RFC5863
Identified Mail (DKIM)
Development, Deployment, and Operations
概要

概要

- RFC5617

  DomainKeys Identified Mail (DKIM) Author Domain Signing Practices (ADSP)

- ADSP とは

  DKIMの認証結果をどのように扱うべきかを、メール送信者がメール受信者に対して公開する枠組み。受信側はその方針をチェックすることができる（ADSP check）

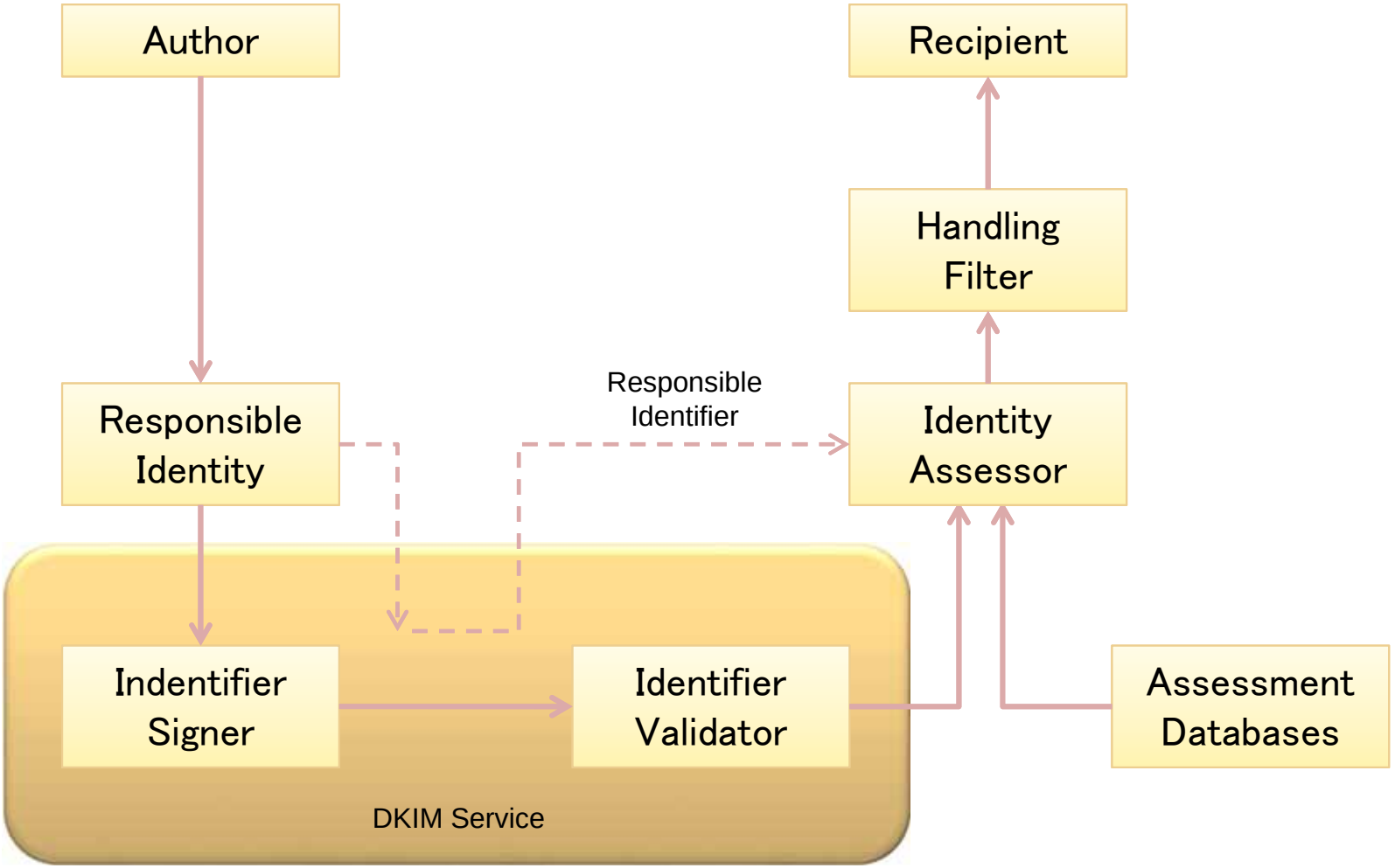
章立て

1. 概要
2. 信頼評価の一機能のとしてのDKIM利用
3. DKIM Key作成・保管・管理
4. 署名
5. 検証
6. 署名の分類
7. 使用例
8. 利用時の懸念事項
9. セキュリティ懸念事項
10. 応答
11. 参照

