

Linux によるネットワークサーバ 構築教材の紹介 第2回

ポリテクセンター中部 西尾 和彦
(中部職業能力開発促進センター)

5 . DNSサーバ

5.4 ネームサーバの動作確認

ネームサービスに必要な設定ファイルを作り終えたら、namedプログラムを起動し、名前解決に関する設定ファイルを確認した後、ネームサーバが正常に動作しているか次の手順で確認します。

```
namedデーモンの起動を確認
リゾルバの設定を確認
/etc/host.confの設定を確認
nslookupによるテスト
```

(1) namedデーモンの起動を確認

namedが起動しネームサービスが開始されているか確認します。次のようにpsコマンドを入力し、/usr/sbin/namedが表示されないようなら手動でnamedを起動します。

```
# ps -x | grep named
```

namedが起動している状態でいずれかの設定ファイルを変更した場合は、/usr/sbin/named.reloadスクリプトを使います。スクリプトを実行すると、しばらくした後“Reloading Database”のメッセージが表示され、変更が反映されます。

```
# named.reload
```

(2) リゾルバの設定を確認

ネームサービスを受けられるようにするには、それがネームサービスを行っているホストか、クライ

アントのホストかにかかわらず、リゾルバ(resolver)が設定されている必要があります。リゾルバはアプリケーションからの名前に関する問い合わせをネームサーバに送り、ネームサーバからの応答をアプリケーションに返送する役割を持ちます。

リゾルバの動作は/etc/resolv.confファイルによって制御されます。このファイルもnamedの設定ファイルと同じく、いくつかのディレクティブとパラメータを組み合わせた書式によって記述します。リスト8は、ネームサーバ(mercury)を1台だけ持つ、chubu.or.jpドメインにおける/etc/resolv.confの設定例です。

リスト8 /etc/resolv.confの例

```
.....
# /etc/resolv.conf
search chubu.or.jp
nameserver 192.168.5.151
.....
```

(3) host.conf の設定確認

クライアント側において設定を確認しておかなければならないもう1つのファイルが/etc/host.confです。/etc/host.confはホスト名の解決をどのように行うか設定するファイルであり、リスト9のように書いておきます。

リスト9 /etc/host.confの例

```
.....
order hosts , bind
multi on
.....
```

リスト10 nslookupの実行例

```
-----  
# nslookup                               ... nslookup を起動  
Default Server: mercury.chubu.or.jp     ... デフォルトサーバの情報  
Address: 192.168.5.151  
  
> 192.168.5.155                           ... IPアドレスの問い合わせ  
Default Server: mercury.chubu.or.jp  
Address: 192.168.5.151  
  
Name: jupiter.chubu.or.jp             ... ホストのFDQN  
Address: 192.168.5.155                 ... ホストのIPアドレス  
  
>  
-----
```

(4) nslookupによるテスト

nslookupとdigはネームサービスのテスト用プログラムです。特にnslookupはリゾルバをほぼ完璧にエミュレートできることから、ネームサーバのデバッグツールとして広く使われています。

nslookupは対話的にコマンドを入力しながら使うほか、パラメータを直接コマンドラインから渡して問い合わせの結果を得ることもできます。対話的に使う場合は、プログラムを起動（nslookupと入力）し、表示されたプロンプト“>”へ問い合わせたい名前やIPアドレスを入力します（リスト10）。

6. メールサーバの構築

メールサーバは、投函されたメールの転送と、このサーバへ投函または他のメールサーバから転送されてきたメールの蓄積および取り出しを自動化するサーバです。Slackwareには、インターネット上における最もポピュラーなメールシステムの実装であるsendmailとin.pop3dがパッケージされているので、学校や職場におけるメールサービスにLinuxマシンを利用することができます。また、PJEにはsendmailのセキュリティを強化するパッチを施したバージョンと、設定ファイルの作成を補助するCFというツールが収録されていますので、インストールしておきます。

