーバー情報のみを DNSSEC 非対応の権威 DNS サーバーに変更した場合、DNSSEC 署名検証をしているフルリゾルバーにおいて DNSSEC 検証エラーが発生し、名前解決に失敗する恐れがあります。

幾つかの著名な DNS サービスプロバイダーを兼ねているドメイン名登録事業者では DNSSEC に対応した権威 DNS サーバーを標準提供しており、それらのサービスを利用しているドメイン名はドメイン名登録者が意識することなく、DNSSEC が有効に設定されている場合があります。

移管先のドメイン名登録事業者が DS リソースレコードの取り次ぎ機能を提供している場合、ドメイン名登録者が DS リソースレコードを適切に管理(設定・削除)することが出来ます。しかし、移管先のドメイン名登録事業者が DS リソースレコードの取り次ぎ機能を提供していない場合、ドメイン名登録者が DS リソースレコードを管理することが出来ないため、移管先の顧客サポート部門で DS リソースレコードの取り次ぎに関するマニュアルオペレーションをサポートする必要があります。

4.2 ドメイン名登録者

本節ではドメイン名登録者が対応すべき内容について取り扱います。なお、ドメイン名登録者がドメイン名の管理業務について委託契約を結んでいる場合、その委託先においても同様の対応が必要になります。

4.2.1 DNSSEC を有効にする前に

4-2.	ドメイン名登録者は DNSSEC を有効にする前に、DS リソースレコードの取り次ぎ機能が提供されていること、権威 DNS サーバーが DNSSEC に対応していること、現時点において DNSSEC が設定されていないことの 3 点を確認しなければなりません（MUST）

ドメイン名登録者は DNSSEC を有効にする前に、次の点について確認しなければなりません（MUST）。

1. DS リソースレコードの取り次ぎ機能の提供有無の確認
   a） ドメイン名を管理するドメイン名登録事業者が DS リソースレコードの取り次ぎに対応していること
   b） DNS サービスプロバイダーを兼ねている登録事業者が DNSSEC 機能（自動的な取り次ぎを含む）を提供していること
      幾つかの著名な DNS サービスプロバイダーを兼ねているドメイン名登録事業者では DNSSEC に対応した権威 DNS サーバーを標準提供しており、このようなサービスでは自動的にドメイン名の DNSSEC が有効に設定される場合がありますので、契約しているサービスの内容について確認することをおすすめします。
2. 利用している権威 DNS サーバーが DNSSEC に対応していること
   権威 DNS サーバーが DNSSEC に対応していない場合、権威 DNS サーバーの DNSSEC 対応（3 章に記載）をドメイン名登録者自身が行うか、DNS サービスプロバイダー等の外部サービスに委託することが出来ます。
3. ドメイン名の DNSSEC が設定されていない状態であること

4-3.	ドメイン名登録者が DS リソースレコードの取り次ぎに対応していない（DNSSEC 非対応の）ドメイン名登録事業者と契約している場合、当該ドメイン名登録事業者が DNSSEC 対応されるまで DNSSEC の有効化を待つか、DNSSEC に対応しているドメイン名登録事業者にドメイン名を移管するなどの対応を選択することが出来ます（MAY）

ドメイン名登録者は DS リソースレコードの取り次ぎに対応していない（DNSSEC 非対応の）ドメイン名登録事業者と契約している場合、当該ドメイン名登録事業者が DNSSEC 対応するまで DNSSEC の有効化を待つか、DNSSEC に対応しているドメイン名登録事業者にドメイン名を移管するなどの対応を選択することが出来ます（MAY）。

ドメイン名を移管する場合は、後述の「4.2.5 ドメイン名の移管（レジストラ・指定事業者の変更）の注意事項」も合わせて確認するとよいでしょう。

4.2.2 DNSSEC の有効化

ドメイン名登録者が DNSSEC を有効にするためには、ドメイン名登録者自身（あるいは代理人）が DS リソースレコードの登録を申請するか、DNS サービスプロバイダー（ドメイン名登録事業者が兼業している場合を含む）等が提供する DNSSEC 機能を有効にする必要があります。

