

# 第3回「ネットワークソフトウェアの設定と起動」

技術部ネットワークグループ 山本好弘

## はじめに

第3回では、インターネットで用いられるサーバソフトの内、各部及び研究室等の規模で主に使用されると思われる、ネーム、メール及びWWWサーバソフトウェアについての紹介と基本的な設定を解説する。また、インターネットで用いられるIP（インターネットプロトコル）についても簡単な解説を行う。

図1に今回のサーバソフトウェアの設定時に参照する仮定のネットワークを示す。実習用サーバ機（tech：192.168.2.1）及びWindowsクライアント機（tech21：192.168.2.10）はネットワークケーブルでHUBと接続されネットワークを構成し、さらにルータを経由して上位のネットワークと接続される。実習用サーバ機が接続されているネットワークのドメイン名はeng.mie-u.ac.jpとし、上位のネットワークのドメイン名はmie-u.ac.jpとする。また、上位のネットワーク(mie-u.ac.jp)にもDNS、Mailサーバ（192.168.1.1）が存在する。

今回使用するソフトウェアはRPM形式のソフトウェアパッケージを用いてインストールを行うこととし、ソースファイルからのインストールについては別の機会に紹介する。

## IP（インターネットプロトコル）

インターネットの歴史、パケット（データ）の構造、プロトコル及びIPアドレスについて解説する。

## ネームサーバ

ドメインネーム、インターネットにおけるドメイン参照の方法を解説し、次にネームサーバの設定について解説する。ネームサーバソフトウェアはBIND8.8.2P5-2を用いる。

BINDで使用する設定ファイルにはIPアドレスとドメイン名とが対応するゾーンデータベースファイルとそのファイルを管理する初期設定ファイルがある。以下に今回使用する設定ファイルを示す。

/named.conf：初期設定ファイル

named.ca：ルートサーバリストファイル

named.local：ゾーンデータベースファイル

named/eng：ゾーンデータベースファイル

実際の作業ではゾーンデータベースファイルeng（正引き）、eng.rev（逆引き）の設定がメインとなる。

## メールサーバ

インターネット上でのメール配信の仕組みを解説し、次にメールサーバの設定について解説する。メールサーバソフトウェアはsendmail-8.9.3-20を使用する。Sendmailの設定ファイルはsendmail.cfであるが、設定を記述するための記述内容がわかりにくいいため、これを簡単に行うためのソフトウェアが別途開発されている。各種ある中で今回はCF-3.7Wpl2-3を用いて設定を行うこととする。

Sendmail以外にもローカルのメール配信を行うprocmailを用い、また各メール配信を制御するaliasesの設定等もあわせて解説する。

## WWWサーバ

インターネット上で最も利用されているWWWサーバ歴史、プロトコル及びコンテンツ配信の仕組みの解説し、次にWWWサーバの設定等について解説する。WWWサーバソフトウェアはapache-1.3.12-5を使用する。

設定ファイルはhttpd.confであり、最近のバージョンより複数あった設定ファイルが統合されている。また、この他に各ファイルのファイルタイプの管理を行うmime.types及びmagicの紹介も行う。

また、コンテンツファイルの取り扱い等も簡単に紹介する。

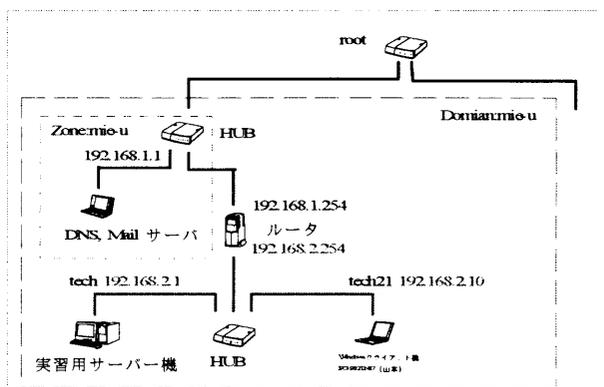


図1 ネットワークの構成

工学部技術部 平成12年度技術講習会

「Linuxを用いたインターネットサーバの構築」  
 - PCの組み立てからサーバの運用まで -

### 第3回 「ネットワークソフトウェアの 設定と起動」

技術部ネットワークグループ  
山本好弘

## プログラム

- インターネットの仕組み
- DNS(Domain Name System)サーバの構築
- メールサーバの構築
- WWW(World Wide Web)サーバの構築

## ネットワークの歴史

### インターネットの歴史

- ARPANET(1969~)  
ARPA(米高等研究計画局)の軍事的な研究  
完全分散型アーキテクチャ  
パケット交換方式
- ↓
- TCP/IPプロトコル(1982)
- ↓
- the Internetへと発展(現在)

