5. メールシステムのデザインと sendmail.cf

Contents

- sendmail.cf とは
- ドメインマスタ
- NULL Client
- PPP Client
- Firewall とメールサーバ
- バックアップメールサーバ
- バーチャルドメイン
- ルールセットのテスト

sendmail.cf とは

- sendmail が参照するファイルの位置を指定
- 配信経路の決定
  - 受理(ローカルへの配信)
  - 送信・転送(リモートへの配信)
    - どのホストに送ろうか
- 配信方法の定義
  - SMTP
  - UUCP

標準的な sendmail.cf

- OS にも標準で添付
  - sendmail に合ったバージョンの sendmail.cf を
- 基本機能
  - 当該ホスト名宛てのメールを受理
  - MX を参照して配信
    - 特に何も設定を変更する必要はない
- 変更が必要な場合
  - メール・ドメインマスタ
  - Lower MX として MX RR の右辺に指定

sendmail.cf のバージョン

- V1: sendmail 5 以前 (無指定は V1)
- V2, V3: sendmail 6.x
- V4: sendmail 8.5 まで
- V5: sendmail 8.6.x
- V6: sendmail 8.7.x
- V7: sendmail 8.8.x

- mqueue ファイルのバージョン
  - 新しい sendmail の生成した qf は古い sendmail で解釈できない

古いsendmail.cfは使えるの?

- sendmail R8 でも基本的に利用可能
- SunOS による拡張($%yなど)には非対応
- Null address (<>, エラーメールの発信アドレス) に対応しているか?
  - 対応していないと @@host になることがある
- list:; 形式に対応しているか?
  - 対応していないと list:;@host になる
- %-hack の解釈は正しいか?

sendmail.cf の準備

- OS 添付のものを修正して利用
- m4 cf 作成ツール
  - sendmail に添付
- mailconf
  - JUNET 時代に活躍
  - 静的配送ルールの生成が容易
- CF
  - sendmail R8 にも対応

CF による sendmail.cf の作成

- パッケージの入手
  - make cleantools; make tools が必要なときも
    - sed を使うとき、perl のパスが違うとき
- sendmail.def を記述
  - Standards/* をベースにする
- make sendmail.cf
  - sendmail.def から sendmail.cf を生成
- sendmail.cf のテストとインストール

.def の最低限の設定

- 当該ホスト名宛てのメールを受理
- MX を参照して配信
- CF_TYPE=R8V7
  - sendmail.8.8 用 V7 形式
- OS_TYPE=bsdos3.0
  - 使用する OS を指定
  - ファイルのパス、mailer の設定など
- BITNET=mx
  - user@node.BITNET への配信

ドメインマスタの設定

ACCEPT_ADDRS='x.co.jp'

- 受理すべきアドレス部 (これがポイント!)

FROM_ADDRESS='x.co.jp'

- 送信時のデフォルトのドメイン部
- 管理用アドレスにはホスト名が付与される
  - root, daemon, postmaster,...

- 複数ドメイン宛を受理
  - ACCEPT_ADDRS='sub1.co.jp sub2.x.co.jp'

NULL Client

- スプールを持たない
- 全てのメールをメールサーバへ
  - メールサーバのアドレスの定義が必要

CF_TYPE=R8V7-null

SPOOL_HOST=mail.x.co.jp

- メールサーバにだけ届くアドレスを記述
  - [] で囲む (lower MX があるときに A RR を参照)
  - IP アドレスを記述

PPPクライアント

FROM_ADDRESS=po.provider.ne.jp

DIRECT_DELIVER_DOMAINS=none

DEFAULT_RELAY=mail.provider.ne.jp

CON_EXP=True

SMTP_MAILER_FLAG_ADD=e

- **必ず** mqueue **にためる**
- **接続時に** sendmail -q **でまとめて配信**

PPPクライアントでの考慮点

- 発信者アドレスの書き換え
  - ローカルのユーザ名と契約のユーザ名
  - userdb, usertable の利用
- 契約していないアドレスからの発信の抑制
  - check_compat ルールセットの利用
- 自動ダイアルアップ時のタイムアウト

  O DialDelay=15s
- 受信は POP 等で

Firewallとメールサーバ(0/3)

- 外部NSと内部NSがある場合
  - 外部 NS には Wildcard MX を定義
    - * IN MX 10 ext-mail.x.co.jp.
      - ホストの存在を見せない
  - 外部 NS に、存在するメールアドレスをすべて定義
    - Wildcard MX を使わない

