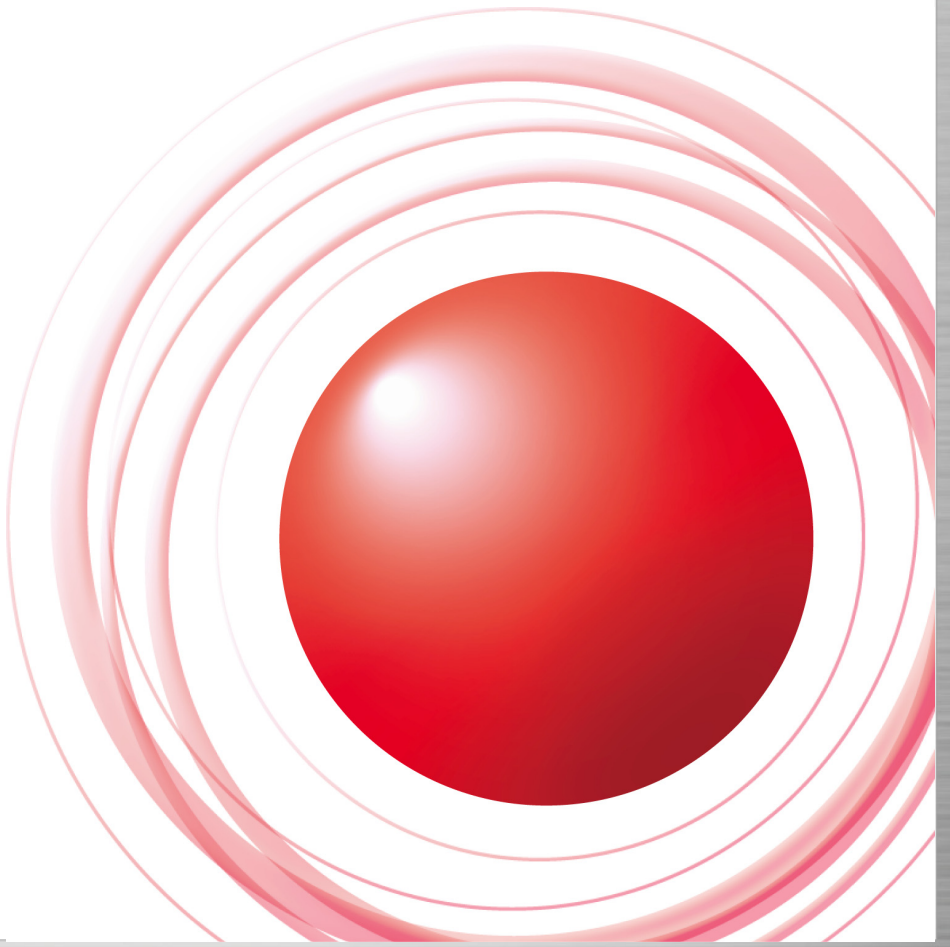
IAjapan 第7回 迷惑メール対策カンファレンス

# ENMA による送信ドメイン認証導入実践

IIJ Internet Initiative Japan



2009/5/19
株式会社インターネットイニシアティブ
メッセージングサービス部 開発運用課
鈴木高彦

Ongoing Innovation

# ENMA とは

- 送信ドメイン認証の (受信側) 検証をおこなう milter
  - Sendmail、Postfix と連携動作
- 認証結果をヘッダとして挿入

認証結果ヘッダの例

```
Authentication-Results: mx.example.jp;
    spf=pass smtp.mailfrom=username@example.com;
    sender-id=pass header.From=username@example.com;
    dkim=pass header.i=@example.com;
    dkim-adsp=pass header.From=username@example.com
```

# 特長

- SPF/Sender ID/DKIM/DKIM ADSP をサポート
  - 複数の認証方式を一括導入
- IIJ サービス環境での運用実績
  - 安定・高速
- オープンソース
  - BSD ライセンス
- 認証エンジン部分をライブラリ化