6.1 sendmailの設定

sendmailのメール配送手順や具体的な動作設定は、

/etc/sendmail.cfファイルに記述します。ただし、書式や記法が難解なので、メールの配送方針に合わせていくつかの情報を与えることによってこのファイルを生成することができる、CFというツールを利用するのが一般的です。ここでは、chubu.or.jpドメインにおいて、以下の環境と方針でメールを処理することを想定し、CFを利用したsendmail.cfの作成手順を紹介します。

chubu.or.jpドメインにはネームサーバ（mercury.chubu.or.jp）が置かれており、sendmailはこのネームサーバのMX（Mail eXchanger）レコードを参照する。

sendmailはメールサーバ（jupiter.chubu.or.jp）上で動作しており、各ユーザのメールボックスはこのサーバ上に設ける。

ドメイン宛のメール（たとえば、nishio@chubu.or.jp）はメールサーバであるjupiter.chubu.or.jpへ配送する。

chubu.or.jpドメイン以外に宛てたメールは、ネームサーバの設定に従って配送される。

以上をまとめたメールの処理環境と配送方針の概要を図4に示します。

メールサーバの動作環境を把握しメールの配送に関する方針が決まったら、CFを利用したsendmail.cfの作成作業に入ります。PJEからインストールされたCF Version3.6は/usr/src/CF-3.6Wbeta6ディレクトリ下に置かれていますので、このディレクトリに移動して作業します。まず最初に必要なツールやサンプルの定義ファイルを得るために、いくつかの

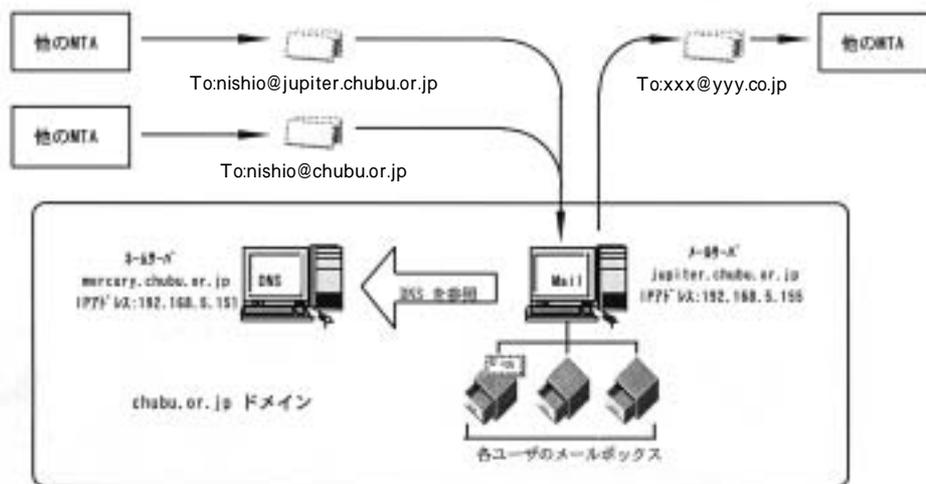


図4 chubu.or.jp ドメインのメール配送方針

リスト11 CFのmake

```

# cd /usr/src/CF-3.6Wbeta6
# make clean
# make cleantools
# make tools

```

makeを行います (リスト11)。

次に、このサーバで利用している sendmail のバージョンに合った雛形の定義ファイル (/usr/src/CF-3.6Wbeta6/* .def) を一つ選び、適当な名前で作成ディレクトリ (/usr/src/CF-3.6Wbeta6/) にコピーします。PJE-0.1.5cm には LAN 環境に合わせたサンプルファイル (pje.def) が収録されていますので、今回はこれをコピーして使います。このとき、コピーしたファイルの名前には必ず .def の拡張子をつけてください。

```
# cp pje.def sendmail.def
```

sendmail.def をメールサーバの環境と配送の方針に合わせて編集します。最終的に必要な行だけを抜粋した内容をリスト12に示します。

sendmail.def から sendmail.cf を生成します。ファイル名の拡張子部分 (.def) を .cf に置き換えて make してください。

```
# make sendmail.cf
```

