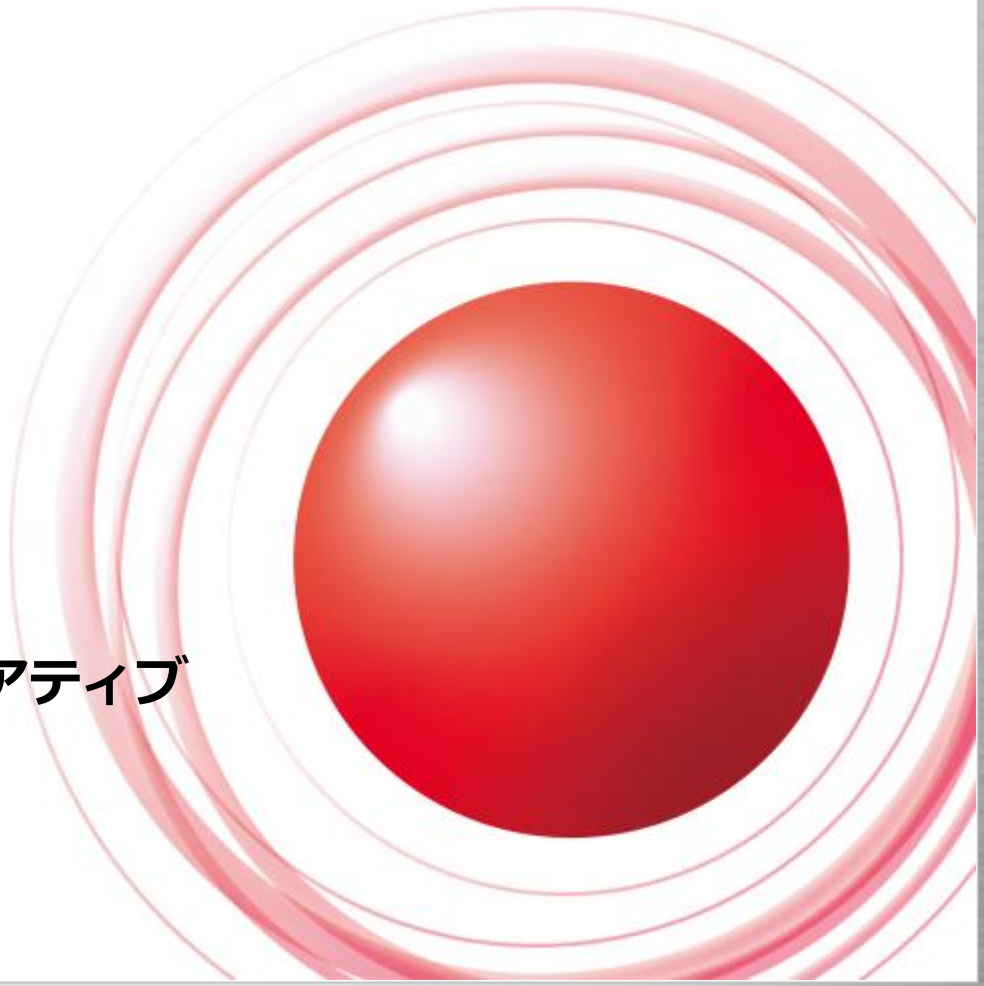
送信ドメイン認証 導入指南 2018

IIJ
Internet Initiative Japan



株式会社 インターネットイニシアティブ
鈴木 高彦

Ongoing Innovation

背景

送信ドメイン認証に対応する目的

- **メールを受け取ってもらう**
  - 送信ドメイン認証に対応していないとメールを受け取ってもらえない例は珍しくない
- **ドメインを悪者から守る**
  - 悪者は守りの手薄なドメインから攻撃を仕掛ける

送信ドメイン認証とは

△ 受信者が spam を見分けるための技術

◎ 送信者がドメインを守るための技術

何を守るか

- **ヘッダ From のドメインを第三者の利用から守る**

From: Takahiko Suzuki <takahiko@iij.ad.jp>

  - 全ての MUA・Webmail で差出人として表示される
  - もちろんエンベロープ From も認証できた方がいい

送信側の導入

「正当な送信」の把握

- **保護すべき「正当な送信」を洗い出す**
  - 色々な部門
  - 色々な方法
  - 色々なドメイン
  - これらの追加・変更
- **全てを把握するのは容易ではない**