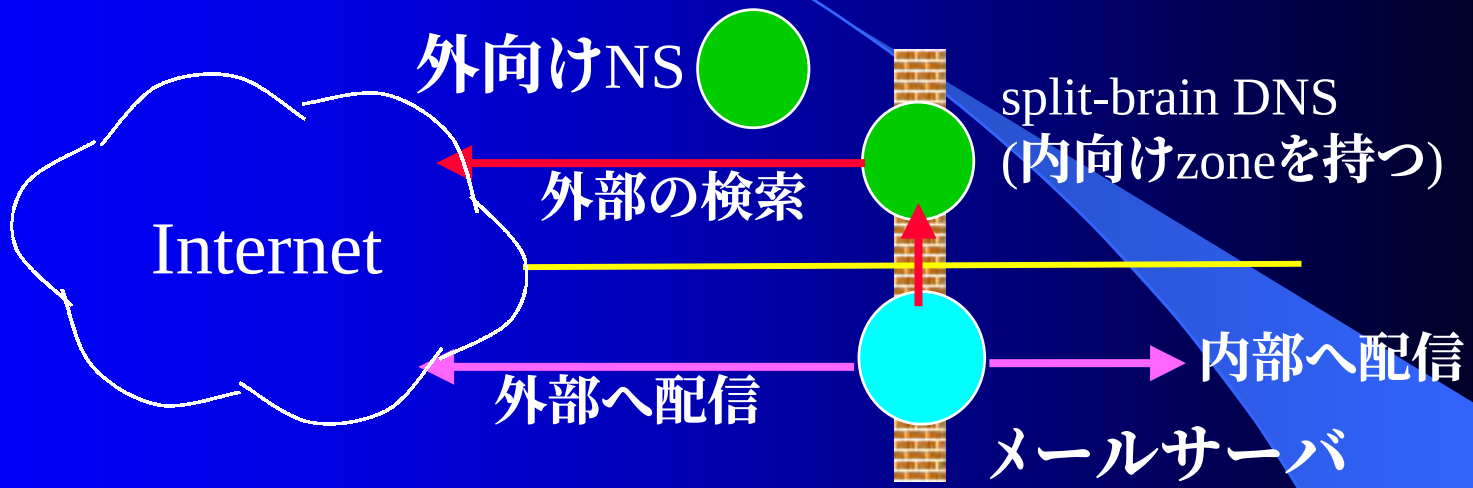
Firewallとメールサーバ(1/3)

ネームサーバとメールサーバの構成

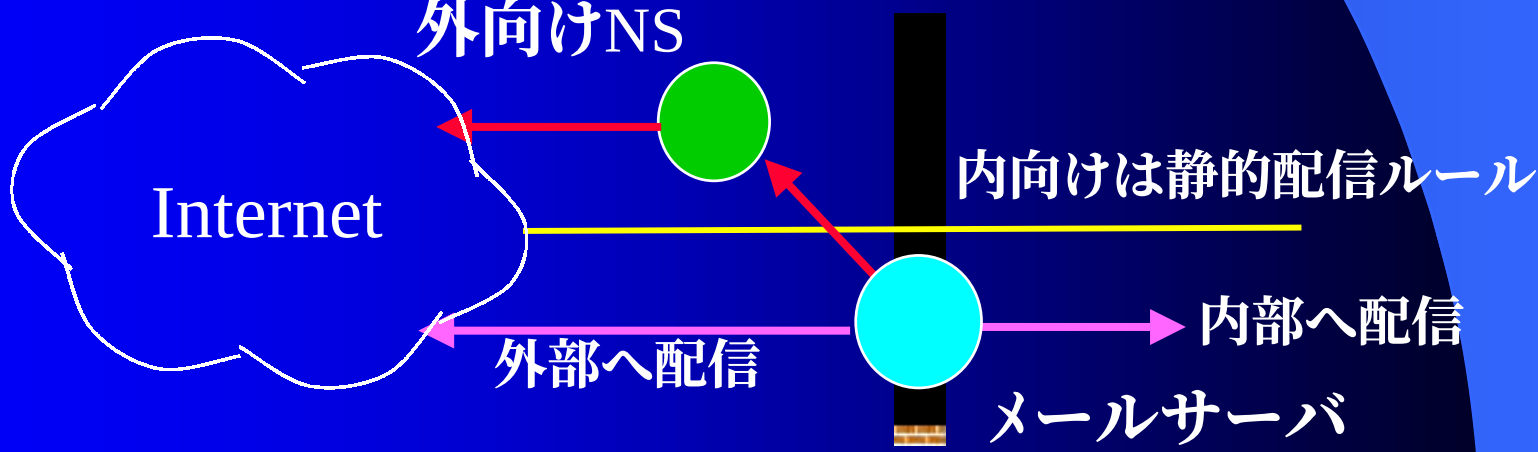
- メールサーバは1台
  a. 内向けzoneを持った外部検索用NSを参照
     - split-brain DNS
  b. 内部は固定ルールで配信