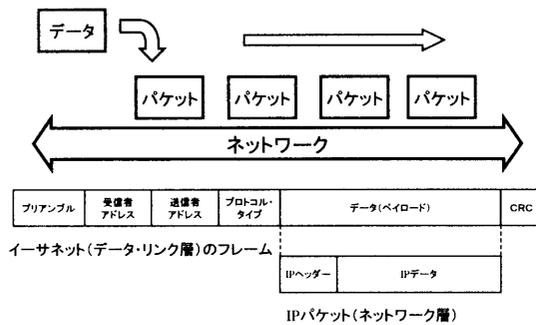
### イーサネットの歴史

- ALOHAシステム(1968~1972)  
ハワイ大学の無線通信システム
- ↓
- イーサネット(1970年代)  
Xeroxのパロアイト研究所(PARC)が同軸ケーブルで実現
- ↓
- IEEE 802.3として標準化(1978)

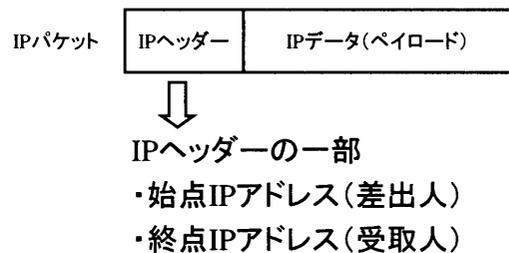
## ISOの参照モデル

|            |   |                                   |          |
|------------|---|-----------------------------------|----------|
| アプリケーション層  | アプリケーション・レベルでの通信プロトコルについての規程                                | TELNET, FTP, DNS, SMTP, POP, HTTP |          |
| プレゼンテーション層 | 送受信するデータの形式や文字コードについての規程                                    |                                   |          |
| セッション層     | セッション・レベルでのプロトコルについての規程                                     | Socket                            |          |
| トランスポート層   | プロセス間通信についての規程  | TCP                               |          |
| ネットワーク層    | ネットワーク上に接続された任意の2ノード間でのデータ転送プロトコルを規程                        | IP                                | ルータ      |
| データ・リンク層   | データのハケット化の方法と送受信プロトコルに関する規程                                 |                                   | ブリッジ     |
| 物理層        | EthernetやRS-232Cなど物理的な媒体の電気的なインターフェース及び基本的なデータの変調方法などについて規程 | イーサネット, ISDN                      | リピータ(ハブ) |

## ネットワークでの送信形態



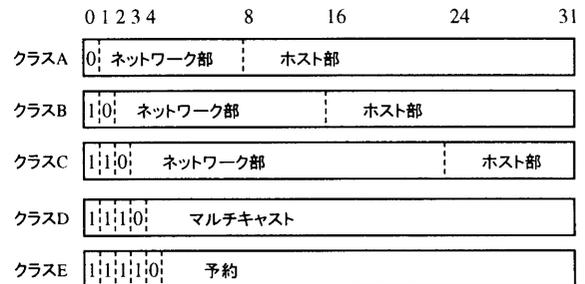
## IPv4(Internet Protocol version 4)



## IPアドレス

- 各ホストに割り当てられるアドレス  
グローバルなネットワーク上では唯一
- 32ビット(IPv4)で表す
- ネットワーク部とホスト部に分けられる  
ネットワーク部: インターネット上の個々のネットワークの識別  
ホスト部: ネットワーク中に存在するホストを識別
- ネットワーク部はNIC(Network Information Center)と呼ばれる組織で一元管理  
日本ではJPNICが管理

## IPアドレスとクラス



## 特殊な意味を持つIPアドレス

- ホスト部がすべて0  
ネットワーク自身を表す  
実際の通信には使用しない
- ホスト部がすべて1  
ネットワーク上に存在するすべてのホストを表す  
ブロードキャスト(送信者以外は全受信者)に使用
- 第1オクテットが127(127.0.0.0)  
同一ホスト上の通信に使用(ループバック)  
通常はホスト部を表す1を付加(127.0.0.1)

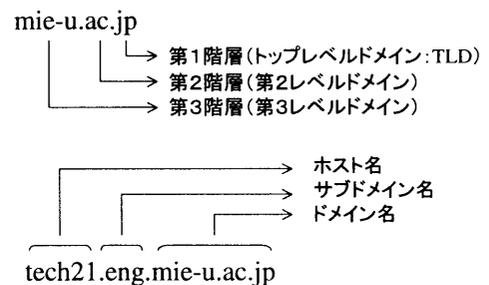
## プライベートアドレス

- 独自の閉じたネットワーク内で自由に使えるIPアドレス
- クラスA 10.0.0.0 ~ 10.255.255.255
- クラスB 172.16.0.0 ~ 172.31.255.255
- クラスC 192.168.0.0 ~ 192.168.255.255