DS リソースレコードを登録申請する場合：
このケースは DS リソースレコードの登録をドメイン名登録者自身（あるいは代理人）で行う必要があることを示しています。利用中のドメイン名登録事業者が提供する権威 DNS サーバー以外の権威 DNS サーバーで DNSSEC 対応を行う場合は、利用中のgドメイン名登録事業者に DS リソースレコードの取り次ぎ機能が無い限り DNSSEC を有効にすることは出来ません。ドメイン名登録者は DNSSEC を有効にするために自身の、または契約先から指定された DS リソースレコードをドメイン名登録事業者が提供する UI から登録申請する必要があります。

DNSSEC 機能を有効にする場合：
このケースは DS リソースレコードの取り次ぎを使用せずに、契約しているサービスの DNSSEC 機能を有効にする必要があることを示しています。この場合、DS リソースレコードの取り次ぎがサービス内容に含まれているため、ドメイン名登録者による取り次ぎ申請は必要ありません（もし DS リソースレコードの取り次ぎがサービスに含まれていない場合、前述した「DS リソースレコードを登録申請する場合」を参照してください）。

4.2.3 鍵の管理

4-4.	DNSSEC の鍵情報について「3.4.4 鍵の管理に関する注記」で説明した通り、ドメイン名登録者が管理責任について確認しておくべきです（SHOULD）

DNSSEC の鍵情報について「3.4.4 鍵の管理に関する注記」で説明した通り、ドメイン名登録者は管理責任について確認しておくべきです（SHOULD）。
一般に、鍵情報のバックアップは権威 DNS サーバー運用者が行うと考えられますが、ゾーンファイルに設定する形で鍵情報を管理している場合、その責任はドメイン名登録者に帰属すると考えられるため、自身でのバックアップを強く推奨します。

4.2.4 権威 DNS サーバーの変更における注意事項

4-5.	ドメイン名登録者は DNSSEC 対応の権威 DNS サーバーを変更しようとする際、ドメイン名の DNSSEC 対応状況により、確認内容が異なることに注意しなければなりません

（MUST）

権威 DNS サーバー（ネームサーバー）の変更において注意すべき内容について説明します。
ドメイン名登録者は DNSSEC 対応の権威 DNS サーバーを変更しようとする際、ドメイン名の DNSSEC 対応状況に応じ、以下の三つに注意しなければなりません（MUST）。

1. 変更後の権威 DNS サーバーが変更前と同じ鍵情報を引き続き使用する場合
   権威 DNS サーバーの変更後も引き続き同じ鍵情報を使用する場合、署名情報を含む DNS データが同一であることを確認する必要があります。

2. 変更後の権威 DNS サーバーが異なる鍵情報を使用する場合
   権威 DNS サーバーを変更することで異なる鍵情報が使用される場合、ロールオーバーが発生した場合と同じ状況が発生します。
   詳細は「3.4.6 鍵のロールオーバーについて」を参照してください。

3. 変更後の権威 DNS サーバーが DNSSEC 非対応の場合
   ドメイン名の管理を外部組織に委託契約している場合等、ドメイン名登録者による変更では無いものについて、管理責任がどちらにあるのかを事前に契約約款などで確認しておくべきです。

いずれの場合においても権威 DNS サーバーを変更することで、DNSSEC 署名検証しているフルリゾルバーにおける名前解決に影響を及ぼす可能性があることを考慮する必要があります。

4.2.5 ドメイン名の移管（ドメイン名登録事業者の変更）における注意事項

ドメイン名の移管について「4.1.2 ドメイン名の移管（ドメイン名登録事業者の変更）における注意事項」と同様に移管手続きについて注意すべき内容について説明します。

4-6. ドメイン名登録者はドメイン名の移管手続きをしようとする場合、ドメイン名の DNSSEC の状態を事前に確認しなければなりません
（MUST）

ドメイン名登録者はドメイン名の移管手続きをしようとする場合、ドメイン名の DNSSEC の状態（有効または無効）を事前に確認しなければなりません（MUST）。
もしドメイン名の DNSSEC が有効な場合、次の内容に注意する必要があります。

