IIJ
Internet Initiative Japan

Network Initiative

Internet Initiative Japan Inc.

DNS最新動向
- spam対策 -

株式会社インターネットイニシアティブ(IIJ)
小林 直 tkoba@iij.ad.jp

目次

- 送信ドメイン認証概要
- 普及率
- レコードの書き方SPF / Sender ID
- レコードの書き方 DomainKeys/DKIM
- その他
- まとめ

送信ドメイン認証概要 – 概要1

- 目的
  - 所有するドメインが詐称される事を防ぐ
    - 有名なドメインは詐称されやすい
  - 正しい送信者によって送信されたことを保証する
    - フィッシングの防止
- ドメイン認証技術
  - IPアドレスベース
    - SPF
    - Sender ID
  - 署名ベース
    - DomainKeys
    - DKIM

送信ドメイン認証概要 – 概要2

- レコードを書く側
  - 所有するドメインを詐称から守れる
    - フィッシング対策
    - ドメイン自体の評判低下を防止
      - 詐称されているだけでspammerのドメイン扱いされる
      - 書いてない状態では受信者は詐称されていないメールか判断できない
- レコードを判定する側
  - spam判定の一要素として使用可能
    - 詐称されているメールはspamの可能性が高い
  - 詐称されていない事が解ったらドメインのReputationを調べる
    - スパマーもレコードを記述するため
    - 判定結果が正しくてもspamの可能性はある

送信ドメイン認証概要 - SPF,Sender ID

Internet Initiative Japan Inc.

- SPF (Sender Policy Framework)
  - エンベロープFROMと接続元IPアドレスから判定
    - SMTPセッションのMAIL FROM
    - メール本文を読まなくても判定できる
    - メール転送に弱い
      - MAIL FROMはそのままで接続元IPアドレスは変わる為
- Sender ID
  - PRAと接続元IPアドレスから判定
    - PRA(Purported Responsible Address)はFrom,Sender等のヘッダから特定
      - 詳細は draft-ietf-marid-core-03.txt に記述有り
    - メール本文を読む必要がある(PRAを特定する為)
    - メール転送も考慮しているが、転送プログラムの改修が必要
      - ヘッダの追加にて対応しているため(Resent-From等)

送信ドメイン認証概要 – DomainKeys、DKIM

Internet Initiative Japan Inc.

- DomainKeys
  - メール本文とヘッダに対して署名
    - 転送はOK
    - 多くのMLでされているSubjectへの文字追加や本文差し込みに弱い
      - 仕様的には再署名で対応
- DKIM
  - メール本文とヘッダに対して署名
    - DomainKeys + IIM + α
    - 大半はDomainKeysと同じ (一部の機能をIIMから取り入れ)

普及率 - 1

JPドメインにおける普及率 2005/10

登録型	登録数	MX	SPF	DK
AD(JPNIC会員)	297	254	13	2
AC(大学系教育機関)	3217	3034	22	1
CO(一般企業)	279529	259130	634	6
GO(政府機関)	834	727	2	0
OR(会社以外の団体)	20723	19389	87	2
NE(ネットワークサービス)	17285	17285	83	5
GR(任意団体)	9043	7687	21	2
ED(小・中・高校など主に18歳未満を対象とする各種学校)	4371	3962	8	0
地域型(都道府県名、政令指定都市名、市町村名)	3880	3153	14	1
汎用JPドメイン	419485	265489	771	37
合計	761287	577012	1656	56

