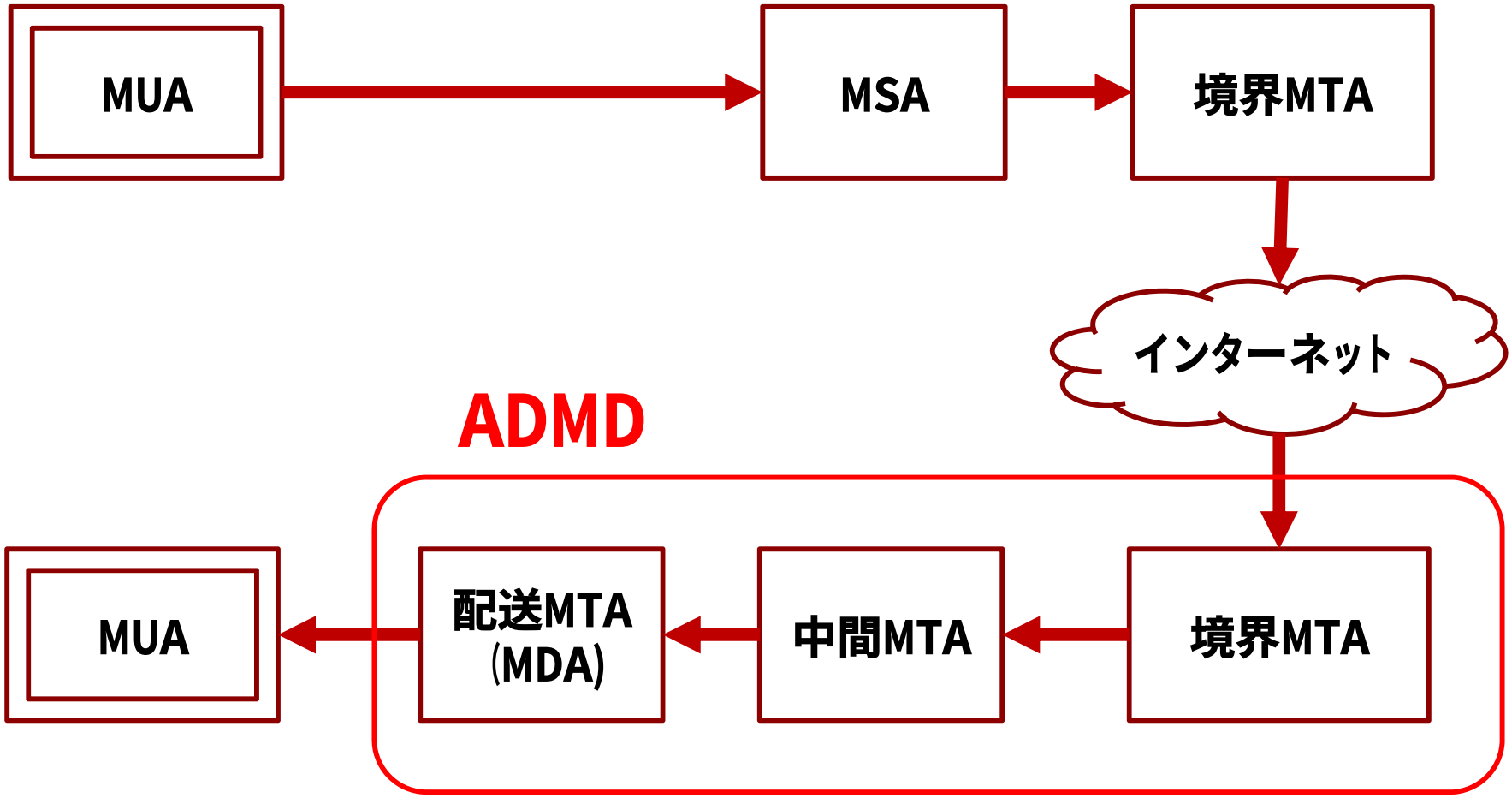
RFC 5451
Message Header Field for Indicating Message Authentication Status
概要