リスト12 /usr/src/CF-3.6Wbeta6/sendmail.def

```

# sendmail.cf のタイプ
CF_TYPE=R8V7

# OS のタイプ
OS_TYPE=linux

# DNS の MX レコードを参照する
MX_SENDMAIL=yes

# このドメインの名前 ($m マクロ)
MY_DOMAIN=chubu.or.jp

# 自ホスト宛のすべてのメールを SMTP で直接配送する
DIRECT_DELIVER_DOMAIN=yes

# サーバ名を省略した ユーザ名@ドメイン名 宛のメールを受け付ける
ACCEPT_ADDRS='$m'

# ユーザ名@サーバ名.ドメイン名 宛のメールを受け付ける
ACCEPT_LOWER=yes

# このドメインから発信されるメールのすべての From:ヘッダを
# ユーザ名@ドメイン名 に置き換える
FROM_ADDRESS='$m'
REWRITE_GENERIC_FROM=lower

# 同一ホストのユーザ間におけるメールの To:/From:ヘッダに
# @ドメイン名 を付ける
ALWAYS_APPEND_DOMAIN=yes

# ドメイン名 が省略されているメールの To:/Cc:ヘッダに
# @ドメイン名 を付加する
RECIPIENT_GENERIC=yes
REWRITE_GENERIC_TO=lower

# 自ドメイン宛または自ドメインから発信されたメールだけを
# 取り扱う
MAIL_RELAY_RESTRICTION=yes
LOCAL_HOST_DOMAIN='$m'
ALLOW_RECIPIENT_DOMAIN='$m'

```

リスト13 sendmail.cfファイルのコピーとデーモンの再起動

```
# cp sendmail.cf /etc
#
# ps -x | grep sendmail
656 ? S 0:00 sendmail: accepting connection on port 25 0:00
837 pl S 0:00 grep sendmail
#
# kill -HUP 656
```

makeが終わると/usr/src/CF-3.6Wbeta6/ディレクトリにsendmail.cfができあがっているはず。sendmailプログラムが設定ファイルを探すデフォルトパスの/etcディレクトリへこれをコピーした後、動作設定の変更を反映するために sendmailを再起動します(リスト13)。

6.2 DNSの設定(MXレコード)

ここで構築しているメールサーバは自ドメインのネームサーバを参照してメールの配送先を決めます。したがって、メールサーバが参照するこのドメインの権威あるネームサーバには、ドメイン内のメールを集約するホストの名前と優先順位、およびホストのIPアドレスが定義されている必要があります(リスト14)。

6.3 POPの設定

各自のメールボックスへ収められたメールは、そ

リスト14 ネームサーバの順引きレコード(部分)

```
jupiter IN A 192.168.5.155
chubu.or.jp. IN MX 0 jupiter.chubu.or.jp.
```

のメールサーバへログインすれば読むことができます。ただし、クライアントのMacintoshやWindowsマシン上のメールソフトへはSMTPで直接転送することができないので、読むことができません。ここで登場するのがPOP(PostOfficeProtocol)の実装であるin.pop3dです。POPはMTAを持たない(つまりSMTPでメールを転送してもらえない)ホストがメールサーバからメールを取り出すために使われるプロトコルであり、sendmail(MTA)とは無関係に機能します(図5)。

Slackwareのsetupプログラムを利用した場合、パッケージシリーズNを選択することにより、POP3の実装であるin.pop3dが/usr/sbinディレクトリにインストールされ、起動設定(/etc/inetd.conf)も終わっています。