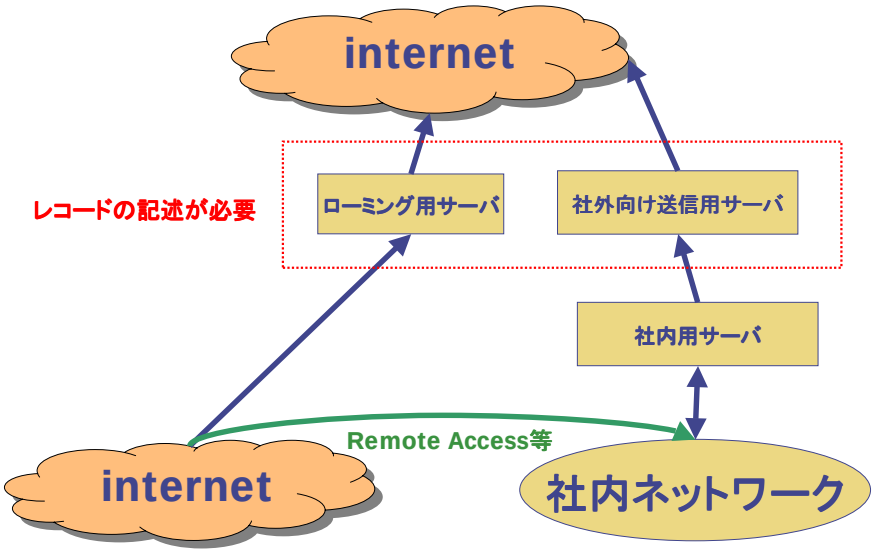
普及率 - 2

- Antispam Working Group
  - http://member.wide.ad.jp/wg/antispam/
  - JPRSとWIDEの共同研究
- 2005/10現在のJPドメインにおける普及率
  - SPF - 0.29パーセント (1656 / 577012)
  - DomainKeys - 0.00パーセント (56 / 577012)

レコードの書き方 Sender ID/SPF – 1

- レコードを書く前の準備
  - メールの出ていくサーバを特定する
    - 記述漏れがあるとそこからのメールは詐称扱いになる
    - 多いほどレコードが長くなるため可能な限り集約
  - 送信ポリシーの周知
    - 個人契約ISPから会社のメールを送らない
    - 送信時にユーザ認証を掛ける
      - 誰でも送れるのでは意味がない
  - ローミングユーザの送信手段を確保
    - リモートアクセスで社内に繋ぐ
    - 適切に認証を掛けた送信用サーバを用意
      - POP before SMTPはIP単位の認証なのでNG (NATの存在)

レコードの書き方 Sender ID/SPF – 2



レコードの書き方 Sender ID/SPF – 3

- レコードのフォーマット – 基本
  - TXTレコードを使用
  - SPF
    - v=spf1 <record> …
      - エンベロープFROMのみ判定
  - Sender ID
    - v=spf1 <record> …
      - エンベロープFROM,PRA両方に使用可能
    - spf2.0/pra <record> …
      - PRAの判定のみ使用可能
    - spf2.0/pra,mfrom <record> …
      - エンベロープFROM,PRA両方に使用可能

レコードの書き方 Sender ID/SPF – 4

- レコードのフォーマット - 修飾子
  - 基本的にPASS以外はallを修飾(マッチしなかった場合のポリシー)
    例: v=spf1 +mx … -all 等
  - + PASS
    - 正しい送信場所を宣言
  - ? NEUTRAL
    - 他の場所からもメールが出る (レコード無い状態と同じ扱いを期待)
  - ~ SOFTFAIL
    - 指定場所以外からメールを出さない (弱い否定)
  - - FAIL
    - 指定場所以外からメールを絶対出さない(強い否定)

最初は?allから始めて問題ないことを確認して~all、-allへ

レコードの書き方 Sender ID/SPF – 5

- レコードのフォーマット – 詳細
  - a
    - aレコードを引いて特定できたIPアドレスならOK
    - 例 a:smtp.example.jp
  - ip4,ip6
    - 指定したネットワークアドレス内ならOK
    - 例 ip4:127.0.0.0/24
  - mx
    - mxレコードを引いて、更に記述されているホストのaを引く
      判明するIPアドレスに一致していればOK
    - 例 mx:example.jp
  - ptr
    - アクセス元IPの逆引き結果が後方一致していればOK
    - 例 ptr:example.jp

レコードの書き方 Sender ID/SPF – 6

- レコードのフォーマット – 詳細
  - all
    - すべてにマッチ
    - 例 ~all
  - exists
    - 指定のAレコードが存在すればOK
    - 例 exists:%{ir}.example.jp
      (1.0.0.127.example.jp などに展開される)
  - include
    - 指定のTXTレコードを引いて再帰的に評価(判定が終わったら戻る)
    - 例 include:spf1.example.jp
  - redirect
    - 指定のTXTレコードを引いて再帰的に評価(判定が終わっても戻らない)
    - 例 redirect:spf1.example.jp

レコードの書き方 Sender ID/SPF – Case study

Internet Initiative Japan Inc.