最初の難関: メールの出口の洗い出し

- **Google がドメインの所有者にメールの受信状況を毎日レポートしてくれればいいのに…**

DMARC aggregate report

DMARC aggregate report

```
<record>
  <row>
    <source_ip>198.51.100.199</source_ip>
    <count>4</count>
    …
  </row>
  <identifiers>
    <header_from>foo.example.com</header_from>
  </identifiers>
  <auth_results>
    <spf>
      <domain>foo.example.com</domain>
      <result>fail</result>
    </spf>
  </auth_results>
</record>
```

DMARC aggregate report

- **送信ドメイン認証の結果をドメイン管理者にフィードバックする**
  - 送信元 IP アドレス
  - (レポートを出してくれているサービスが受け取った) 通数
  - 送信ドメイン認証評価結果
  - 大抵1日1回
- **主要な Mailbox Provider がレポートを提供**
  - Google, Yahoo! (US), AOL, Microsoft, Facebook, LinkedIn, XS4ALL, Mail.Ru, …

DMARC aggregate report

- **どんな送信を把握できるか**
  - 正常な送信
  - 把握している出口での設定ミスなどによる認証失敗
  - 把握していない出口からの送信
    - 把握していない送信サービス、自宅からの送信など
  - **悪意ある第三者による送信**
    - 導入後のドメインの不正利用の監視にも有用
- **様々な情報が見えるので関係者に話を通すこと**
- **XML 形式なので慣れないと読みづらい**
  - DMARC レポートの可視化サービスも選択肢
    - 重要なデータを預けるので選択は慎重に

DMARC レコード

- **DMARC aggregate report をリクエストするためには DMARC レコードを宣言する**
  - DNS TXT レコードを所定の形式で宣言するもの

```
_dmarc.example.com. IN TXT "v=DMARC1; p=none;
                            rua=mailto:dmarc@example.com"
```

  - この時点では必ず “p=none” にしておくこと
  - レポート送信先の宣言の “mailto:” を忘れやすいので注意
  - レポート送信先は複数指定可能
  - レポートの送信先が外部ドメインの場合は追加の宣言が必要

```
_dmarc.example.com. IN TXT "v=DMARC1; p=none;
                            rua=mailto:dmarc@example.net"
example.com._report._dmarc.example.net. IN TXT "v=DMARC1"
```

  - 特殊な要件がない限り同じドメインで受け取るのがオススメ

DMARC レコード

- **サブドメインすら把握できていない場合は組織ドメインに書けば認識してくれる**
  - Public Suffix の下のレベルが組織ドメイン
    - https://publicsuffix.org/
      - .com, .co.jp, .jp, .日本, …
    - abc.def.example.com なら example.com が組織ドメイン
  - From: username@abc.def.ghi.example.com の場合、以下の順に DMARC レコードを探索する:
    1. _dmarc.abc.def.ghi.example.com
    2. _dmarc.example.com
  - example.com に DMARC レコードを書いておけば配下のサブドメインは全てカバーできる
    - **無効にする手段はない**
    - **任意の階層のサブドメインに対応しているわけではない**

DMARC

- **Domain-based Message Authentication, Reporting, and Conformance**
  - RFC7489
- **大きく2つの機能に分けられる**
  - ポリシーの宣言
  - レポートの送信

DMARC ポリシー

- **認証失敗したメールの取り扱い方について、ドメイン所有者の期待を宣言**
  - “none” (何もしない)
  - “quarantine” (隔離)
  - **“reject” (拒絶)**
    - 「うちのドメインを名乗る (=ヘッダ From に持つ)、認証できないメールは拒絶して欲しい」
- **メールの配送を、送信者ではなく、ドメイン所有者の意向を踏まえて決定する**
  - 受信者のポリシーにもよるので、必ず従われるとは限らない

DMARC “reject” ポリシー

- **“reject” はかなり強力なポリシーだが、宣言する側にもそれだけの理由 (と覚悟) がある**
  - メールが届かないことよりも悪用されるリスクの方が大きい
  - ドメインがフィッシングのターゲットにされやすい
  - 金融機関・行政機関など
- **受信側はそのまま受け入れることを推奨**
  - Google, Yahoo! (US), AOL, Comcast なども DMARC ポリシーを尊重して拒否 (一部例外あり)
  - 国内 ISP での導入はあまり進んでいない
- **“reject” ポリシーを宣言すると、本当に reject される状況にあると思って扱ってよい**