# 送信ドメイン認証の導入

# 送信ドメイン認証の導入

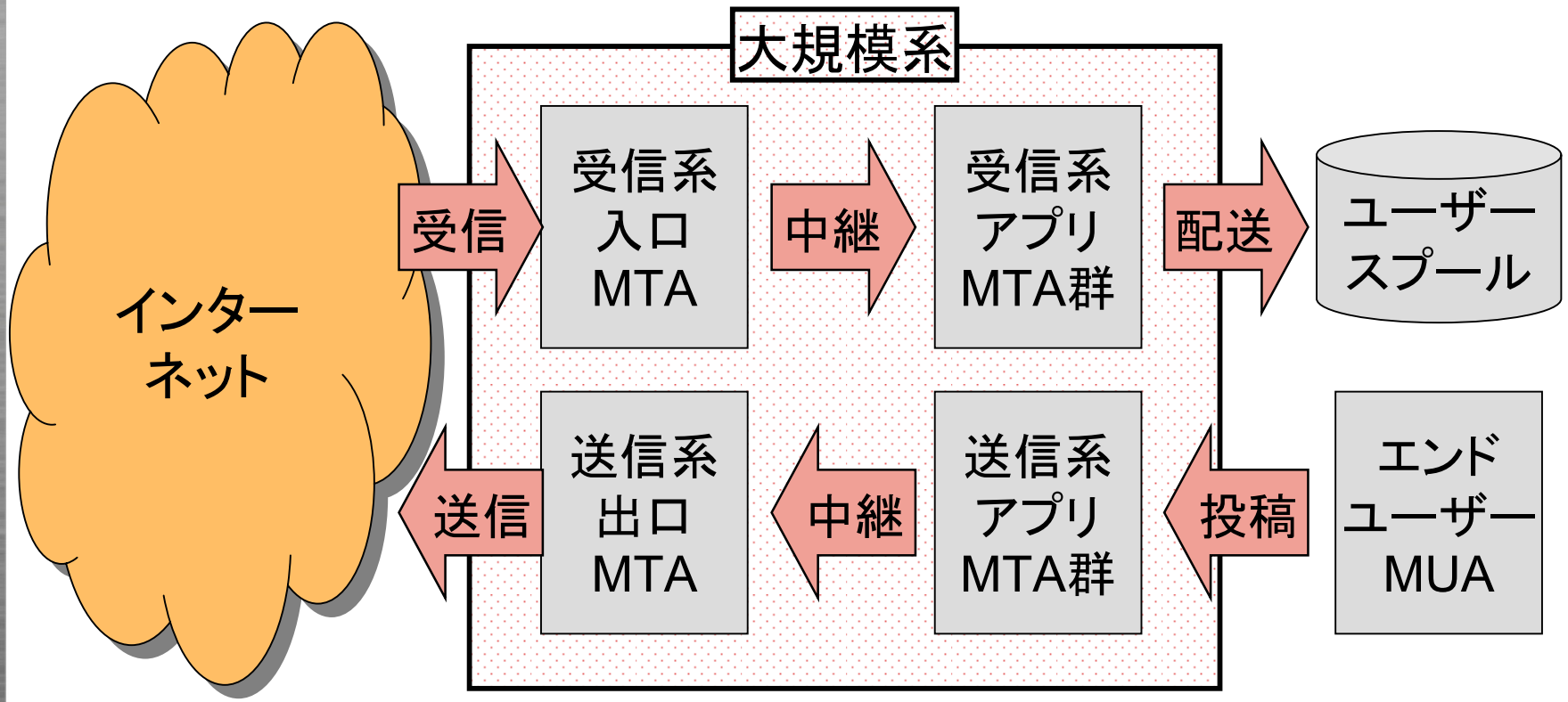
- システム構成のポイント
- ENMA 導入のポイント

# システム構成のポイント

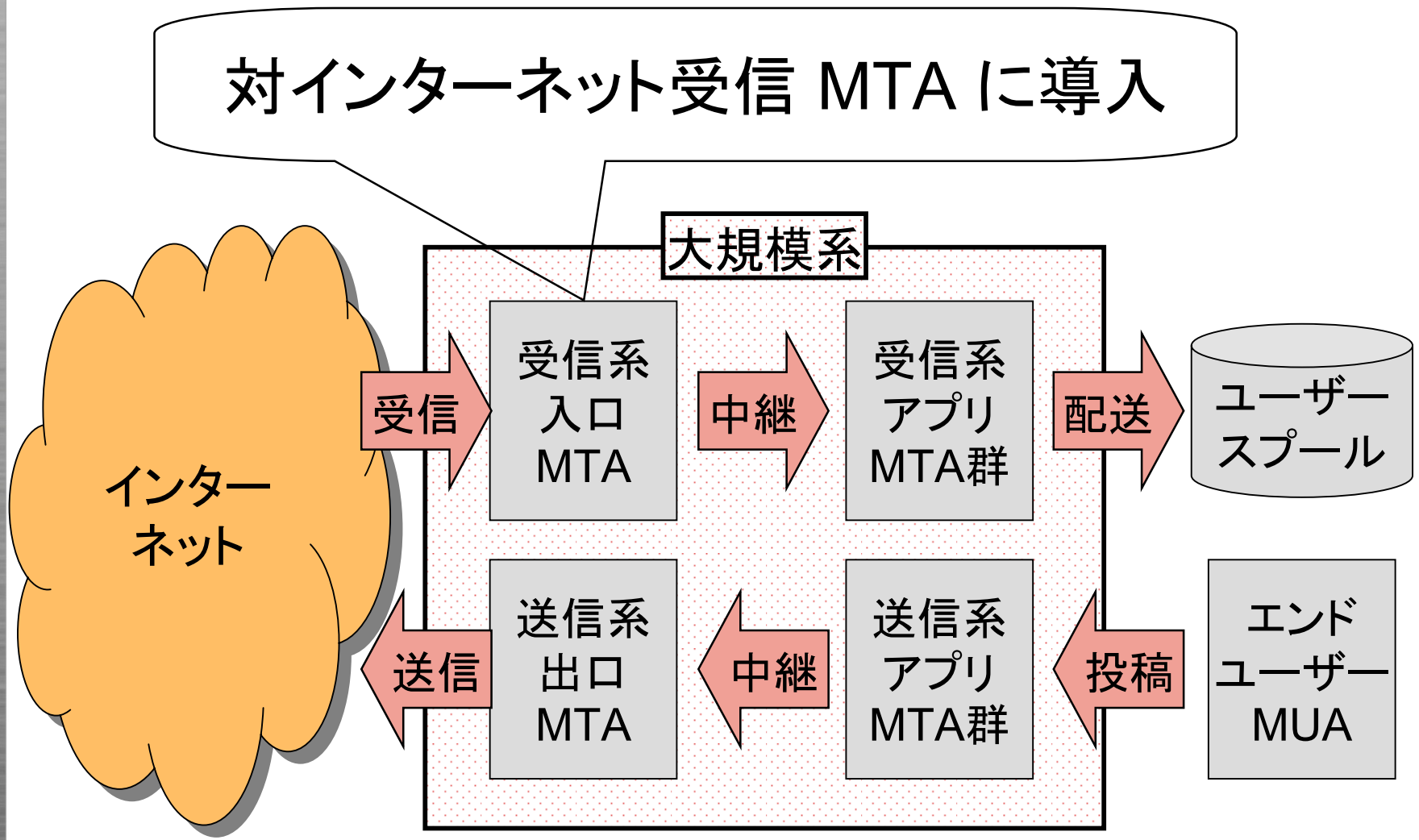
- インターネットから直接メールを受け取る メールサーバ (MTA) に導入する
  - 送信側 MTA の IP アドレスを取得する必要があるため (SPF/Sender ID)
  - メールヘッダや本文が系の内側で改変される影響を受けない (DKIM)
  - Reject などのアクションがとりやすい
- インターネットに送信するメールに認証処理をおこなわないようにする
  - 送信メールに Authentication-Results ヘッダを付与しないように