- DNSレコード

```
example.jp      IN MX  10 mx.example.jp
mx.example.jp IN A        10.0.1.1
example.jp      IN A        10.0.1.2
example.jp      IN TXT     “v=spf1 ip4:10.0.0.0/24 a mx -all”
```

- 判定フロー
  1. 接続元IPが10.0.0.0/24内にマッチすればPASSそれ以外なら次へ
  2. example.jpのA(10.0.1.2)と一致すればPASSそれ以外なら次へ
  3. mx.example.jpのA(10.0.1.1)と一致すればPASSそれ以外なら次へ
  4. allですべてマッチするのでFAIL

レコードの書き方 Sender ID/SPF – DNS

Internet Initiative Japan Inc.

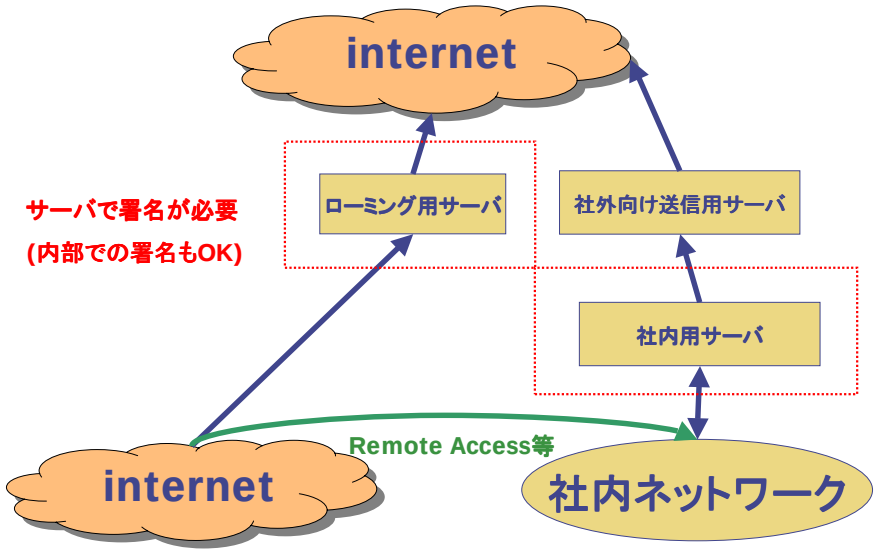
- DNS色々
  - DNSサーバ的にも優しいレコードを書こう
    - ダメな実装でも誤判定しないように
    - 複雑すぎるレコードは嫌がらせ(多重include等)
  - 複雑に書くとその分DNSサーバへの問い合わせが増える
    - 最小でTXTレコード分の1クエリ 必要
    - a,exists,ptr　　1クエリ
    - mx　　MXの1クエリ + MXレコード分のAクエリ数
    - include,redirect　参照先のレコード分が加算
    - ip4,ip6,all　　0クエリ、(ip6は一部の実装でエラー)

ip4のレコードとallだけが一番安全

レコードの書き方 DomainKeys、DKIM - 1

- レコードを書く前の準備
  - 基本的な準備はSPFの場合と同じ
    - メールの出ていくサーバを特定
    - 送信ポリシーの周知
  - 相違点
    - サーバに署名用の秘密鍵を持つ必要がある
    - **署名するプログラムを用意する必要がある**
    - 送信するサーバと署名するサーバが別でも問題ない
      - 内部で署名して、別サーバでインターネットに送信など
    - 公開鍵はDNSに登録
      - 1ドメインあたり複数の鍵を持つことが可能