## ドメイン名とは?

- IPアドレスで表記されたホスト(コンピュータ)を人間に分かり易く識別する手段
- あるホストが属しているネットワークを階層的な名前構造で表す
- ネットワークを区分する名前をドメイン名と呼ぶ

## 日本でのドメイン名の構造



## 名前解決の方法

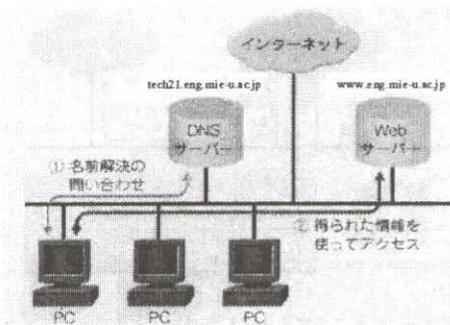


- UNIX(互換OS)  
 /etc/hosts ファイルに記述  
 127.0.0.1 localhost  
 192.168.2.1 tech21.eng.mie-u.ac.jp tech21
- インターネット上のすべてのホストを記述しなければならない
- DNS(Domain Name System)  
 南カリフォルニア大学で1984年に発表された仕組み

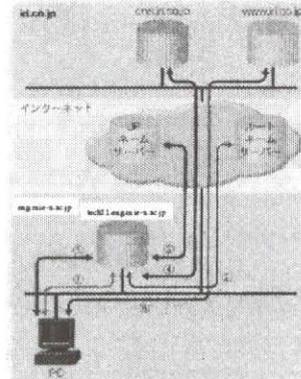
## DNS(Domain Name System)

- DNSサーバに名前解決のデータベース(ホスト名とIPアドレスの変換テーブル)を持たせる
- インターネット全体では分散データベースとして機能する

## 自ネットワーク内における名前解決

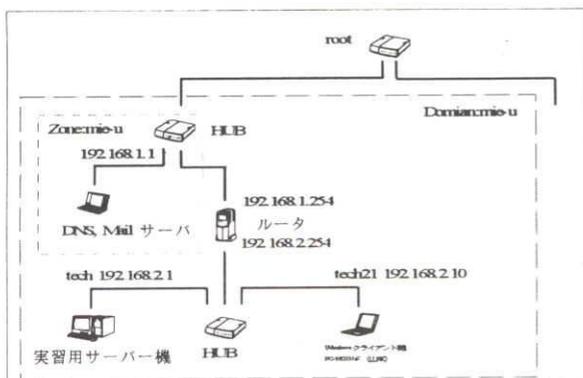


## インターネットにおける名前解決



## DNSサーバの構築

- 使用するソフトウェア(パッケージ)  
 bind 8.8.2P5-2  
 bind-utils 8.8.2P5-2  
 caching-nameserver 6.0-2
- 設定ファイルの設定
- 起動と設定(動作)の確認  
 nslookupコマンド
- サーバの設定変更



## ソフトウェアのインストール

- ソースファイルをコンパイルする  
付属のドキュメントに従い自分のシステムに適合するようにコンパイル
- RPM(Redhat Package Management system)  
システムに適合したコンパイル済みのソフトウェア(パッケージ)を扱える
- turbopkg  
RPMをCGI化したもの  
CD, ftpなどよりまとめてアップグレードするのに便利

## RPM(Redhat Package Management system)

- rpm -コマンド ソフトウェア(パッケージ)名
- 基本的なコマンド
  - i インストール
  - e 削除
  - U アップグレード
  - q クエリー(インストールされたソフトウェアの情報を表示)

## 設定ファイルの種類

- 初期設定ファイル  
/etc/named.conf
- ルートサーバリストファイル  
/var/named/named.ca
- ゾーンデータベースファイル  
/var/named/named.local  
/var/named/eng ネットワーク構成に合わせて作成

## 初期設定ファイル named.conf

- namedの起動時の振る舞いを制御
- ステートメント(制御命令)で構成  
acl, include, key, logging, options, server, zoon
- 通常はoptions, zoonステートメントで記述  
option:DNSサーバ全体のオプションを記述  
1回だけ記述可能  
zoon:ドメインごとに管理されたデータを指定