# 大規模な系への導入

- 受信用と送信用の系を分離している
- 受信用の系が多段構成になっている

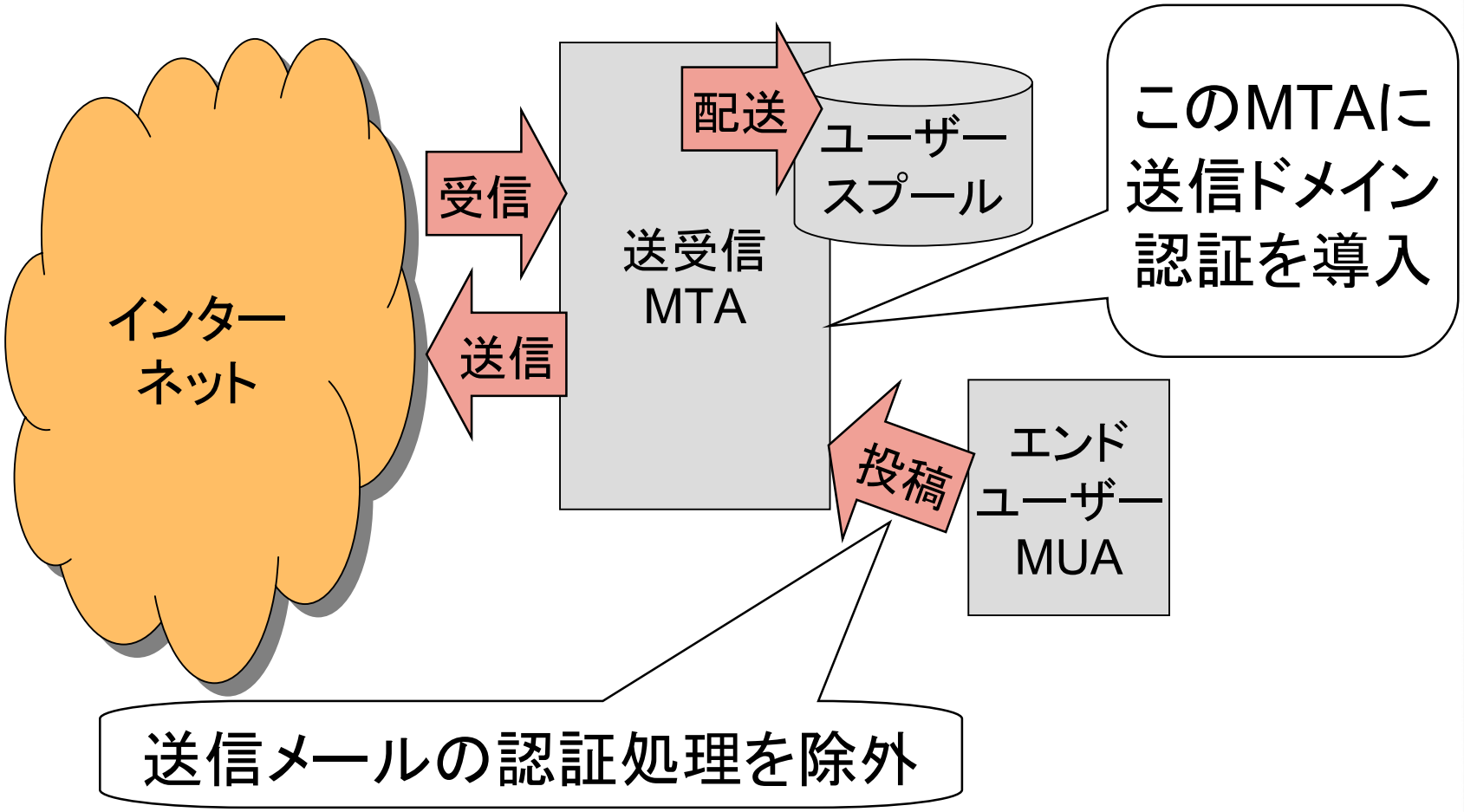


# 大規模な系への導入



# 小規模な系への導入

- 送受信全てを1箇所で処理している



# ENMA 導入のポイント

- ENMA の導入/設定
- MTA の設定

# ENMA の導入/設定

## enma.conf

- MTAからの接続を受けるソケット

milter.socket:  inet:10025@127.0.0.1

- (必要に応じて) 認証処理の除外アドレスレンジ

common.exclusion_addresses: 192.0.2.0/24

- Authentication-Results ヘッダで使用するサーバ識別子

authresult.identifier: mx.example.jp

## その他

- PIDファイル作成用ディレクトリの作成

$ mkdir /var/run/enma
$ chown daemon:daemon /var/run/enma
(ディレクトリやユーザー/グループなどは要件に応じて変更)

# MTA の設定

## Sendmail の場合

- **sendmail.mc**

```
INPUT_MAIL_FILTER(`enma', `S=inet:10025@127.0.0.1')
```

- **enma.conf** (Sendmail 8.13以前の場合のみ)

```
milter.sendmail813: true
```

## Postfix の場合

- **main.cf**

```
smtpd_milters = inet:127.0.0.1:10025
```

- **enma.conf**

```
milter.postfix: true
```

# これだけで導入は完了！

# 認証結果の活用

# 認証結果の活用

- 認証結果の意味/扱い方
  - 認証結果のどこに注目すればよいか
- 送信ドメイン認証で何ができるか
  - 具体的な認証結果の活用例

# 認証結果の参照

- Authentication-Results ヘッダ (RFC5451) に認証結果を記録
  - 複数の認証結果をまとめて記述できる
  - RFC4408 が定める Received-SPF ヘッダは使用しない
  - 受信時に既に付いていた Authentication-Results ヘッダは削除する
    - 偽の認証結果を記載したヘッダの挿入を防ぐ