メールサーバ 1台

a


外向けNS
split-brain DNS
(内向けzoneを持つ)
Internet
外部の検索
外部へ配信
内部へ配信
メールサーバ

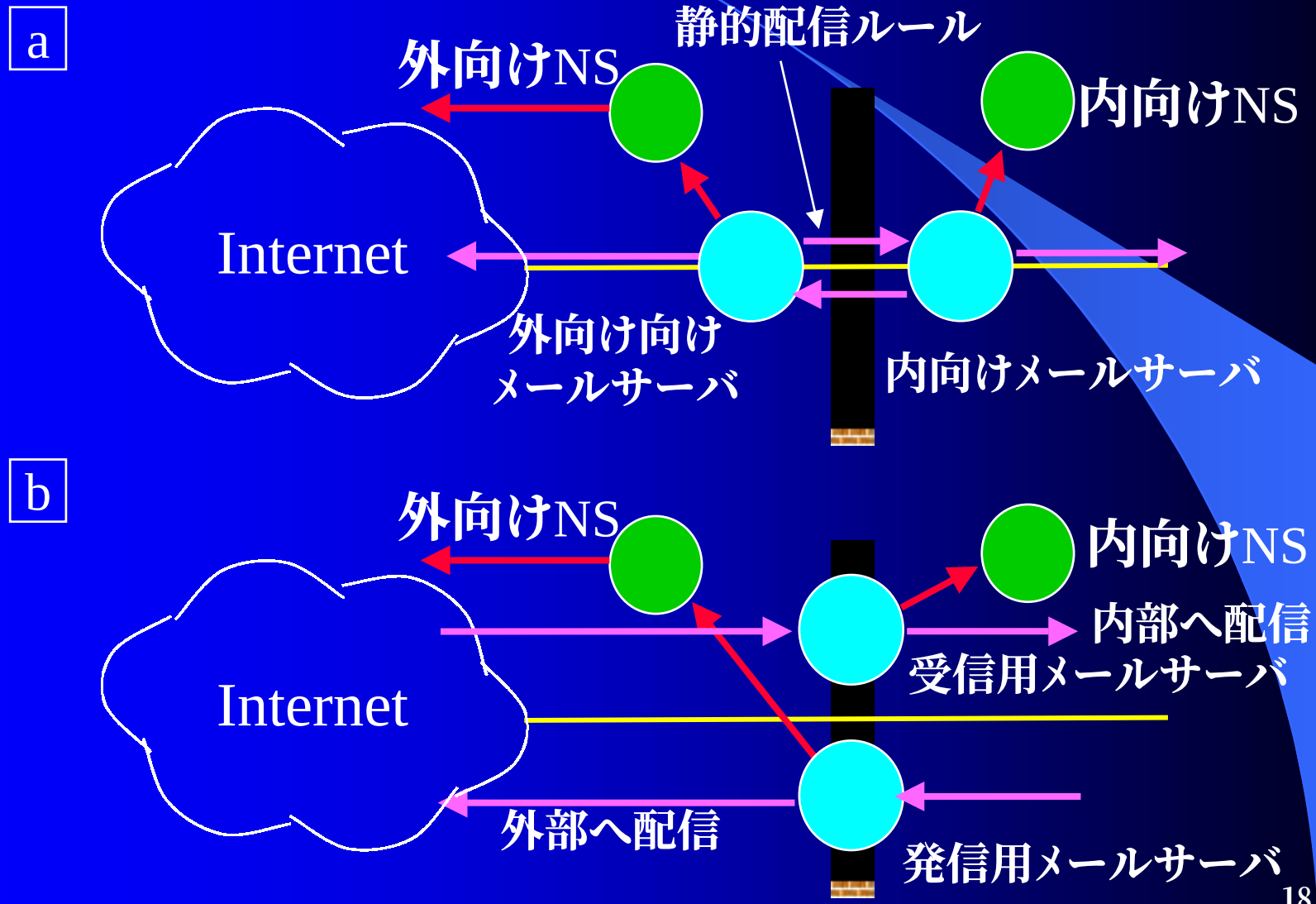
b


外向けNS
内向けは静的配信ルール
Internet
外部へ配信
内部へ配信
メールサーバ

Firewallとメールサーバ(2/3)