変更履歴(メモ)

- 2.1 電子メールにおけるメールの信頼評価について

- 2.1 電子メールにおけるメールの信頼評価について



- 2.2 Assessment Identifier のためのDKIM Tagの選択
  - メッセージのSignerは正確なデータの提供と、データがAssessorへ配達されるという事を理解する必要がある
  - 上記が曖昧であると受信者がどの情報を選択し評価しているかを、送信側で認識できなくなる
  - DKIMには混同されやすい3つのパラメータがあるが、1つは正式な入出力を定義し、残り2つは監査とデバックで用いられるものである
  - DKIM-Signatureヘッダーにはs=,d=,i=のパラメータがある

- d=は必須
- s=とi=はOption設定可能
- それぞれの説明

- d=は必須
- s=とi=はOption設定可能
- それぞれの説明

署名するDomain Nameの選択

dkim.jp

- 署名するドメインの単位と、
  その仕分け方法について

受信者側での評価について

- DKIMはRFC 5322のFromフィールドの
  ドメインの正しさを評価するものである
- DKIMの評価結果に応じて、どのように処理するかは
  Assessorが提供可能である

- Stream Risk vs Organizational Trust のeach combinationごとのハンドリング例
- 状態の説明

- 概要
  - DKIMは公開鍵方式を基に、PKIで重要となる鍵の配備と、鍵の認証はDNSを介して実装される
- 署名者が必要なインターフェイスは以下
  - 秘密鍵によって署名されたメッセージに対するDNSレコードを管理できる能力
  - 特定のキーの使用はセレクタによって認識された特定のメッセージを制限する
  - DKIMの鍵レコードを持つドメインの正規所持者は署名されたメッセージに対しての責任を負う

- 3.1 秘密鍵の管理
  - 秘密鍵は可能な限り少ない人数で管理する必要があり、管理者の異動があった場合には管理されていた秘密鍵は置き換える必要がある
  - 鍵は厳重に管理し、定期的な監査が必要である
  - 秘密鍵は一つの場所にあるべきであり、使用されなくなった場合は削除されるべきである
  - 秘密鍵は可能な限り閉じられた環境で作成されるべきである
  - 説明責任と監査のために
    - 署名鍵は単一の名称で管理する必要がある
    - 複数の署名者が署名する際には、個々に異なる鍵を用いる必要がある

- 3.2公開鍵の管理
  - DKIMの使用のため、DNS管理者は以下の2つが必要とされる
    - DNSにDKIMのためのレコードを作成する
    - 偽のDNSレコードの挿入など悪意のあるユーザから守るためのセキュリティを提供する
  - セキュリティを確保するためにDNSSEC[RFC4034]も使用できる

- 3.2.1セレクタの割り当て
  - セレクタは管理者のニーズによって割り当てが可能である
    - 新しい鍵の展開する
    - 他の外部委託先などに一定期間委任したい場合

- 3.3 ユーザ個別管理の問題点
  - 外部の検証者が追加情報なしに、証明書の細部を検証できない
  - ユーザの鍵数が多い場合、検証者によるローカルのキャッシュ効率が減少する
  - ユーザ数が多い場合、個々の端末上の秘密鍵の管理が疎かになる

- 3.4 第三者署名の鍵とセレクタの管理
  - 大量配信などを行うサービスにおいては、第三者が署名鍵を保有します
  - 第三者署名者の署名鍵は、ドメイン管理者や他の第三者署名者とは異なる署名鍵とします
  - 第三者署名者の権限を最小限とする事で、セキュリティリスクを下げます

- 3.5 鍵ペアとセレクタのライフサイクル管理
  - 実装者は非対称鍵のペアのライフサイクル管理を確立し、ドキュメント化、そのプロセスを監視する必要がある

- 3.5.1 キーの配備プロセス例
  1. 鍵ペアの生成
  2. 必要とされるセレクタを生成し、DNS管理者へ依頼する
  3. DNS管理者はセレクタの登録リクエストを検証し、問題なければ以下を実施する
     1) セレクタを割り当てる
     2) 対応する鍵をDNS上に配備する
     3) DNSのアップデートを待つ（必要であれば）
     4) 署名する機器に割り当てられたセレクタを報告する
  4. 署名者は鍵レコードの登録の正当性を検証する
  5. 署名者は新しい鍵ペアを使用して署名の作成を開始する
  6. 署名者は使用されなくなった秘密鍵の公開を中止する