1. DNSSEC 非対応のドメイン名登録事業者に移管しようとする場合
   ドメイン名の移管後に DNSSEC の有効化・無効化をドメイン名登録者が制御することが出来なくなるため、ドメイン名登録事業者における対応に時間を要する場合があります。ドメイン名登録者は移管手続きをしようとする前にそのドメイン名の DNSSEC を無効化することで、そうしたトラブルを回避することが出来ます。

2. ドメイン名の移管後に権威 DNS サーバーを変更しようとしている、または変更する可能性がある場合
   ドメイン名登録者は、DNSSEC 署名検証しているフルリゾルバーにおける名前解決に影響を及ぼす可能性があることを考慮する必要があります。前セクションの「4.2.4 権威 DNS サーバーの変更における注意事項」も参照してください。

ドメイン名登録者はドメイン名の移管の前後に DNSSEC の登録情報が意図した内容であることを確認することで、移管後に起こりうる DNSSEC のトラブルを回避出来る可能性が高まります。

付録 1. 本ガイドラインにおける要求項目と要求レベル

掲載章と対象者	#	項目	Must	Must not	Should	May
2 （第 2 章 フルリゾルバー）	2-1	公開しているフルリゾルバーは全て DNSSEC 対応にしなければいけません	○			
	2-2	例えば 3 つの異なる IP アドレスで 3 台のフルリゾルバーを運用している場合に、3 台中 2 台を DNSSEC 対応にして 1 台は未対応のまま残すという運用をしてはいけません		○		
	2-3	DNSSEC の署名情報を保持する RRSIG リソースレコードには、署名の有効期間に関する絶対時刻の情報が含まれます。フルリゾルバーに設定される時刻が合っていない場合、本来有効であるはずの署名を無効と扱う恐れがありますから、NTP 等の手段を使用してフルリゾルバーの時刻を信頼できる時刻ソースと同期しておかなければいけません	○			
	2-4	DNSSEC を有効にすると、問い合わせに対する応答パケットのサイズが大きくなります。今日では、UDP で IP フラグメンテーションを発生させることは推奨されていないので、EDNS のバッファーサイズを一般的な MTU 値を超えた値に設定してはいけません		○		
	2-5	EDNS のバッファーサイズは、リゾルバーソフトウェアのデフォルト値を使用すべき			○	
	2-6	DNS の名前解決失敗、DNSSEC の署名検証失敗の原因を詳細に伝える仕組みとして、RFC 8914 で拡張 DNS エラー（EDE: Extended DNS Errors）が規定されています。本ガイドライン執筆時点では主要なフルリゾルバーソフトウェアは対応済みですが、まだ実装から十分な時間を経ていないので、ある程度機能の安定動作が確認されてから導入を検討してもよい				○

掲載章と対象者	#	項目	Must	Must not	Should	May
	2-7	リゾルバーに設定として与えられるトラストアンカーには、最新のものを使用しなければいけません	○			
	2-8	トラストアンカーの更新は手動による作業でも行えますが、特別な理由がなければパッケージに付属しているトラストアンカーを使用すべき			○	
	2-9	DNSSEC 署名検証エラーがログ出力できるソフトウェア・アプライアンスの場合は、出力するように設定すべき			○	
	2-10	応答の RCODE ごとの問い合わせ数を取得して統計的に出力できるように設定すべき			○	
	2-11	署名検証の成功数、失敗数などのメトリクスが取得できる場合には、それらを取得すべき			○	
	2-12	通常のフルリゾルバーの監視項目に、時刻同期状況の確認、DNSSEC 署名された名前が解決できることの確認を加えなければいけません	○			
	2-13	統計情報を出力し、RCODE として SERVFAIL が返される数または割合の変化を調べる仕組みを構築すべき			○	
	2-14	フルリゾルバーの DNSSEC 対応は、まず最低限必要な条件を満たすことから始めて、段階的に高度化することを検討すべき			○	
	2-15	本ガイドラインが定めるレベル 1 での運用期間中に統計情報の取得を開始し、ある程度の期間問い合わせの傾向を見てからしきい値を決めるべきです			○	
	2-16	原因の究明は、一般にその情報展開（利用者、利用者に説明するサポート部門など）とあわせて考える必要があります。フルリゾルバーが扱うドメイン名の数は膨大なので、全てのドメイン名で生ずる障害情報を共有する必要はありません。その影響度に応じて情報展開をすべきです			○	
	2-17	IP エニーキャストや負荷分散技術を使用して IP アドレスを共有する物理			○	