- メールサーバを2台
  - 外部DNSを参照するメールサーバ
  - 内部DNSを参照するメールサーバ
    - 方式 a
      - 両者間は静的経路設定
    - 方式 b
      - 受信専用メールサーバ
      - 発信専用メールサーバ

メールサーバ2台


a
静的配信ルール
外向けNS
内向けNS
Internet
外向け向け
メールサーバ
内向けメールサーバ
b
外向けNS
内向けNS
内部へ配信
受信用メールサーバ
Internet
外部へ配信
発信用メールサーバ

内部向けメールサーバ

- 外部への静的配信ルール

DIRECT_DELIVER_DOMAINS=x.co.jp

DEFAULT_RELAY=external.x.co.jp

  - 外部向けメールサーバ

外部向けメールサーバ

- **内部への静的配信ルール**

STATIC_ROUTE_FILE=x.static

x.static:

```
GW    [12.34.56.78]
#       (internal.x.co.jp)
DOM  x.co.jp
```

  - **メールサーバ宛てのメールは受理可能**

Firewallとメールサーバ(3.1/3)

- どちらも1台でなんとかする
  - a. 内側に first MX を向けておく

```
inner-host  IN  MX  10  inner-host
            IN  MX  20  gw
```

    - 1st-MX と直接通信できず、タイムアウトが発生

  - b. 内側は A RR を参照して配信

```
inner-host  IN  A  12.34.56.78
            IN  MX  10  gw
```

Firewallとメールサーバ(3.2/3)

c. 内側を別の枝にマップ

- inner.domain.jp → inner.domain.jp.local
  - sendmail.cf でアドレス変換
  - STATIC_ROUTE_FILE の MAP 行 (CF)

d. 1台に複数のデーモンを起動する

- IP アドレスにバインド
  - 外向け named と 内向け named
  - 外向け sendmail と内向け sendmail
    - O DaemonPortOptions=Address=12.34.56.78

NSを1台で(cont.)

- a, b 方式の問題
  - 内外の直接の通信は不可なのに
  - 外から内部ホストの情報が見える
    - bind8 の allow-query だけでは役不足
- a 方式の問題
  - 直接通信不可のホストに外部から接続を試みさせるべきではない
    - 1st MX が内側ホスト
    - 2nd MX がゲートウェイ
      - 余計なタイムアウト待ちの発生

NSを1台で(cont'd)

- b 方式の具体的設定
  - ゲートウェイから内部への配信
    - 静的経路定義
    - A RR を見る
      - 1st-MX が自分だった場合の挙動の変更

      TRY_NULL_MX_LIST=True (CF)

      O TryNullMXList=True (sendmail.cf)
  - 正しく設定できていないと

    local configuration error

ゲートウェイ内クライアント

- **なんでもGWへ**

  DIRECT_DELIVER_DOMAINS=none

  DEFAULT_RELAY=internal.x.co.jp

- **内部直接配送**

  DIRECT_DELIVER_DOMAINS=x.co.jp

  DEFAULT_RELAY=internal.x.co.jp

- **スプールなし**

  – NULL Client

バックアップメールサーバ(cont.)

- 1st-MXの障害発生時に代わりに受信
  - 2nd-MX
- 2nd-MX から直接配信
  - aliases を共有
  - 同じアドレスを受理
    - 全てのユーザ

      ACCEPT_ADDRS=
    - 特定のユーザ

      SECONDARY_*=
      - 指定したもの以外は、1st-MX の回復を待つ

バックアップメールサーバ (cont'd)

- aliases を共有
  - NIS などの共有手段の利用
  - ローカル aliases と共有 aliases に分離
  - OA/etc/aliases, nis: mail.aliases
- ML のバックアップ
  - アーカイブ問題
  - 連番問題

バーチャル・ドメイン(cont.)