- 3.5.2 鍵の失効プロセス例

1. 署名者は署名作成のための秘密鍵の使用を中止する
2. 署名者は署名する機器上のメモリのコピーを含めた秘密鍵の全てのレコードを削除する
3. 署名者はDNS管理者に署名する鍵及び対応する鍵レコードがある特定の日付で使用されなくなったことを通知する
4. DNS管理者はセレクタの中止リクエストを検証する。そのリクエストに問題なければ以下を実施する
   1） セレクタの削除がスケジュールされる
   2） 削除日時まで待つ
   3） 署名者は空の公開鍵を含めたセレクタの撤回を発行するか、セレクタの削除を行う
5. 検証者にとって、空の公開鍵を伴うセレクタの検証と、セレクタがない状態の検証の機能的な違いはないが、唯一の違いとして、空の公開鍵は明示的にその公開鍵が取り消されたことを意味し、間違って異なった公開鍵が再利用されないという意味を持たせることができる

- 概要
  - DKIMの証明書はアウトバウンドメールサービスで以下の2つをサポートする
    - DNS管理インターフェースは名前にアンダースコアを含むDNS名とリソースレコードを作成でき、維持することができる
    - 署名生成モジュールはアウトバウンドメールサービスで呼び出され、メッセージヘッダにDKIM-Signature:を加える
- 4.1 DNSレコード
  - 受信者はDKIMの署名を検証するために、メッセージの署名に関連付けられた公開鍵を取得する
  - DNSレコードは公開鍵情報を提供するDKIM RRを含むことが必要である

- 4.2 署名するモジュール
  - 署名するモジュールは信頼される組織内で配備することができる
  - 例えば、部門レベルのエージェント、アウトバウンド境界にあるMTAや、場合によってはMUAでも配備可能である
  - その選択はソフトウェア開発、管理者の負担、セキュリティリスクなどに依存する

- 4.3 署名するポリシーとプラクティス
  - すべての組織において、どのようなメッセージにどのようなドメイン／鍵／セレクターを用いて署名するのかポリシーを持つべきである
  - メッセージは、送信者・コンテンツなど様々であるため、ポリシーは時間の経過によって変更すべきである。

- 概要
  - DKIMにおける“Verifying”（検証）とは、メッセージの認証(authentication)の機能となる。
  - 承認(authorization)機能はない。Authorization機能は他のreputation情報と合わせて利用することが必要
- 5.1 利用のスコープ
  - DKIM署名に失敗した場合の扱い
  - 不正な署名が添付されているメールは、全く署名の無いメールよりも『可もなく不可もない(no better and no worse)』メールだと扱うべき。
- 5.2 署名のスコープ
  - 署名範囲での確認を実装する必要がある
  - 例）l= (length)オプションがある場合、指定のサイズのメッセージを署名の範囲とする

- 5.3 デザイン上のスコープ
  - DKIMの認証が失敗する可能性がありえることも考慮する必要がある。例えば下記の状況の場合。
    - 正規のprivate key所有者による秘密鍵管理が不十分である
    - 正規のドメインオーナーのDNSゾーンレコード管理が不十分である
    - キー情報が正しいDNSサーバから取得できなかった場合
    - DNSのゾーン所有者が変わった場合（つまり、管理ができていない）
  - DKIMのキー取得は高信頼性の実現よりは実装のしやすさを重視している。したがって、高信頼のキー管理＆アクセスを必要とする場合、DNS以外の実装も考慮する必要がある。
- 5.4 Inbound mail filteringでの扱い
  - 信頼のあるソースからの正しいDKIM署名がついたメッセージはホワイトリスト化することによって、リソースを多く使うコンテントフィルタの処理を省くことが可能
  - ただし、DKIMの規定にはReputation判定方法に関しては触れていない
  - DKIM署名の確認機能を持たないAdaptive(学習型の)スパムフィルターはDKIM署名を無視する設定にするのがよい（なぜ？）

- 5.5 メーリングリスト(ML)等の中継メッセージでの扱い
  - MLでの中継におけるDKIM実装は要注意。
  - 受信者（Verifyer)が転送された（＝修正された）メッセージの確認を正しく行えるように 中継者の処理を確認するべくDKIM署名（“verifiable claim of responsibility”）の実装が必要
- 5.6 検証結果の作成、送信、利用の扱い
  - 署名の検証機能と署名の利用は独立させた方がよい。その理由は下記の通り
    - DKIM署名を利用するアプリケーションはDKIMの署名検証機能がない可能性がある
    - DKIM署名検証が終わった後にメッセージが変更される可能性がある（転送時等？）
    - メッセージリード時に署名検証の為のキーにアクセスできない場合がある
  - （独立性を持つものの）検証と利用の間において誤ったResultsヘッダが挿入されないよう十分にセキュアにする必要がある。
  - Resultsヘッダを付与するIntermidiary（中継者）は、不正なResultsヘッダが挿入された場合、利用者が区別できるようにしなくてはならない。
    - 中継Verifierからの転送メールにおける扱いに関してはRFC5451を参照のこと