掲載章と対象者	#	項目	Must	Must not	Should	May
		サーバーが複数ある場合に、それを区別する技術として RFC 5001 で NSID(ネームサーバー識別子)が定義されています。自分の AS から調査を行う際には、NSID を表示させるオプションを指定して(例えば dig/delv などを)実行し、応答したインスタンス名を把握しておくべきです				
	2-18	外部サービスそのものに障害が発生する可能性もあるので、自分の AS からも信頼の連鎖を調査できるようにしておくべき			○	
	2-19	フルリゾルバーが扱うドメイン名の影響度が大きい場合や、積極的な対応を取ってでも復旧をより早めたい場合は、障害が発生しているドメイン名の連絡先(jp ドメインの場合は Whois 検索で表示される「公開連絡窓口」か「技術連絡担当者」)にコンタクトをしてもよい				○
	2-20	影響度の大きいドメイン名で署名検証の問題が発生している場合、他の ISP でもその異常が観測され、原因の切り分けや修正依頼が既に行われている可能性もあります。日本国内は dnsops.jp[2-11]、海外の情報は dns-oarc[2-12]で交換されているので、常時アンテナを張っておくべきです			○	
	2-21	署名検証失敗の原因が攻撃ではないと判明し、復旧にある程度時間がかかることが見込まれる場合には、ネガティブトラストアンカーの設定を検討してもよい				○
	2-22	NTA の設定は運用への影響が大きいため、NTA を設定する基準または条件、誰が承認するのか、設定方法および解除方法などは事前に取り決めて手順化しておくべきです			○	
3 (第 3 章	3-1	権威サーバーに設定される時刻が合っていないと、本来有効であるはずの署名がリゾルバーに無効と扱われる	○			

掲載章と対象者	#	項目	Must	Must not	Should	May
権威 DNS）		恐れがありますから、NTP 等の手段を使用して権威 DNS サーバーの時刻を信頼できる時刻ソースと同期しておかなければいけません				
	3-2	ゾーン内のリソースレコードに署名を行うサーバー上には秘密鍵情報が存在する必要があります。したがって、サーバー上に存在する秘密鍵情報に対する何らかの保護手段を設定しなければいけません	○			
	3-3	DNSSEC Signer はその性質上、署名用の秘密鍵（ZSK の秘密鍵）にアクセスできる必要があります。秘密鍵が第三者に漏洩すると DNSSEC のセキュリティ機能が損なわれるため、不特定多数がアクセス可能な公開サーバー上で秘密鍵を保持するのは避けるべきです			○	
	3-4	KSK の秘密鍵は、原則はオフラインにて保管し、必要な場合にだけ DNSSEC Signer にロードして使用し、使用後は削除するという運用にすべきです			○	
	3-5	署名作業は、あるレコードに対して一度行ったら終わりではなく、署名の有効期間が終了する前に同じレコードに対して再度署名をし直す必要があります。手動による作業は間違いが起こりやすいので、特別な理由がない限り、DNSSEC に関連する作業には自動化の仕組みを導入すべきです			○	
	3-6	新規に DNSSEC を導入するのであれば、KSK/ZSK ともに暗号アルゴリズムとしてはまず ECDSA-P256-SHA256 を検討すべきです			○	
	3-7	サーバーソフトウェア/アプライアンスやロードバランサー、ファイアウォールが未対応の場合には、広く普及している RSA-SHA-256 でサービスを開始し、問題が解消した時点で ECDSA-P256-SHA256 への移行を検討してもよいです				○