# 各認証方式のおさらい

- SPF/Sender ID
  - 正当なIPアドレスから送信されているか検証
- DKIM
  - 署名者 (Signer) による正当な署名の確認
    - **送信者 (Author) の確認はしない**
  - メールの改ざん検知
- DKIM ADSP
  - 送信者の正当な署名 (Author Domain Signature) の検証
  - Author Domain Signature がないメールの扱い

# 認証結果の例 - SPF

trusted-user@example.com が送信し、example.jp が受信したメールの認証結果

認証をおこなったサーバー

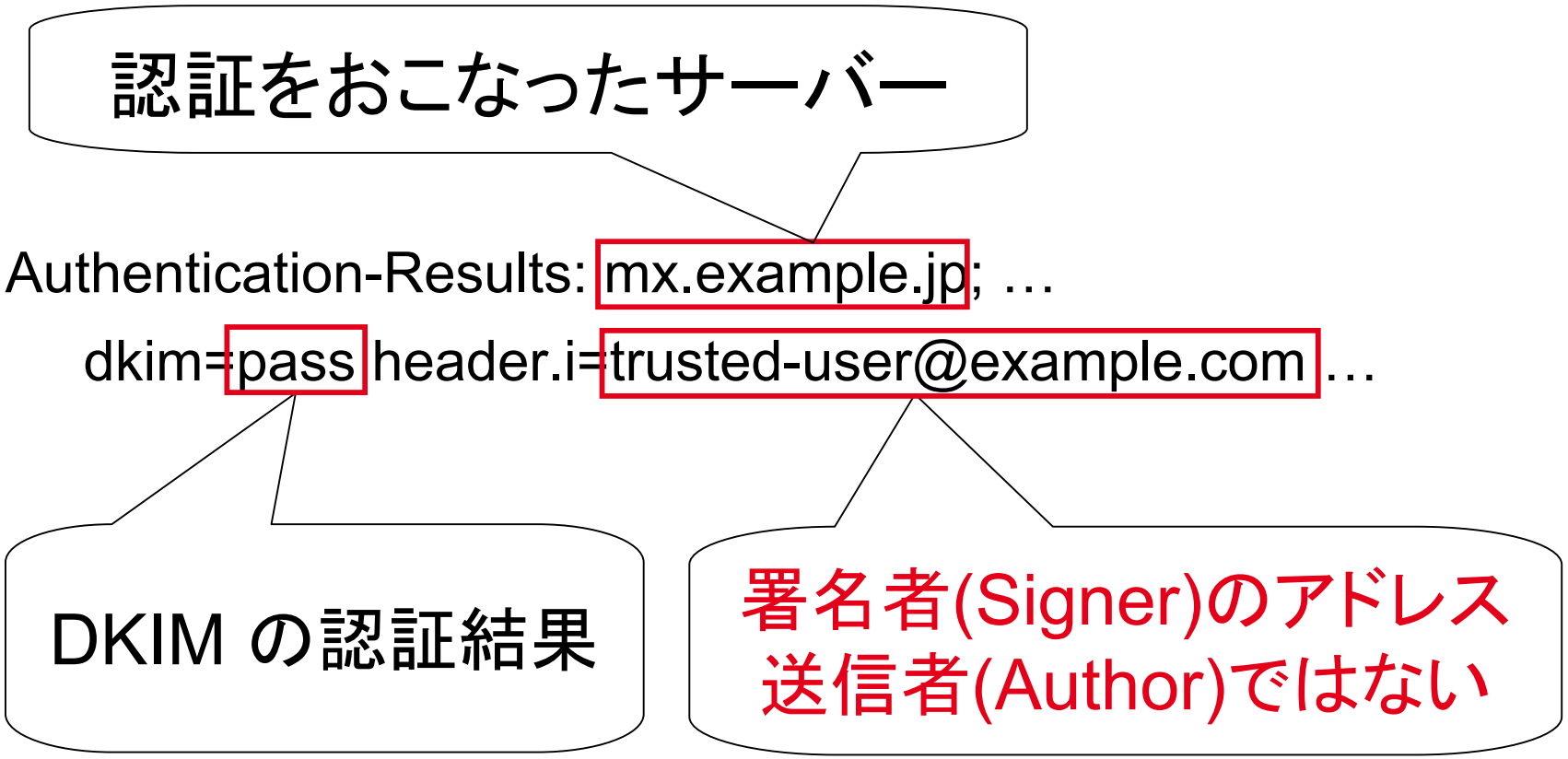
Authentication-Results: mx.example.jp;
spf=pass smtp.mailfrom=trusted-user@example.com …