6. 署名の分類

	署名の種類	意味	例
①	シングルドメイン署名	送信メールのドメインがすでに親ドメインになっている場合	Mail Fromのドメインがcompany A.comになっている
②	ペアレントドメイン署名	送信メールのドメインがサブドメイン単位であっても親ドメインの署名を使う	Marketing.domain.example, sales.domain.exampleでも全てd=domain.example とする
③	第三者署名	メールの送信者がDKIMの実装ができず、署名者のドメインがMail Fromのドメインと異なる場合。	ISP/ESPなどのプロバイダからの送信の場合 Reputation Provider
④	信頼できる第三者署名	署名機能と鍵管理が信頼できる第三者に委託されて送信される場合 サブドメインのみを第三者委託してDKIM署名をつける場合は、該当するサブドメインのキーペアを委託することになる. DNS Delegationと合わせて利用することも考えられる	DKIM署名システムを自社で持てない場合
⑤	複数署名	一つのメッセージに複数のDKIM署名が含まれるケース	①企業でサブドメインとペアレントドメインの両方の署名をつける場合 ②サービスプロバイダがプロバイダと個々のユーザの両方の署名をつける場合 ③Forwarderの場合メッセージの正しさを保証するために、オリジナルと合わせてForwarderの署名をつける ④Reputation Providerが『お墨付き』として署名することも含む

- 概要
  - メッセージタイプ、署名者の位置づけなど様々な状況によって異なる署名が作成される可能性がある。このセクションではDKIM配備に向けて幾つかの利用例と懸念事項を記述する

- 7.1 ドメイン単位での署名
  - すべてのメッセージでd= を同一とする署名を行う方法。これが最も一般的な利用。

- 7.2 同一ドメイン内におけるメールタイプ毎の署名
  - 送信されるメールタイプ毎にサブドメインを分け、サブドメインごとに個別の署名を行う方法

- 7.3 ADSP
- 7.3.1 概要
  - 送信されるすべてのメールに署名を行う場合、受信者はより積極的にドメイン情報をフィルタリングに利用する事ができる。詳細はADSP(RFC 5617)にて解説
  - すべての正規メールを制御できない可能性を考慮すると、ADSPはすべての利用者に推奨されるものではない
  - ADSPを採用する際には、配送不能よりもフィッシング攻撃のリスクに重きをおいて、慎重に検討を行うべきである
- 7.3.2 定義
  - ADSPではRFC 5322 From AddressをAuthor Address、そのアドレスのドメインをAuthor Domainとして扱う
  - SDIDとAuthor Domainが一致したものをAuthor Signatureとして扱う
  - DKIMではAuthor Domainとヘッダーのドメインには関連性はなく、またAuthor Domain以外の署名は無効とみなされる

- 7.3 ADSP
- 7.3.3 ADSP利用例
  - dkim=all/dkim=discardableの設定を行う場合、受信側で検証に失敗すると配信されない可能性があるため、開始時に入念に確認が必要である
  - dkim=all/discardableの場合、Author DomainとSDIDが一致が必須である
    例えばcompanyA.exampleドメインからmarketing.companyA.exampleで署名されたメールは配信されない、もしくは破棄される事を理解する必要がある。
  - ADSP による制御は、純粋なDKIMによる検証よりも厳しく制御される

- 7.4 委任署名
  - 複数のドメインの送信を代行している送信事業者などから送信する場合を想定
    送信事業者は企業のドメイン(またはサブドメイン)を利用して署名する

- 7.5 第三者のサービスプロバイダー
  - Email Service Provider(ESP)がメールサービスを提供する際、契約している
    Client自身の署名ではなくESPの署名をつける
  - Client毎に認識を分ける方法としては以下の手法がある
    - SDIDは共有し、AUIDで認識させる
    - SDIDのサブドメインで認識させる

- 7.6 Reputationに基づく署名
  - 送信者の振る舞いごとに署名を選択する方法
    評価にもとづいてSDIDを選択し、ADIDに送信者の情報を用いる