掲載章と対象者	#	項目	Must	Must not	Should	May
	3-8	扱うゾーンの規模が大規模でない限りはサーバーソフトウェアやアプライアンスが提供するデフォルトの値を使用すべきです			○	
	3-9	DNSKEY リソースレコードの TTL は、フルリゾルバーにキャッシュされるため、極端に短い期間を指定することは避けるべきです			○	
	3-10	再署名までの間隔が短いと、署名有効期間を過ぎてしまい DNSSEC の署名検証に失敗する可能性が生じます。通常は終了時刻－TTL 値より前に再署名が実行されるべきです			○	
	3-11	署名アルゴリズムとして RSA よりも効率的な ECDSA が RFC 6781 の執筆時よりも普及していることから、ECDSA を採用した場合の使用期間はある程度柔軟に、より長い期間を設定してもよい				○
	3-12	KSK は CDS/CDNSKEY リソースレコードを使用した DS リソースレコードの更新方法が規定されていますが、本ガイドライン執筆時点では広く普及していないことから、運用者の手作業が発生せざるを得ません。その場合、二重 KSK 法（または二重署名法）を使用すべき			○	
	3-13	ゾーンウォーキングが問題になるのは TLD など大量の委任を持つ特殊なゾーンに限られますので、特別な理由がなければ NSEC を選択すべきです			○	
	3-14	NSEC3 リソースレコードを使用する場合、RFC 9276 の規定に従い、イテレーションは 0 を使用しなければいけません	○			
	3-15	NSEC3 リソースレコードを使用する場合、ソルトは使用してはいけません		○		
	3-16	権威 DNS サーバーに設定される時刻がずれると提供する署名の検証に失敗する可能性が高まるので、時刻同期状況は必ず確認しなければいけません	○			

掲載章と対象者	#	項目	Must	Must not	Should	May
	3-17	権威 DNS サーバーが提供するデータまでの信頼の連鎖が構築可能であることを確認しなければいけません	○			
	3-18	DNSSEC 検証エラーが発生するリスクを軽減するため、RRSIG リソースレコードの署名有効期間、特に終了時刻が想定よりも切迫していないかを監視すべきです			○	
	3-19	鍵のロールオーバーが進行中の場合、RFC 7583 を参照しながら想定されたとおりに作業が進行していることを確認すべきです			○	
	3-20	日常的なゾーン情報のロード・更新に関する情報や権威サーバーへのアクセスログに加え、自動化されたプロセス(署名の更新やロールオーバー)に関するエラー情報を確認するようにすべきです			○	
	3-21	DNSSEC 署名を維持したまま権威 DNS サーバー運用サービスを変更することを可能にする手段として、RFC 8901 で Multi-Signer DNSSEC モデルが規定されています。将来このモデルへの対応が必要となることがあるかもしれませんが、本ガイドライン執筆時点では、直ちに対応しなくてもよい				○
	3-22	ZSK のロールオーバーについては、セクション 3.3.4 で説明した DNSSEC ポリシーに従い、自動化ツールで処理を行うべきです。(セクション 4.2 に記述したようにロールオーバー中の監視はすべきです			○	
	3-23	DNSSEC の運用開始から KSK の第 1 回ロールオーバーまでには少なく見積もっても 1 年、ECDSA などの強力な暗号鍵を使用している場合にはもう少し後でもよい				○
	3-24	親ゾーンの DS リソースレコードと自ゾーンの DNSKEY リソースレコードを照合することになります。照合の際			○	

掲載章と対象者	#	項目	Must	Must not	Should	May
		には鍵タグの照合だけにとどめず、ダイジェスト値も照合すべき				
	3-25	DNSSEC の無効化には幾つかのリスクがありますから、緊急時の対応手段の最初に検討するものではありません。別の手段での対策をまず検討し、最終手段として無効化を選択するという位置づけにすべきです			○	
4（第 4 章ドメイン名登録・登録管理関係者）	4-1	ドメイン名登録者が維持管理するドメイン名において DNSSEC を有効にするために、ドメイン名登録事業者はレジストリの DS リソースレコードの取り次ぎ機能を提供すべきです			○	
	4-2	ドメイン名登録者は DNSSEC を有効にする前に、DS リソースレコードの取り次ぎ機能が提供されていること、権威 DNS サーバーが DNSSEC に対応していること、現時点において DNSSEC が設定されていないことの 3 点を確認しなければなりません	○			
	4-3	ドメイン名登録者が DS リソースレコードの取り次ぎに対応していない（DNSSEC 非対応の）ドメイン名登録事業者と契約している場合、当該ドメイン名登録事業者が DNSSEC 対応されるまで DNSSEC の有効化を待つか、DNSSEC に対応しているドメイン名登録事業者にドメイン名を移管するなどの対応を選択することが出来ます				○
	4-4	DNSSEC の鍵情報について「3.4.4 鍵の管理に関する注記」で説明した通り、ドメイン名登録者が管理責任について確認しておくべきです			○	
	4-5	ドメイン名登録者は DNSSEC 対応の権威 DNS サーバーを変更しようとする際、ドメイン名の DNSSEC 対応状況により、確認内容が異なることに注意しなければなりません	○			
	4-6	ドメイン名登録者はドメイン名の移管手続きをしようとする場合、ドメイン	○			