レコードの書き方 DomainKeys、DKIM - 2



レコードの書き方 DomainKeys、DKIM - 3

- レコードのフォーマット – ポリシー
  - ドメインのポリシーを記述
    - _domainkey.ドメイン名のTXTレコードで記述
      - 例: _domainkey.example.jp IN TXT “t=y; o=~”
  - 使用可能タグ
    - t
      - テストモード(検証失敗しても成功と同じ扱いを要求)
      - 使用可能文字は y のみ
    - o
      - 該当ドメインのサインポリシー
      - ~サインされていないメールもある、 -すべてサインされている
    - r
      - コンタクトアドレス (検証失敗時の問い合わせ等)
    - n
      - コメント

レコードの書き方 DomainKeys、DKIM - 4

- レコードのフォーマット - セレクタ
  - 署名の検証に使用する公開鍵を登録
  - <keyname>._domainkey.ドメイン名 のレコードで記述
    - 例: key1._domainkey.example.jp IN TXT "k=rsa; t=y; p=MF…“
  - 使用可能タグ
    - g
      - 鍵の使用用途 (ドメイン単位、ユーザ単位の署名等)
    - k
      - 鍵のタイプ(デフォルトはrsa)
    - p
      - 公開鍵データ(BASE64エンコード)
    - t
      - テストモード
    - n
      - コメント

レコードの書き方 DomainKeys、DKIM - 5

- レコードのフォーマット - セレクタ (DKIM拡張)
  - 使用可能タグ
    - v
      - レコードのバージョン。現時点では“DKIM1”のみ
    - h
      - 使用するハッシュのアルゴリズム。標準では”sha1”
    - s
      - レコードの使用可能用途。“*” 何でも ”email” EMAIL のみの2種類。

レコードの書き方 DomainKeys、DKIM – Case study

- DNSレコード
  - _domainkey.example.jp IN TXT “t=y; o=~”
  - demo._domainkey.example.jp IN TXT “v=DKIM1; k=rsa; p=…”
- ヘッダ
  - DKIM-Signature: a=rsa-sha1; c=nowsp; d=example.jp; s=demo; h=To:…; b=Yw…
- 判定フロー
  1. demo._domainkey.example.jp 引いて公開鍵を取得
     - 鍵が無かった場合は _domainkey.example.jpを引いてポリシー判定 で終了
  2. 取得できた鍵を元にメールを検証
     - 失敗時は_domainkey.example.jpを引いてポリシー判定 で終了

レコードの書き方 DomainKeys、DKIM – DNS

Internet Initiative Japan Inc.

- DNS色々
  - レコードの構造的に殆ど負荷にならない
    - 再帰的に参照する仕組みが無いため
    - 現状では対象も少ない
  - 普及率を見て解るとおり使用しているところも少ない
    - jp以外でもSPFに比べると圧倒的に少ない
    - サインする実装が必要なことも原因では

その他

Internet Initiative Japan Inc.

- SPF レコードが IANAに登録 (99)
  - http://www.iana.org/assignments/dns-parameters
- bind9は未知のレコードも扱える
  - RFC3597 - Handling of Unknown DNS Resource Record (RR) Types
- 気になる点
  - 対応していないサーバは？
  - キャッシュサーバやクライアント側リゾルバの対応は？
  - 当面はTXTレコードで十分

まとめ

- 適切に運用しようとすると結構大変
  - 間違った書き方による正常なメールの受信拒否
    - 検証結果によって受信拒否する所も
    - メールサーバを追加した場合の考慮漏れ
  - メール配送系の再設計が必要になる場合も…
    - これを期に見直しを
- 自己防衛
  - 送信者詐称により加害者として誤認される事は避けられる
  - 何もしなければspammerドメイン扱いに
  - 一度失った信用を取り戻すのは大変
    - From:example.co.jp　REJECT とか書かれたら…

関連リンク

- internet-drafts
  - http://www.ietf.org/internet-drafts/draft-delany-domainkeys-base-03.txt
  - http://www.ietf.org/internet-drafts/draft-allman-dkim-base-00.txt
  - http://www.ietf.org/internet-drafts/draft-lyon-senderid-core-01.txt
  - http://www.ietf.org/internet-drafts/draft-schlitt-spf-classic-02.txt
- IANA
  - http://www.iana.org/assignments/dns-parameters
- RFC
  - http://www.ietf.org/rfc/rfc3597.txt