## named.confのデフォルトの設定

```
options {
    directory "/var/named";
    // query-source address * port 53;
};

//
// a caching only nameserver config
//
zone "." {
    type hint;
    file "named.ca";
};

zone "0.0.127.in-addr.arpa" {
    type master;
    file "named.local";
};
```

## named.confの設定

```
zone "eng.mie-u.ac.jp" {
    type master;
    file "eng";
};

zone "2.168.192.in-addr.arpa" {
    type master;
    file "eng.rev";
};
```

## ルートサーバリストファイル named.ca

- namedプログラム(BIND)に対してルートゾーンにサーバの情報を与える
- 13個のルートサーバを登録  
負荷の分散(各国)で安全性を高める
- 内容が変更されたら最新のファイルに置きかえる  
ただし更新は希である(1997年8月22日)  
ftp://ftp.nic.ad.jp/internic/rs/domain/named.root  
ftp://ftp.rs.internic.net/domain/named.cache

## ゾーンデータベースファイル 正引き(eng)

```
@ IN SOA tech.eng.mie-u.ac.jp. root.tech.eng.mie-u.ac.jp. (
    2000091200 ; Serial
    28800 ; Refresh
    14400 ; Retry
    3600000 ; Expire
    86400 ) ; Minimum

IN A 192.168.2.1
IN NS tech.eng.mie-u.ac.jp.

localhost IN A 127.0.0.1

loghost IN CNAME localhost

tech IN A 192.168.2.1

tech21 IN A 192.168.2.10
```

## ゾーンデータベースファイル 逆引き(eng.rev)

```
@ IN SOA tech.eng.mie-u.ac.jp. root.tech.eng.mie-u.ac.jp. (
    2000091200 ; Serial
    28800 ; Refresh
    14400 ; Retry
    3600000 ; Expire
    86400 ) ; Minimum

IN NS tech.eng.mie-u.ac.jp.

IN PTR eng.mie-u.ac.jp

IN A 255.255.255.0

1 IN PTR tech.eng.mie-u.ac.jp.

10 IN PTR tech21.eng.mie-u.ac.jp.
```

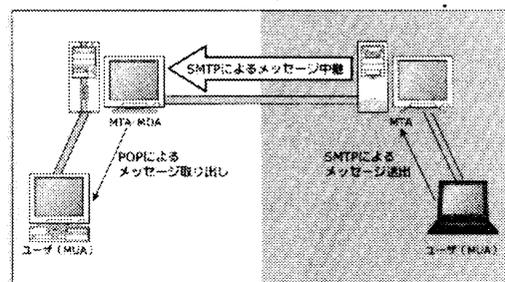
## 起動と動作確認と設定の追加

- bindの起動  
/etc/rc.d/init.d/named start
- nslookupコマンドによる確認  
[root@tech log]# nslookup tech21.eng.mie-u.ac.jp  
Server: tech.eng.mie-u.ac.jp  
Address: 192.168.2.1  
Name: tech21.eng.mie-u.ac.jp  
Address: 192.168.2.10
- /etc/resolv.conf にDNSサーバを記述  
nameserver 192.168.2.1

## メールサービスの概要

- MTA(Mail Transfer Agent)  
メールメッセージの配送を受け持つプログラム  
他のMTAへのメッセージの転送  
MDAに自ホストへの配信(格納)を依頼
- MDA(Mail Deliver Agent)  
メールメッセージの配送を受け持つプログラム  
メールメッセージをメールスプールに格納(配信)
- MUA  
ユーザがメールメッセージを作成・整理したり、それを受信したりするプログラム  
メッセージの配信をMTAに依頼  
メールスプールからメッセージを引き出す

## メールの流れ



SMTP(Simple Mail Transfer Protocol)  
POP(Post Office Protocol)

## sendmailの構築手順

- 使用するソフトウェア(パッケージ)  
sendmail-8.9.3-20  
CF-3.7Wpl2-3
- 設定ファイルの編集、作成  
sendmail.def → sendmail.cf  
aliases → aliases.db
- 動作確認
- 起動

## メールサーバの仕様

- ホストtech.eng.mie-u.ac.jpは、Mail Exchangerとして動作し外部からのメールを受け取る
- ホストtech.eng.mie-u.ac.jpはスプールホストも兼ねており、外部から自ドメインあてのメールをすべて受け取り、自らのメールスプールに格納する
- ドメインeng.mie-u.ac.jp内のホストは、すべてホストtech.eng.mie-u.ac.jpにPOPにてアクセスしメールを取得する
- eng.mie-u.ac.jpドメインから外部へのメールは、外部のMail Exchangerを探し、すべて直接配送する
- メール「From:」フィールドには、ホスト名が削除された「user@eng.mie-u.ac.jp」の形式でアドレスが設定される