掲載章と対象者	#	項目	Must	Must not	Should	May
		名の DNSSEC の状態を事前に確認しなければなりません				

付録 2. 本ガイドラインの記述に関連する DNSSEC の RFC 一覧

本付録では、第 2 章、第 3 章でフルリゾルバーや権威 DNS サーバーを DNSSEC に対応させる際に、使用予定のソフトウェアやアプライアンスが準拠しているかを確認すべき RFC の一覧を示します。

これらは最小限の機能的要件です。ガイドライン本編では、これら以外にも DNSSEC の運用に関する RFC や、付加的な機能に関する RFC も参照されていますのでご注意ください。

【リゾルバー・権威サーバー共通】

・基本動作仕様・関連リソースレコード・基本暗号アルゴリズム

- RFC 4033, 4034, 4035

・ハッシュを使用する不在証明（NSEC3）

- RFC 5155

・暗号アルゴリズム・ハッシュ化アルゴリズムの追加

- RFC 4509, 5702
- RFC 6605

【リゾルバー固有】

・トラストアンカーの自動更新

- RFC 5011

・ネガティブトラストアンカー

- RFC 7646

おわりに

本ガイドライン案は、令和4年度総務省事業「ISP におけるネットワークセキュリティ技術の導入に関する調査」及び令和5年度総務省事業「ISP におけるネットワークセキュリティ技術の導入及び普及促進に関する調査」の結果として、同実証事業へ参加した実証事業者の意見・有識者検討会メンバーの意見を基に作成されました。

本ガイドライン案作成には同実証事業の有識者検討会メンバーである、日本 DNS オペレーターズグループ/株式会社 JPIX 石田慶樹氏、日本 DNS オペレーターズグループ/株式会社インターネットイニシアティブ 其田学氏、東京大学 関谷勇司氏、日本 DNS オペレーターズグループ/NTT コミュニケーションズ株式会社 高田美紀氏、GMO インターネットグループ株式会社（2024 年 2 月末日まで） 永井祐弥氏、フィッシング対策協議会/トレンドマイクロ株式会社 野々下幸治氏、株式会社日本レジストリサービス（2023 年 9 月末日まで）米谷嘉朗氏の各氏による検討の結果、メンバーの合意を得ました。

また、本ガイドライン案に関連する成果物として、鍵のロールオーバーを行う際のタイミングに関する考慮事項を記述した RFC 7583 の和訳「DNSSEC における鍵のロールオーバーのタイミングに関する考慮点」を株式会社日本レジストリサービスの「DNS 関連技術情報」ページで公開しました。

謝辞

本ガイドライン案執筆にあたりご協力いただいた、藤野貴之氏、日本 DNS オペレーターズグループ 中島智広氏、株式会社日本レジストリサービス 栫邦雄氏、森下泰宏氏へここに感謝の意を表します。

また、RFC 7583 の和訳の掲載にご協力いただいた株式会社日本レジストリサービスに感謝の意を表します。

令和 6 年 7 月 8 日時点

DNSSEC ガイドライン専門家チーム

- 石田 慶樹（日本 DNS オペレーターズグループ/株式会社 JPIX）
- 岡田 雅之（長崎県立大学）
- 栫 邦雄（株式会社日本レジストリサービス）
- 末松 慶文（日本 DNS オペレーターズグループ/ NTT ドコモビジネス株式会社 ）
- 関谷 勇司（東京大学）
- 其田 学（日本 DNS オペレーターズグループ/株式会社インターネットイニシアティブ）
- 高田 美紀（日本 DNS オペレーターズグループ/ NTT ドコモビジネス株式会社 ）
- 水野 稔晴（一般社団法人日本ネットワークインフォメーションセンター）