1台のホストで複数アドレスを利用

- ユーザ空間の共有

```
USERTABLE_MAPS='domain1=hash:/etc/map1 ¥
                domain2=hash:/etc/map2'
```

- ユーザ空間の分離(1)
  - 1つのホストに複数のIPアドレス
  - アドレスごとにsendmailを起動

```
O DaemonPortOptions=Address=1.2.3.4
```

  - chroot で環境を分離

バーチャル・ドメイン(cont'd)

- ユーザ空間の分離(2)
  - sendmail.cf でがんばる
    - アドレスごとに local mailer を切り替え
  - /etc/passwd とは別のデータベースを利用
  - POP 等専用

sendmail.cf のテスト

- 配信先の簡単なチェック
  - sendmail -bv
- ルールセットによるアドレス書換のチェック
  - sendmail -bt
- 配信のテスト
- /etc/sendmail.cf の置き換え
  - 古いものは残しておくこと

配信先の簡単なチェック

- sendmail -bv
  - 一般ユーザでチェックする場合は mqueue の場所を適当に指定(permission)

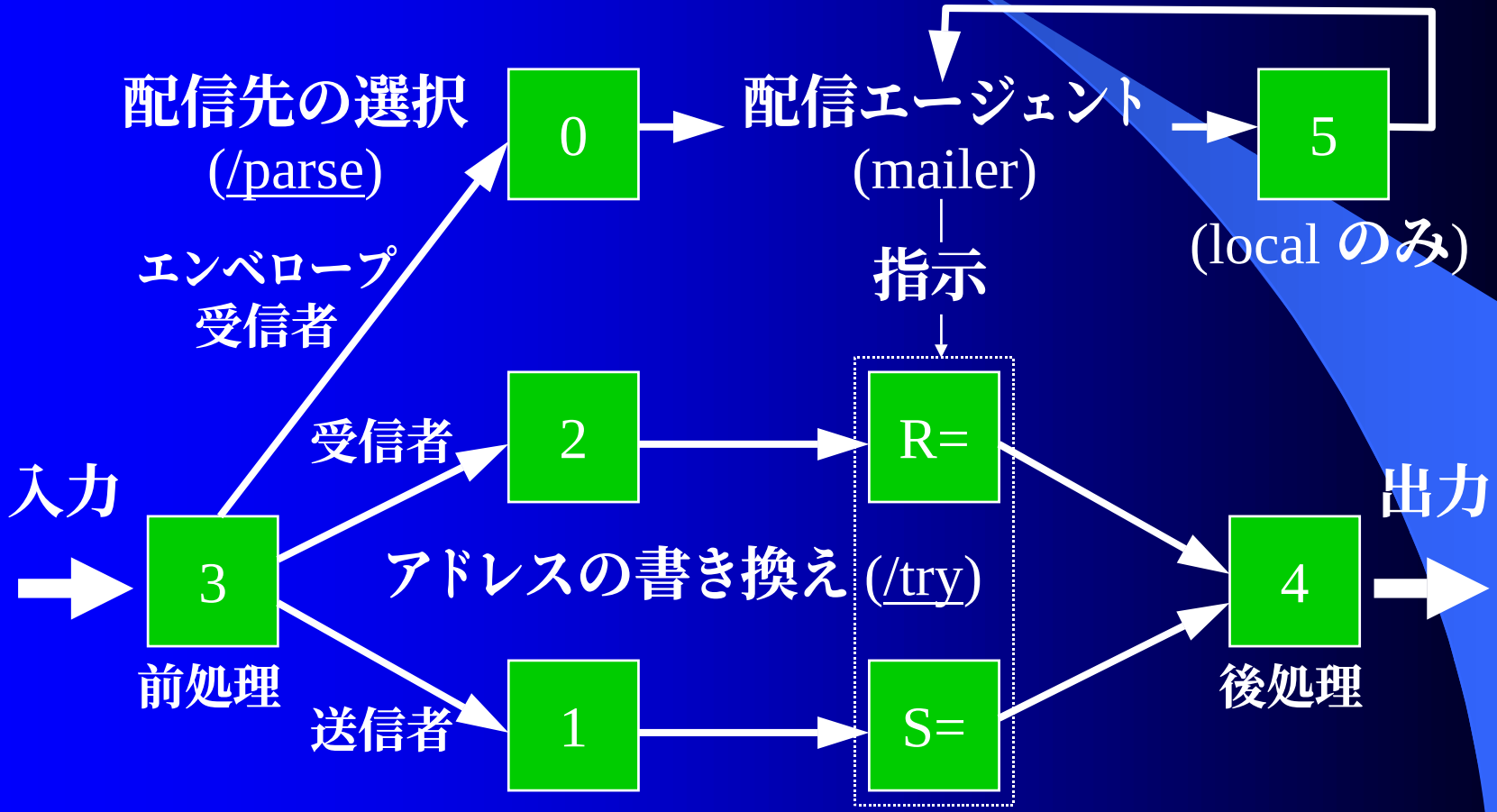
% sendmail -C new.cf -oQ/tmp -bv user@domain