6.4 メールサービスのテスト

sendmailとネームサーバが構成できたら、sendmailがMTAとして正常に動作するかmailコマンドを使ってテストします。mailはメールの発信と受け取りに最低限必要な機能を備えたコマンドであり、ほとんどのUNIX系OSが標準で持っています。また、POPについてはmailコマンドではテストで

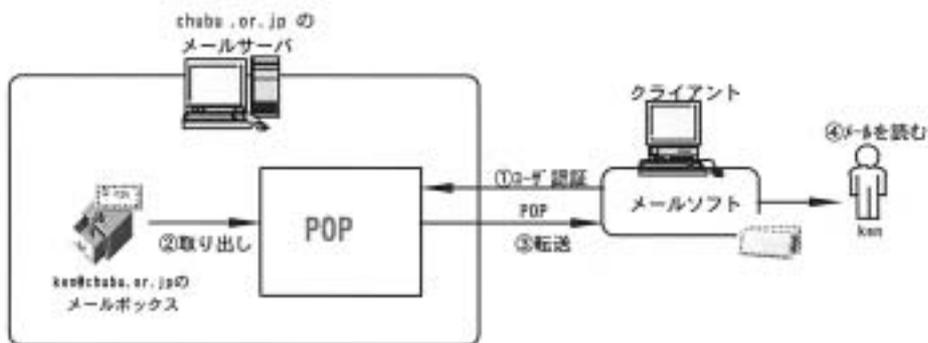


図5 POPによるメールの取り出し

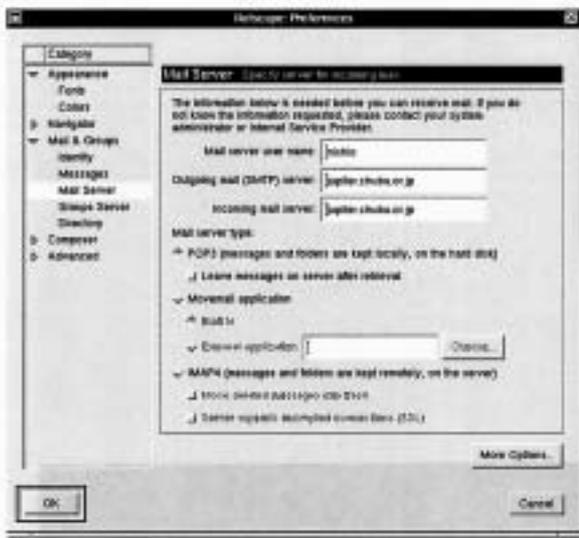


図6 NetscapeCommunicatorのメール設定例

きないので、メールの取り出しにPOPを利用するNetscapeCommunicator等を使ってテストします(図6)。

7. Webサーバ

インターネットの火付け役ともなったWWW (World Wide Web) は、従来の外部へ向けた情報発信だけでなく、イントラネットやグループウェア的な環境を構築するインターフェースとして利用される事例が増えています。ここでは、Slackwareに収録されているWebサーバパッケージであるapacheを設定し、Linuxマシン上でWWWの基本的なサービスを展開する手順を紹介します。

7.1 apacheの起動スクリプト

Slackwareのsetupプログラムは、HTTPサーバプログラム(httpd)の起動スクリプトを/etc/rc.d/rc.httpdに作成し、これを/etc/rc.d/rc.Mから実行するように設定します。また、これらのスクリプトはシステムの起動時に実行されるため、特に調整する必要はありません。

7.2 ネームサーバの設定

Webサーバへ接続するユーザの利便性に配慮し、

このサーバへwww.DomainNameの名前でアクセスできるようにネームサービスを設定します。リスト15はchubu.or.jpのプライマリネームサーバが持つ順引きファイル(たとえば/etc/namedb/named.hosts)に、Webサーバであるmercuryの別名をwwwとして登録した例です。

リスト15 namedの順引きファイルへWebサーバ用の別名を定義した例

```
www      IN      CNAME   mercury.chubu.or.jp.
```