- **RFC 5451**
  - Message Header Field for Indicating Message Authentication Status
- **概要**
  - Authentication-Resultsメッセージヘッダーの定義とMUA/MTA側の対応について

- 用語
  - 境界MTA
    - インターネットと組織境界内のユーザとの間のゲートウェイとして機能するMTA
  - 配送MTA
    - ユーザーのinboxまたは他の最終配送先へのメッセージ配送を行うMTA
  - ADMD（管理ドメイン）
    - 自組織で信頼出来る範囲を指す

メール配信の流れ



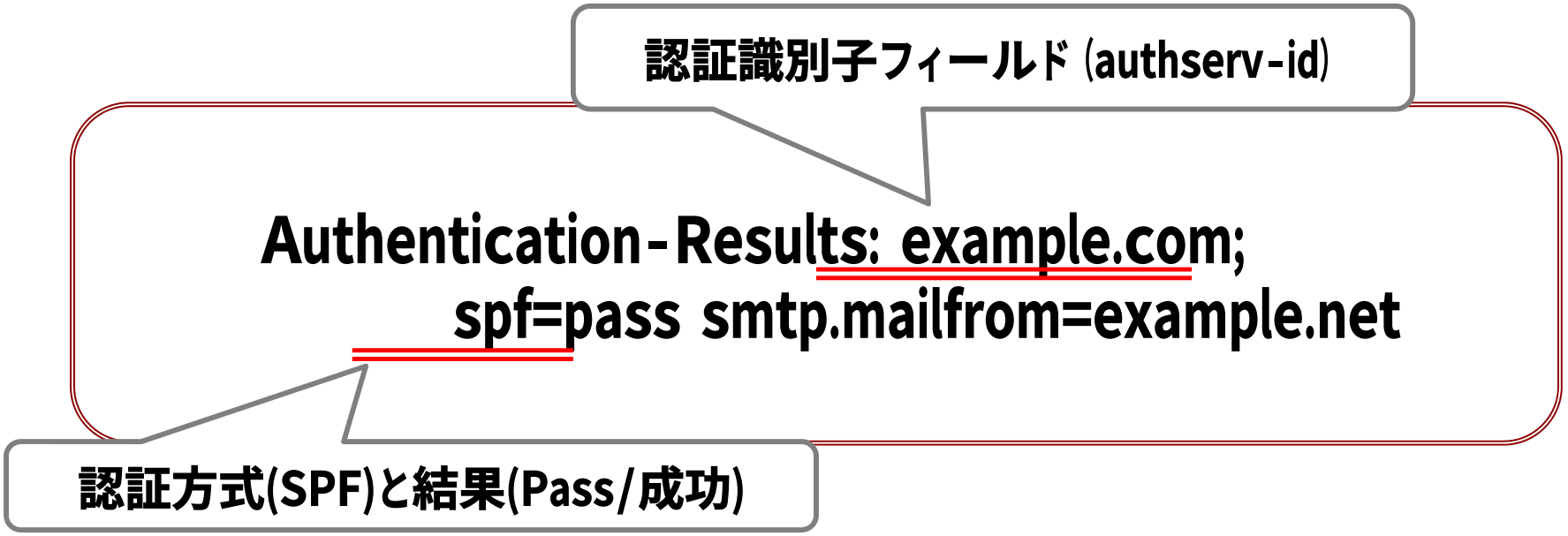
• Authentication-Resultsメッセージヘッダーとは

- メール認証 (SPF,DKIMなど)の結果を保持する
- いくつかのMTAを中継し、複数のヘッダーが追加されることもある
- ADMD内のMTAでは、管理上同じ認証識別子フィールド (authserv-id) を使うことが望ましい
- 書式は以下 (詳細はRFC P.8-10参照)

```
authres-header = “Authentication-Results:” [CFWS] authserv-id
        [ CFWS version ]
        ( [CFWS] “;” [CFWS] “none” / 1*resinfo ) [CFWS] CRLF
```