DMARC “reject” ポリシー

- **“reject” を目指すべきか？**
  - “reject” ポリシーを宣言するハードルは低くない
    - ドメインを名乗る全てのメールを完全に把握する
      - 狙われやすい大きな組織ほど準備に時間がかかる
      - **aggregate report は準備の大きな助けになる**
    - ML に投稿できない
      - MLを通ると SPF/DKIM が fail するので reject される
      - 顧客への連絡に使用するメールと従業員が使うメールのドメインを分離するなどの対策
  - **フィッシング対策としての効果は絶大**
    - PayPal: 2013 年にフィッシングの報告が 70% 減少
    - Twitter: 1.1億通/日の詐称メールが数千通/日に減少
    - https://www.agari.com/dmarc-numbers/

DMARC “reject” ポリシー

- **Yahoo!, AOL の “reject” ポリシー導入事件**
  - 2014年4月、”yahoo.com” が悪者に多用されるため、クレームに耐えかねて急遽 “reject” ポリシーを導入
  - ML 経由したメールが一斉に reject されて大混乱
  - 悪者は悪用できなくなった “yahoo.com” に**見切りをつけて** “aol.com” に移る
  - 2週間後、急増したクレームに耐えかねて “aol.com” も “reject” ポリシーを導入
  - 両社とも “reject” ポリシーの副作用についてはよく理解していたが、副作用にこだわっているような状況ではなかった
- **悪者は DMARC を導入していないドメインを渡り歩く**
  - 今被害に遭っていなくても準備は進めておく

DMARC “none” ポリシー

- **“reject” ポリシーは万人向けではない**
- **“none” ポリシーでも書いた方がいい**
  - **とりあえず “none” を書いても副作用がない**
    - 国内携帯キャリアや多くの ISP も宣言済み
  - **aggregate report を受け取れる**
  - 受け取り手による SPF・DKIM の解釈のブレがない
  - alignment (ドメインの一致) に relaxed mode を使うと SPF や DKIM による送信ドメイン認証対応がちょっとラクになる

DMARC

- **アメリカ国土安全保障省 (DHS) 指示 BOD 18-01**
  - 発行: 2017年10月16日
  - 対象: All Federal Executive Branch Departments and Agencies
    - ざっくり .gov ドメイン
  - 原文: https://cyber.dhs.gov/assets/report/bod-18-01.pdf
- **DMARC ポリシーに関して**
  - 90日以内に “p=none” を宣言すること
  - 1年以内に “p=reject” を宣言すること

送信メールを DMARC に pass させる

- “spf=pass” + alignment
- “dkim=pass” + alignment

SPF 概要

- SMTP エンベロープFrom を接続元 IP アドレスで認証
- ドメイン管理者は正当な IP アドレスを DNS TXT レコードを使って宣言
- 転送・ML には対応不可

```
C: EHLO outbound-mta.iij.ad.jp
S: 250 …
C: MAIL FROM:<takahiko@iij.ad.jp>
S: 250 …
C: RCPT TO:<someone@example.com>
…
```

```
iij.ad.jp. IN TXT "v=spf1 ip4:192.0.2.123
                   ip4:198.51.100.234
                   include:thirdparty.example.com -all"
```

SPF レコードの書き方

- **原則として “ip4”, “ip6” のみを使う**
- **メール送信を代行するサービスを利用する場合、そのサービスから提供される SPF レコードを “include” で参照する**
  - “include” は1回の評価で10個まで
- **最後は “-all”**
  - “~all” と扱われ方はあまり変わらない

```
example.jp. IN TXT "v=spf1 ip4:192.0.2.123
                           ip4:198.51.100.128/30
                           include:thirdparty.example.com -all"
```

- M^{3}AAWG Best Practices for Managing SPF Records
  - https://www.m3aawg.org/Managing-SPF-Records

SPF 経由で DMARC に対応させる

- **SPF の場合に要求される alignment**
  - strict mode: エンベロープ From のドメイン = ヘッダ From のドメイン
  - relaxed mode:エンベロープ From の組織ドメイン = ヘッダ From の組織ドメイン
  - 基本的には strict mode で実現できるように頑張る
- **エンベロープ From とヘッダ From の一致を満たせないケースも**
  - エンベロープ From をバウンスの回収に使う送信システム
  - DKIM で対応するしかない