SPF の認証結果

信頼できるアドレス
(pass の場合のみ)

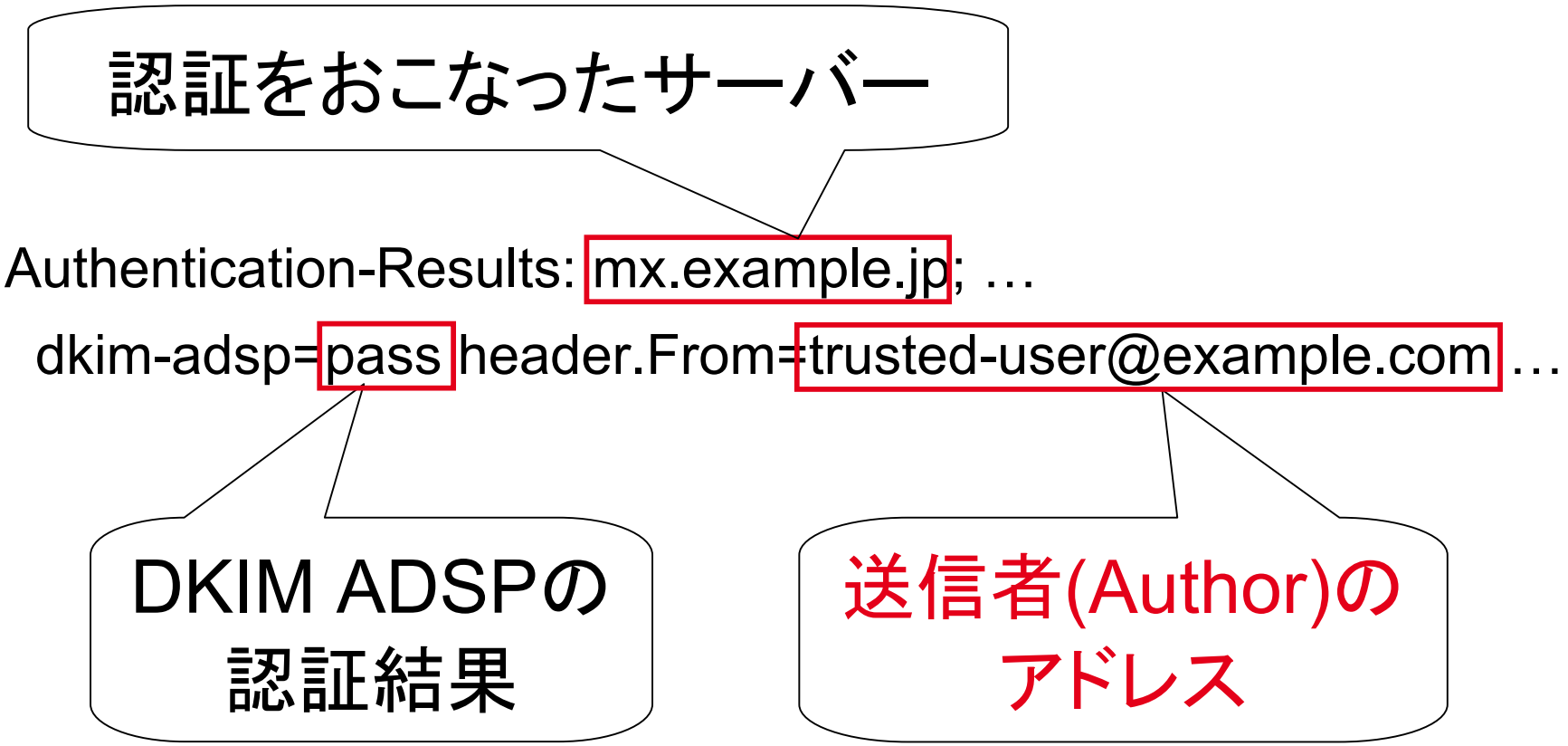
# 認証結果の例 - DKIM

trusted-user@example.com が署名し、example.jp が受信したメールの認証結果



# 認証結果の例 - DKIM ADSP

trusted-user@example.com が送信し、example.jp が受信したメールの認証結果



# SPF/Sender ID の認証結果

<table>
<tr><th>認証結果</th><th colspan="3">意味</th></tr>
<tr><td>none</td><td colspan="3">送信側が SPF/Sender ID に対応していない</td></tr>
<tr><td>pass</td><td colspan="3">認証成功、正当なホストから送信されている</td></tr>
<tr><td>neutral</td><td rowspan="5">認証失敗</td><td rowspan="3">不正なホストから…</td><td>送信されているかもしれない</td></tr>
<tr><td>softfail</td><td>送信されている可能性が高い</td></tr>
<tr><td>hardfail</td><td>送信されている</td></tr>
<tr><td>temperror</td><td colspan="2">DNS エラーなどの一時的なエラー</td></tr>
<tr><td>permerror</td><td colspan="2">送信側の認証情報の記述ミス</td></tr>
</table>

# DKIM の認証結果

| 認証結果 | 意味 | |
|---|---|---|
| none | メールは署名されていない | |
| pass | 認証成功、署名者による署名を確認 | |
| neutral | 認証失敗 | 署名認証の処理過程のエラー |
| fail | | メールは改ざんされている |
| temperror | | DNS エラーなどの一時的なエラー |
| permerror | | 送信側の認証情報の記述ミス |

# DKIM ADSP の認証結果

<table>
<tr><th>認証結果</th><th colspan="3">意味</th></tr>
<tr><td>none</td><td colspan="3">送信側が DKIM ADSP に対応していない</td></tr>
<tr><td>pass</td><td colspan="3">認証成功、送信者による署名を確認</td></tr>