motonori@wide.ad.jp... deliverable: mailer smtp, host wide.ad.jp., user motonori@wide.ad.jp

ルールセット・テストのポイント

- リモートのメールアドレス
  - 3,0 あるいは /parse、さらに 5
  - smtp 等のリモート配信用 mailer に渡されるか
- ローカルのメールアドレス
  - 3,0 あるいは /parse
  - local mailer に渡されるか (受理の確認)
- 期待するドメイン部が付与されるか
  - /try コマンド (ヘッダ、エンベロープのチェック)

ルールセット処理の流れ (アドレスの解析)



ルールセットのチェック(1)
リモートへの配信

- sendmail -bt

  % sendmail -C new.cf -bt

  > 3,0 motonori@wide.ad.jp

  rewrite: ruleset 3   input: motonori @ wide . ad . Jp

  :

  rewrite: ruleset 0 returns: $# smtp $@ wide . ad . jp. $:
  motonori < @ wide . ad . jp >

- R5 sendmail までは 0 address
- R8 sendmail からは 3,0 address (または /parse)

ルールセットのチェック(2)
ローカルへの配信

```
% sendmail -C new.cf -bt
> 3,0 motonori@wide.ad.jp
rewrite: ruleset   3   input: motonori @ wide . ad . Jp
 :
rewrite: ruleset   0 returns: $# local $: motonori
>
```

ルールセットのチェック(3)
ルールセット５

- ルールセット5
  - sendmail R8 より
  - 3,0 で local mailer が選択された場合
  - aliases チェックの後に適用

> 5 motonori

rewrite: ruleset　5　input: motonori

rewrite: ruleset　5 returns: $# smtp $@ spool $:
　motonori < @ spool >

ルールセットのチェック(4)
アドレスの書き換え

- ヘッダ･エンベロープの書き換えの確認

> /tryflags HS

> /try smtp motonori

Trying header sender address motonori for mailer smtp

rewrite: ruleset 3 input: motonori

:

rewrite: ruleset 4 returns: motonori @ wide . ad . jp

Rcode = 0, addr = motonori@wide.ad.jp

配信のテスト

% sendmail -C new.cf -oQ/tmp -v user@host
From: user@host
To: user@host

This is a test
.
%

- -oQ/tmp は一般ユーザでテストするときのみ必要

SMTP受信のテスト

```
# sendmail -C new.cf -bs
220 mail.wide.ad.jp ESMTP Sendmail 8.8.8
MAIL FROM:<motonori>
250 <motonori>... Sender ok
RCPT TO:<motonori>
250 <motonori>... Recipient ok
DATA
354 Enter mail, end with "." on a line by itself
test
.
250 TAA13313 Message accepted for delivery
QUIT
221 mail.wide.ad.jp closing connection
```

sendmail.cf の入れ替え(cont.)

- /etc/sendmail.cf にコピー
  (もしもの時のために古いものは残しておく)
- daemon sendmail の再起動

```
# ps aux | grep sendmail
  72  ??  .... accepting connections on port 25 (sendmail)
 195  ??  .... sendmail: OAA12345 mail.y.co.jp.: user open
# kill -HUP 72
```

  - SIGHUP が利用できるのは R8 sendmail から
  - R5 sendmail は殺して再起動

sendmail.cfの入れ替え(cont'd)

- 動作していなかった場合

  # /usr/lib/sendmail -bd -q1h

  - -bd
    - デーモン･モード(受信用)
  - -q1h (1時間) -q30m (30分)
    - mqueue に溜まったメールの配信再試行間隔
- メールを受信しないホスト
  - 受信用デーモンは不要

  # /usr/lib/sendmail -q1h

Sendmail.cf のまとめ

- どのアドレスを受理するか
- アドレスごとの配信方法の選択
  - 配信先を静的に定義
  - ネームサーバ (MX) を参照
- まずメール・システムのデザインを決める