DKIM 概要

- **ヘッダおよび本文に電子署名を施し、公開鍵で認証**
  - ヘッダ From を認証できる
- **公開鍵はドメインの DNS レコードにぶら下げる**
- **転送には対応可、ML には対応不可**

```
DKIM-Signature: v=1;a=rsa-sha256;c=relaxed/simple;
    h=To:From: Subject (略);d=iij.ad.jp;s=omgo2;t=1510031431;
    x=1511241031;bh=pvZ1fKe/ (略);b=UqcV8lw4(略);
From: Takahiko Suzuki <takahiko@iij.ad.jp>
```

```
omgo2._domainkey.iij.ad.jp. IN TXT “v=DKIM1; k=rsa; p=MIIBIjAN(略)"
```

DKIM 作成者署名と第三者署名

- **作成者署名 (Author Signature)**

```
DKIM-Signature: v=1;a=rsa-sha256;c=relaxed/simple;
    h=To:From: Subject (略);d=iij.ad.jp;s=omgo2;t=1510031431;
    x=1511241031;bh=pvZ1fKe/ (略);b=UqcV8lw4(略);
From: Takahiko Suzuki <takahiko@iij.ad.jp>
```

  - iij.ad.jp ドメインの下に公開鍵をぶら下げられるのは iij.ad.jp の正当な所有者だけ

- **第三者署名 (Third-Party Signature)**

```
DKIM-Signature: v=1;a=rsa-sha256;c=relaxed/simple;
    h=To:From: Subject (略);d=example.com;s=foobar;t=1510031431;
    x=1511241031;bh=pvZ1fKe/ (略);b=UqcV8lw4(略);
From: Takahiko Suzuki <takahiko@iij.ad.jp>
```

  - 作成者 iij.ad.jp と署名者 example.com の関係が不明。example.com は悪意あるドメインかもしれない
  - **ほぼ役に立たない**

DKIM 署名設定

- **ヘッダの正規化は relaxed がオススメ**
  - 通過する際 MTA によって整形される場合があるため
- **ダイジェストアルゴリズムは SHA-256**
- **暗号化方式は RSA (1024bit 以上)**
- **署名の有効期限は MTA が再送を諦める時間を考慮**
  - Postfix, sendmail のデフォルトは 5日
- **署名対象ヘッダ**
  - まずは自分の組織から発信されているメールを観察する
  - RFC6376 5.4.1. を参考にする
    - 大半のケースではこれで十分
  - 自組織で使用している独自ヘッダなどあれば追加する

DKIM 公開鍵管理

omgo2._domainkey.iij.ad.jp. IN TXT “v=DKIM1; k=rsa; p=MIIBIjAN(略)"

- **3ヵ月～1年程度で交換する**
- **セレクタを使うと1つのドメインに複数の公開鍵が同時に存在でき、ローテーションをおこないやすい**
  - 送信システムを委譲する場合、セレクタに DNS の複階層を使うと管理がしやすい
    - “s=2017.office”, “s=2018.office”
    - “s=2017.marketing”, “s=2018.marketing”

DKIM 公開鍵管理

- **複数のドメインを一括で管理したい場合、鍵管理を第三者に委譲する場合は CNAME RR を使う**
  - “s=key1”, “s=key2”, “s=key3”
  - 異なるドメインへの委譲も可能

```
key1.example.jp   IN  CNAME  key1.dkim.example.com
key2.example.jp   IN  CNAME  key2.dkim.example.com
key3.example.jp   IN  CNAME  key3.dkim.example.com
key1.example.net IN  CNAME  key1.dkim.example.com
key2.example.net IN  CNAME  key2.dkim.example.com
key3.example.net IN  CNAME  key3.dkim.example.com
```

DKIM 経由で DMARC に対応させる

- **DKIM の場合に要求される alignment**
  - strict mode: AUID (DKIM-Signature ヘッダの “d=”) = ヘッダ From のドメイン
  - relaxed mode: AUID (DKIM-Signature ヘッダの “d=”) の組織ドメイン = ヘッダ From の組織ドメイン
  - やはり基本的には strict mode で実現できるように頑張る
- **第三者署名では DMARC に対応できない**
  - DKIM 対応をうたう送信事業者でも、作成者署名に対応していない場合があるので注意
- **DKIM による対応が本命**
  - エンベロープ From とヘッダ From が異なるケースに対応可
  - 転送に対応可
  - DKIM はインターネット標準 (STD76)