- 7.7 Agentや仲介者による署名
  - MLや知人への転送メールなど、発信元の責任範囲で無い場合を想定
    転送者自身の署名をつける必要がある

- 8.1 標準的でない送信方法による問題
  - DKIMはプライベートキーへのアクセスが制限されているという事で信頼性を担保しているが、以下の例ではその制限のために署名を行う事ができないことに留意が必要である。
    1. 複数のISPのアドレスを持ち、ISP AのアカウントからISP BのアドレスをFromアドレス(RFC5322)に用いて送信する場合
    2. 記事を知人に転送するサービスなどを利用する場合
       個人のアドレスを用いて転送する事が可能であるが、署名を行う事ができない
    3. アカウントAからBへ転送設定を行なっている場合

- 8.2 内部メールの保護
  - 一つの組織内のメールは一般的には使用できるものである。ただし、社内メールが組織内で配送されたものであるというわけではない。DKIMはこのようなケースを救済する手段になる。すべての送信メールに署名を行う場合、有効な署名がついていないものを見分ける必要がある。
  - DKIMは、組織内のメールが詐称されていない事を判断するために利用できるが、MLや転送メールのように、DKIMが無効になる可能性があることを検討しなければならない。

- 8.3 署名の粒度
  - どの程度の粒度で署名を作成するか検討が必要である
    - ドメインで署名を共有し、i=をユーザ単位にする
    - ユーザ単位に個々の署名を作成する
    - d=をユーザ単位に設定する
  - 複数のドメインに対し、同一の署名設定は可能だが、推奨ではない
    - 一つのドメインに対して異なるセレクターを用いて署名をする事は技術的に可能
    - しかしセレクターは鍵管理が目的であり、受信者次第でユニークとして扱われない可能性があるので推奨しない
  - 多くの場合ユーザ単位の署名は非現実的である
    - 受信側がDKIM署名を検証する場合、より簡易な方法を取る事が多い
    - 受信側はユーザ単位の評価では無く、ドメイン単位の評価に興味を持っている
    - パフォーマンス観点から、ユーザ単位の個々の署名を行うより、i=タグで関連付ける事の方が効果的である

- 8.4 Email基盤エージェント
  - DKIMの実装は、境界のMTAに実装される事を想定している
  - 以下はemail基盤への推奨である

  - Outband
    - MSA(Mail Submission Agent)、または境界のMTAは送信ポリシーに従っている事の確認が必要である
    - DKIM送信ポリシーに従っていない場合には、運用者へのアラートを送信できるようにする事が必要である
    - MSAはMUAが署名をつけるケースがあることを考慮する必要がある。このような場合には、(MSAは)署名を変更しないような考慮が必要である。
    - 通常MUAは脆弱であり、Reputationの低下につながりかねないため
      MUAに署名機能を持たせるべきでは無い。
  - Inbound
    - DKIM配備の際、他のmail基盤において、DKIMの役割以外で、確認を妨げるような変更が行われない事を確認する必要がある
    - inbound MTA、MDAは認証結果をRFC5451のAuhentication-Results Headerとして埋め込むことが可能である

- 8.4 Email基盤エージェント
  - Intermediaries
    - inbound/outboundMTAを指し、それぞれの機能に沿う必要がある。
    - Intermediariesがメッセージを変更すると署名を破壊する可能性がある
    - Intermediariesは
      - SpamFilterをInboundに配備する必要があり
      - 壊れたすべての署名を削除する必要がある
        また
      - 変更前に評価する事ができ
      - その結果を埋め込むことができる
      - 評価結果を含んだ署名を行うことができる

- 8.5 MUA
  - DKIMは境界のMTA等で利用される事を想定しているが、MUAがDKIMの署名と認証をする事も可能である
  - Outbound
    - MSA経由で送信する場合であっても、MUAが署名する事が可能である
    - MUAであっても直接送信される場合には送信MTAとして考えられる
    - (8.4項のoutbound MTAにおける注意事項の実装検討も必要である)
    - たとえMUAであっても、ADMDに直接送信するMUAの場合は、MTAとしての扱いとする必要がある
  - Inbound
    - MUAは、受信時に経由するMTA／MDAの検証結果を信頼する事ができる、すなわちAuthentication-Results:ヘッダフィールドを直接参照して、検証してよい
    - MTAによりメッセージ変更が発生して、署名検証が失敗場合には、署名無しと同じ扱いとしてよい
    - DKIM検証結果が署名無しのメール実際のユーザの認識以下の評価であった場
    - MUAはメッセージがinbound MTAで変更される事を許す必要がある