<tr><td>unknown</td><td rowspan="7">認証失敗</td><td rowspan="3">そのドメインからDKIM署名のないメールは…</td><td>送信され得る</td></tr>
<tr><td>fail</td><td>送信されない</td></tr>
<tr><td>discard</td><td>送信されないので署名のないメールは破棄してもよい</td></tr>
<tr><td>nxdomain</td><td colspan="2">送信者のドメインは存在しない</td></tr>
<tr><td>temperror</td><td colspan="2">DNS エラーなどの一時的なエラー</td></tr>
<tr><td>permerror</td><td colspan="2">送信側の認証情報の記述ミス</td></tr>
</table>

# 認証結果の意味

- 複数の認証方式
- 多様な認証結果
- どこに注目すればよいか?
- どのように解釈すればよいか?

# 注目すべき認証結果

- SPF/Sender ID の認証成功 (pass)
  - 正当な送信ドメインからの直送の確認
  - Envelope From/Header From は信頼できる
- DKIM ADSP の認証結果全般
  - 送信ドメインによる正当な署名 (Author Domain Signature) の確認
  - Header From は認証結果に基づいて扱ってよい
  - ただし、DKIM ADSP を宣言しているドメインはまだ少ない

# 注意すべき認証結果

- SPF/Sender ID/DKIM の認証失敗
  - SPF/Sender ID は転送によって、DKIM は ML などによって、悪意がなくても認証に失敗する
- DKIM の認証成功 (pass)
  - 署名の完全性を示すもので、“誰の”署名であるかには関知しない
    - 送信者 (Header From) と署名者が全く別の場合もある
  - 送信ドメインによる署名の確認は DKIM ADSP の結果を参照する

# 注意すれば使えるケース

- よく詐称に使われる特定ドメインを名乗る迷惑メールを捨てる
  - 転送やML投稿…
    - をしないことがわかっている場合
    - による認証失敗を許容できる場合
  - 特定ドメインの認証失敗に基づきブラックリストを適用する
  - ただし、<u>一般に認証失敗結果のブラックリストへの適用は効率的ではない</u>
    - 悪意あるドメインから正当に送信されるメール

# 認証結果による振り分け例 (1)

“信頼性のある”ホワイトリストの実現 (SPFの例)