DKIM Crypto Update (dcrup)

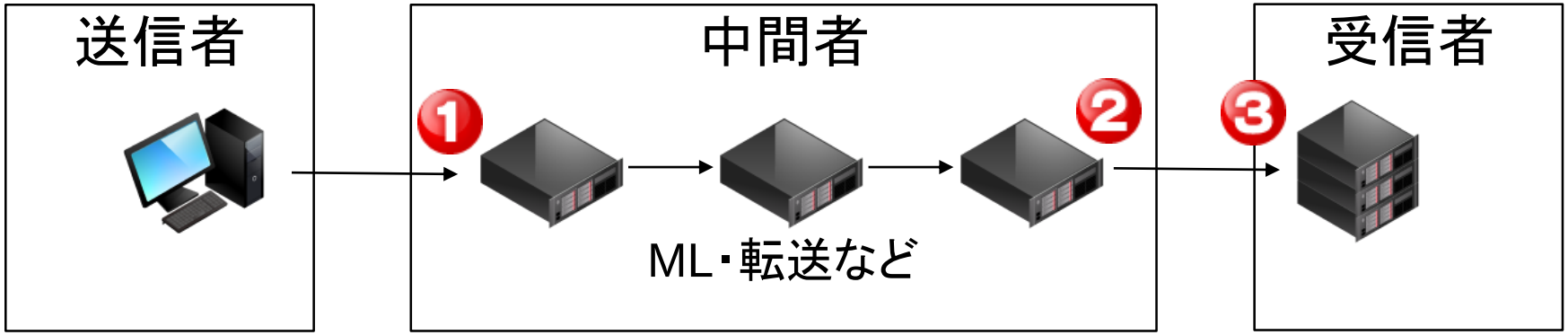
- **計算機の進化に負けないよう暗号強度を上げる**
- **RSA の場合、1156bit より大きな鍵では DKIM 公開鍵レコードが 256 byte に収まらない**
- **より強力な暗号化方式のサポート**
  - 楕円曲線デジタル署名アルゴリズム (“ecdsa256”)
  - エドワーズ曲線デジタル署名アルゴリズム (“ed25519”)
  - 参考: http://d.hatena.ne.jp/kazu-yamamoto/20171114/1510635277
- **RSA鍵のFingerprintのみをDNSに格納する (rsafp)**
  - Fingerprint は公開鍵の SHA-256 ハッシュ値
  - 公開鍵本体は DKIM-Signature ヘッダに格納し、Fingerprint を使って検証する
- **いずれの方式も受信側の普及を待ってからの投入になるため、当面は様子見でよい**

送信ドメイン認証の最後の課題

- **メーリングリスト**
  - Subject や本文が書き替えられるので DKIM 署名が壊れる
  - ML サーバが間に入るので IP アドレスに依存した SPF も使えない

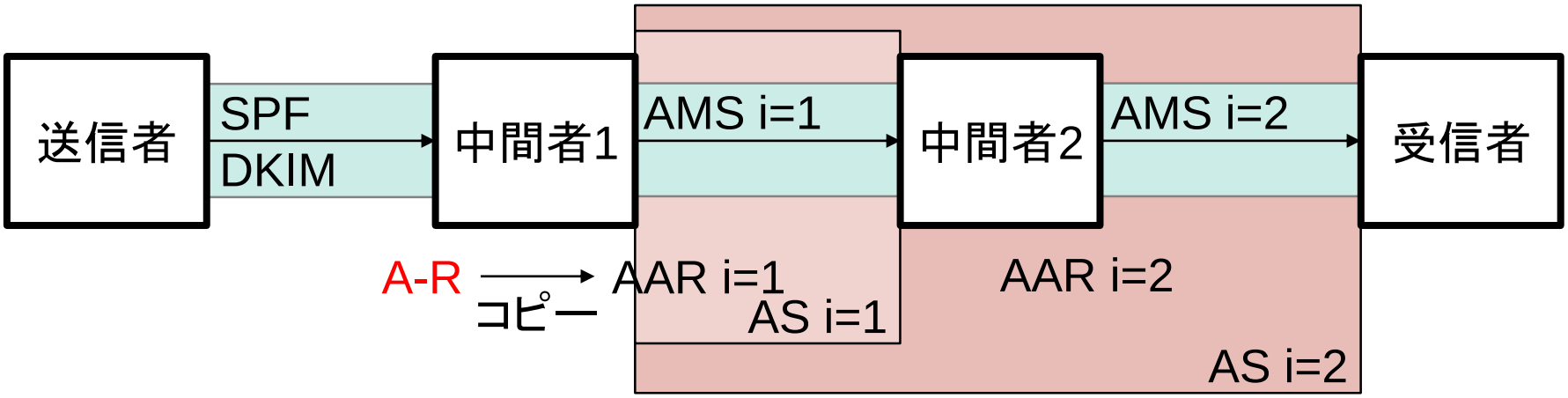
ARC (Authenticated Received Chain)