7.3 apacheの設定ファイル

apacheは起動時にhttpd.conf、srm.conf、およびaccess.confの3つのファイルを参照します。httpdの起動時に-dオプションを使ってディレクトリを指定していなければ、これらのファイルは/var/lib/httpd/confディレクトリに収められます。また、このディレクトリには3つのファイルの雛形であるhttpd.conf-dist、srm.conf-dist、access.conf-distも置かれています。各々のファイルはhttpdプログラムによって既定されるディレクティブとパラメータによって記述する必要があり、httpdの動作条件、アクセス制御、およびHTMLファイルの置場所等を設定します。表3に各々のファイルがどのような目的に使われるのかを示します。

Slackwareのインストール時に作成されるデフォルトの設定ファイル群は、インストール直後のapacheの状態に合わせてあり、標準的なWebサイトの運用に近い形で設定されています。したがって特に理由がある他は、次節以降に示すディレクティブの設定を見直せば良いでしょう。

表3 apacheが参照する設定ファイル (/var/lib/httpd/confディレクトリ)

ファイル名	機能
httpd.conf	apacheの基本的な動作条件を記述します。
srm.conf	Webサイトの入口にあたるHTMLファイルや、アカウントユーザのホームページに関する設定を記述します。
access.conf	ディレクトリのアクセス制御と意味づけを設定します。

7.4 管理者のメールアドレス

エラーメッセージ等に埋め込まれる、管理者のメールアドレスをhttpd.confのServerAdminディレクティブに設定します。

```
ServerAdmin nishio@chubu.or.jp
```

7.5 Webサイトのルートディレクトリ

開設したWebサイトの入口となるHTMLファイルの置場所をsrm.confのDocumentRootディレクティブに設定します。たとえば、/home/wwwディレクトリを作成しsrm.confに以下のDocumentRootディレクティブを書いた場合、

```
http://www.chubu.or.jp/index.html
```

でアクセスされたドキュメントの実体は、/home/www/index.htmlになります。

```
DocumentRoot /home/www
```

srm.confを変更した後、このWebサイトのルートディレクトリが正しく設定できたか、以下の手順で確認します。

httpdを再起動する。

テスト用のHTMLファイル(index.html)をドキュメントルートで設定したディレクトリ/home/wwwに作る。

WebブラウザへサーバのURLを入力し、テスト用のページが正しく表示されるか確認する。

7.6 ユーザ用のディレクトリ名とファイル

サーバにアカウントを持つユーザは、自身のホームディレクトリ内にコンテンツを置き、これをWeb上へ開示することができます。たとえば、chubu.or.jpのWebサーバにアカウントを持つユーザ(nishio)のホームページへアクセスするには、以下のように`~/UserName/`を付加したURLを入力します。

```
http://www.chubu.or.jp/nishio/
```

各ユーザは、自身のホームディレクトリに開示するHTMLファイルや画像ファイルを置きます。ただし、ホームディレクトリの直下にコンテンツを置

くのではなく、Webへ開示するファイル群を収めた専用のディレクトリを設けます。各ユーザが開示するホームページ用のファイルを収めるディレクトリと、末尾が“/”で終わるURLでアクセスされたときに最初に読まれるファイルの名前は、srm.confの中で設定します。

ユーザのWeb用ホームディレクトリの名前

```
UserDir public_html
```

最初に読み込まれるファイル

```
DirectoryIndex index.html
```

この例では、各ユーザは自身のホームディレクトリ下にpublic_htmlディレクトリを作り、その中にindex.htmlを置くことによって、`~/UserName/`のアクセスを、`~/public_html/index.html`へ導くことができます。ただし、Slackware3.6.0のadduserユーティリティを使って作ったユーザのホームディレクトリには、オーナーのみr+w+xのパーミッションが設定されています。したがって、ここをWWW用に使うのなら、次のように操作して、その他のユーザにもr+xを与える必要があります。

```
# chmod o+r+x /home/nishio
```