認証をおこなったサーバー名が一致

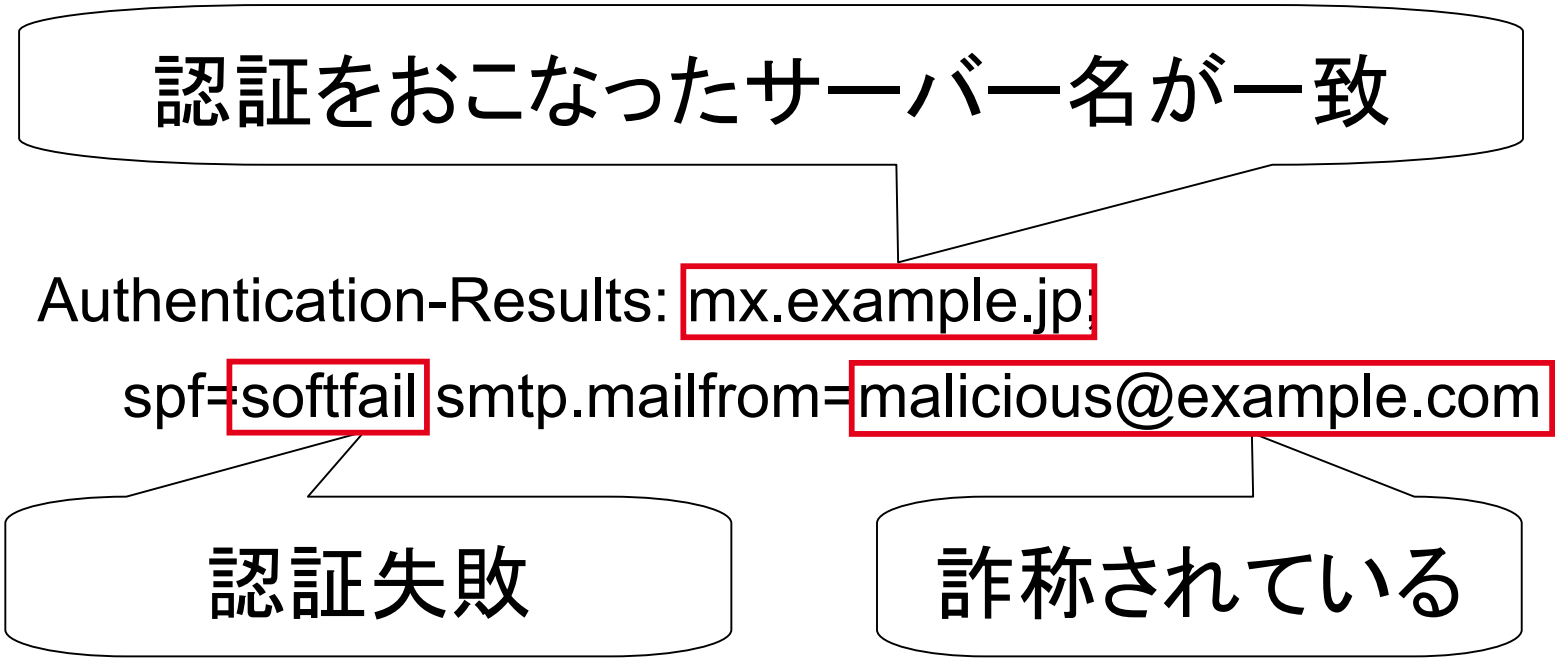
Authentication-Results: mx.example.jp;
spf=pass smtp.mailfrom=trusted-user@example.com

SPF 認証成功

このアドレスにホワイトリストを適用する

# 認証結果による振り分け例 (2)

送信元情報が詐称されているメールを隔離する




⇨ 隔離　※転送や正当な送信者が誤ったサーバから送信してしまうケースに注意

# まとめ

- 送信ドメイン認証の受信側の導入は簡単
- 現状ではホワイトリストとしての利用が有効
- 認証失敗をフィルタで利用する場合は慎重に
- 将来的には、ドメインによるレピュテーション

# 今後の展望

# 今後の展望

- 認証結果に基づくアクションの実行
  - ドメイン指定
    例) 特定のドメインが騙られた場合のみ受信拒否したい
  - 受信/拒否/スロットリング
- DKIM 署名
- コンテンツの整備
  - Web サイト
  - 各種ドキュメント
  - http://enma.sf.net/

# 手伝っていただける方も募集中

- バグ報告
- 各種アイデア/提案
- 各OS/ディストリビューション向けパッケージング
- ドキュメント整備
  - マニュアル
  - 導入記
  - Webサイト

enma-users-jp@lists.sourceforge.net
までご一報ください