- **送信者から最初に受け取った際の認証結果をバケツリレー式に伝播させる**



- 2 で 1 の認証結果を保存し、署名をする
- 3 は 2 が署名した 1 の認証検証を参照できる

ARC



- **AAR が最初の認証結果を保存する**
- **AMS はほぼ DKIM であり、直近の送信元の正当性を提供する**
  - 中間者がヘッダや本文を改変する前提なので、直近の署名しか検証できない
- **AS が chain の正当性を保証する**
  - chain 中の全 ARC 関連ヘッダに署名し、改ざんを防ぐ

A-R: Authentication-Results
AAR: ARC-Authentication-Results
AMS: ARC-Message-Signature
AS: ARC-Seal

ARC

- **DMARC の検証ができなかった場合に参照する**
  - 転送や **ML に対応**できるようになる
  - ARC が普及すると送信ドメイン認証が完成する
- **マルチホップ可**
  - 通過するたびに検証と署名をおこなう
- **中間者が信頼できる前提**
  - ホップ上の全ての中間者を信頼できる必要がある
  - 中間者のホワイトリスト・レピュテーションが必要
  - “信頼” の連鎖 (chain)

ARC

- **受信時のポリシーは複雑になる**
  - “dmarc=reject” かつ “arc=pass“ な場合にどうするべきか？
    - DMARC を尊重して拒絶する
    - ARC を尊重して受け取る
- **普及状況**
  - Google (送受信), AOL (受信) が対応
  - 様子見の Mailbox Provider も多い
    - 特に送信側の対応がすごく手間
  - 普及するとしてももう少し先

忘れてよいもの

- Sender ID (RFC4406, RFC4407)
- DKIM ADSP (RFC5672)