• Authentication-Resultsヘッダー例 1

– 本仕様に準拠しているMTAによって配送され、何らかのメッセージ認証が適用された



• Authentication-Resultsヘッダー例 2

– 本仕様に準拠している2つの別々のMTAを通して、複数のメッセージ認証が適用された

ヘッダーは上に追加されていく

```
Authentication-Results: example.com;
        sender-id=hardfail header.from=example.com;
        dkim=pass (good signature) header.i=sender@example.com

Authentication-Results: example.com;
        auth=pass (cram-md5) smtp.auth=sender@example.com;
        spf=hardfail smtp.mailfrom=example.com
```

初期対応の認証方式

- 認証方式はRFC改訂で追加されることがある
- 新方式の実装はIANAに登録されるまで”x-”プレフィックを使用
- 仕様は他RFC参照

認証方式	仕様文書	補足
auth	RFC4954	SMTP認証 (SMTP AUTH)
dkim	RFC4871	
domainkeys	RFC4870	
iprev	本RFC (RFC5451)	IP逆引きによる検証
sender-id	RFC4406	
spf	RFC4408	

認証結果一覧

- 結果詳細はRFC P.12-15参照

認証方式	結果
auth	none, pass, fail, temperror, permerror
dkim	none, pass, fail, policy, neutral, temperror, permerror
domainkeys	none, pass, fail, policy, neutral, temperror, permerror
iprev	pass, fail, temperror, permerror
sender-id	none, pass, policy, neutral, temperror, permerror, hardfail, softfail
spf	none, pass, policy, neutral, temperror, permerror, hardfail, softfail

- セキュリティ・問題点
  - 偽のヘッダーが付与されている可能性がある
    - 基本的にADMD以外が付与したヘッダーは信用すべきではない
    - 偽ヘッダーに対してMTA/MUAそれぞれで対策を講じる必要がある。

MTAの対応

- 検証した情報のみヘッダーに付与する
  - 例えばSPFではドメイン部分の検証なので、ローカルパートの情報は付与しない（info@example.com を
- ヘッダーは上に追加してゆく
- メールサーバーのポリシーで受信拒否する場合はDSNを投げるのではなくてSMTPで拒否すべ
- 偽ヘッダーへの対応
  - 信頼出来るホスト一覧を所持する
  - 信頼されていないMTAのヘッダーについて
    - すべて削除してもよい
    - 自組織のauthserv-idを使っていた場合は削除しなければならない

- MUAの対応
  - 指定があった場合のみヘッダー情報を解釈する
    - デフォルトOFFにしておく
  - 偽ヘッダーへの対応
    - 信頼出来ると判断したADMDが付与したヘッダーのみ解釈する
      - 信頼出来る識別子を登録しておく

- 各認証方式のサンプル

auth

```
Authentication-Results: example.com;
        auth=pass (cram-md5) smtp.auth=sender@example.com
```

dkim

```
Authentication-Results: example.com;
        dkim=pass (good signature) header.i=sender@example.com
```

domainkeys

```
Authentication-Results: example.com;
        domainkeys=pass header.from=info@example.com
```

iprev

```
Authentication-results: mail.sample.com;
        iprev=pass policy.iprev=192.168.9.50 (bmsred.sample.com)
```

sender-id

```
Authentication-Results: example.com;
        sender-id=hardfail header.from=example.com
```

SPF

```
Authentication-Results: example.com;
        spf=pass smtp.mailfrom=example.net
```

今後の課題

- 信頼性向上の案
  - プロトコルを拡張
    - SMTPなどの拡張で偽ヘッダーを防ぐことが出来る
  - 署名を付ける
    - DKIMのように組織が署名を付けることでヘッダーを信頼出来るようになる。