受信側の導入

認証結果活用の原則

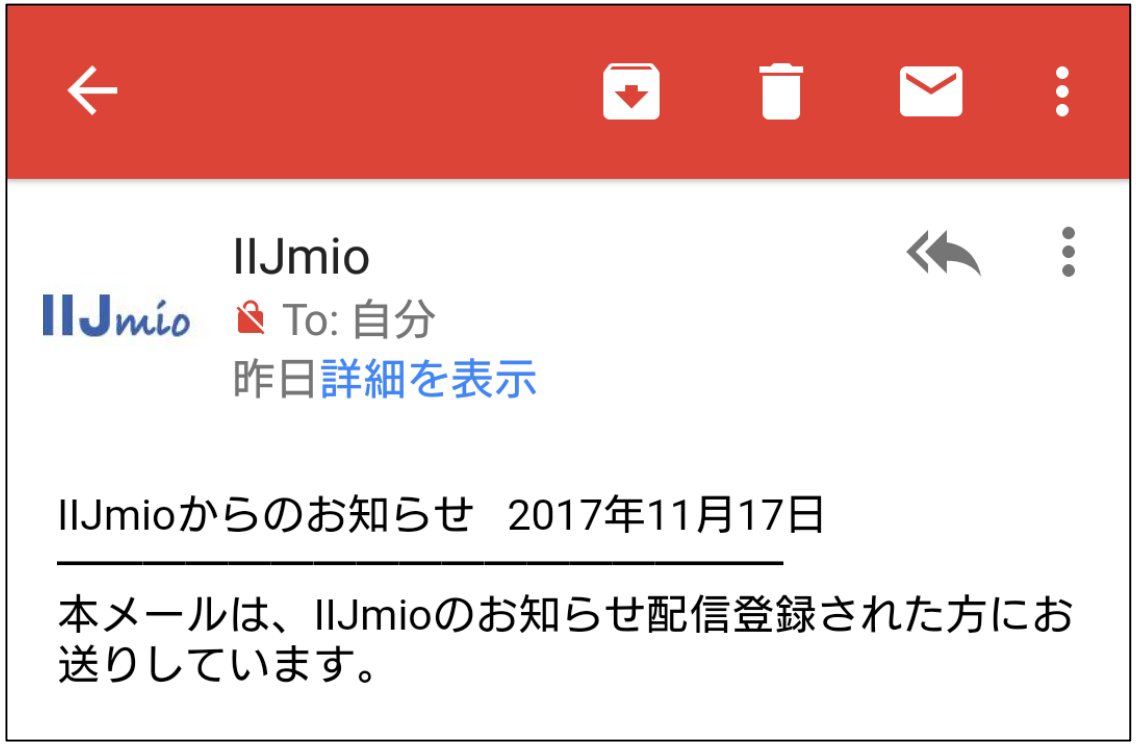
- **最初に見るのは DMARC**
  - DMARC の “quarantine”, “reject” はできる限りそのまま受け入れる
  - “reject” 宣言しているドメインにはそれだけの覚悟がある
- **SPF, DKIM のスコアは “pass” と “pass 以外” の区別で十分**
  - 正当な出口から送信されているか、そうでないか
  - DMARC の出現もあり “fail”, “softfail”, “neutral”, “none” に大きな違いはない
- **SPF, DKIM では必ずドメインと組み合わせて評価**
  - ホワイトリスト
  - ドメインレピュテーション
- **DKIM は作成者署名の検証結果のみ参照する**

認証結果活用のユースケース

- **pass しているドメインを使ったホワイトリスト**
  - 「このドメインから正当に送られているメールは spam フィルタをスキップして受け取る」
  - 見た目の似た別のドメイン（cousin domain）を使った攻撃が存在するため、ホワイトリストとの組み合わせは必須
- **認証必須ドメインの指定**
  - 「このドメインを名乗る、認証に pass していないメールをフィルタ・隔離・拒絶」
  - **自組織ドメイン**
  - 特定の会社・ドメインを騙ったフィッシングが流行った場合の受信側でできる対策
    - 本当は騙られているドメインに DMARC “reject” ポリシーを宣言して欲しい

BIMI (Brand Indicators for Message Identification)

- **DMARC の検証ができたメールにブランドのロゴ画像を表示する**
  - Webmail や MUA での利用を想定



BIMI 受信時の処理の流れ

- **まずは送信ドメイン認証**

```
DKIM-Signature: v=1; (略);d=iij.ad.jp;s=omgo2; (略)
From: Takahiko Suzuki <takahiko@iij.ad.jp>
BIMI-Selector: v=BIMI1; s=weekend.jp;
```

- **BIMI セレクタと組み合わせて DNS を参照**

```
weekend.jp._bimi. iij.ad.jp IN TXT "v=BIMI1; z=64x64,512x512;
   f=png,jpg; l=https://image.iij.ad.jp/bimi/logo/"
```

  - セレクタは宛先、時間、ブランドなどによって使い分ける

- **MUA や Webmail 用に結果をヘッダに載せる**

```
BIMI-Location: v=BIMI1; l=https://image.iij.ad.jp/bimi/logo/512x512.png,
   https://image.iij.ad.jp/bimi/logo/64x64.png
```

  - Authentication-Results ヘッダと同様、信頼関係が前提
  - サイズや画像形式はポリシーに合わせて選択

BIMI

- **DMARC 導入のモチベーションとして**
- **BIMI-Selector ヘッダで間接的にロゴの URL を指定する**
- **仕様については議論中**
  - 信頼できないドメインのロゴを表示するのは危険
  - 信頼をどう担保するかについて議論中
- **ドラフト:**
  - https://github.com/authindicators/rfc-brand-indicators-for-message-identification

まとめ

まとめ

- 送信ドメイン認証は自分のドメインを守る技術
- まず DMARC レコードを書きましょう
- 悪用される前に送信ドメイン認証の準備を粛々と進めましょう