## 設定ファイルの編集、作成

- sendmailのバージョンに合致した雛形ファイルをsendmail.defのファイル名で作成
- メールサーバの仕様を実現できるようにsendmail.defファイルを編集する
- CFを用いてsendmail.defファイルよりsendmailの設定ファイルであるsendmail.cfを生成する

## 作業手順

- /usr/src/CF-3.7Wpl2/Standards/ディレクトリにあるsendmail-v8.defファイルをsendmail.defとして作成する
- エディタでsendmail.defファイルを編集する
- makeコマンドでsendmail.cf ファイルを作成
- /etc/mail/aliasesファイルを必要に応じ編集する
- newaliasesコマンドでaliasesからデータベースファイルaliases.dbを作成する

## sendmail.defの設定

```
CF_TYPE=R8V8
VERSION=3.7W
VERSION_SEPARATOR=.
LOCAL_VERSION=MailExchanger
OS_TYPE=linux
MY_DOMAIN=eng.mie-u.ac.jp
MY_NAME=tech
OFFICIAL_NAME=$w.$m
MY_ALIAS=/etc/mail/aliases
FROM_ADDRESS=$m
RECIPIENT_GENERIC=yes
REWRITE_GENERIC_FROM=yes
REWRITE_GENERIC_TO=yes
ACCEPT_ADDR=$m
PROCMAIL_MAILER_PATH=/usr/local/bin/procmail
PMLocal_MAILER_ARGS=procmail -Y -a $h -d $u
LOCALDELIVER_PROCMail=yes
ALIAS_FILE_PATH=/etc/mail/aliases
COPY_ERRORS_TO=postmaster
MAIL_RELAY_RESTRICTION=yes
```

## aliasesファイルの編集

メーリングリストの作成  
/etc/mail/aliasesを編集

以下の行を追加  
ml: :include:/etc/mail/list

リストファイルを記述する  
(メンバーのメールアドレスを登録)  
/etc/mail/list

## Sendmailの動作確認と起動

- Sendmailをテストモードで起動し動作を確認  
/usr/sbin/sendmail -bt -C ./sendmail.cf  
/parse yama@tech21.eng.mie-u.ac.jp
- sendmailの起動  
動作確認のためスクリプトから起動  
/etc/rc.d/init.d/sendmail restart  
確認後turbo serviceでsendmailを登録

## DNSへの登録、POPサーバの起動

- DNSのゾーンファイル(eng)にMXを登録

```
IN      A           192.168.2.1
IN      NS          tech.eng.mie-u.ac.jp.
IN      MX          0           tech.eng.mie-u.ac.jp.
localhost IN A           127.0.0.1
loghost IN  CNAME       localhost
tech    IN  A           192.168.2.1
tech21  IN  A           192.168.2.10
```

- /etc/inetd.confの設定を変更の後、再起動

```
#
# Pop and imap mail services et al
#
#pop-2 stream tcp  nowait root  /usr/sbin/tcpd ipop2d
pop-3 stream tcp  nowait root  /usr/sbin/tcpd ipop3d (コメントアウトを取る)
#imap  stream tcp  nowait root  /usr/sbin/tcpd imapd
#
```

/etc/rc.d/init.d/inet restart (inetデーモンを再起動)

## World Wide Web

- 1989年に欧州合同素粒子原子核研究機構(Conseil European pour la Recherche Nucleaire)のTim Berners-Leeにより提案された広域情報システム
- 特徴
  - ①ハイパーテキストを介してインターネット上のすべての情報にアクセスすることが可能
  - ②インターネット上のすべての情報をシームレスに結合する統一された簡単な方法で、インターネット上のあらゆる種類の情報に一律にアクセスすることを可能にする
  - ③クライアント/サーバモデルに基づいている

## Webサーバ(httpd)の歴史

- CERN httpd(96年7月開発終了 Ver. 3.0A)  
スイスの欧州合同素粒子原子核研究機構(CERN)のTim Berners-Leeにより開発された世界初のサーバソフト
- NCSA httpd(最新 Ver. 1.5.2a)  
National Center for Supercomputing Applications(イリノイ大学)の学生Bob McCoolが'93年の始めごろに開発  
新機能(CGI, SSI)を搭載しCERNに替わり主流へ
- Apache httpd(最新 Ver. 1.3.12)  
NCSA Ver. 1.3の強化(機能の追加、バグの修正)版として95年にベータ版をリリース  
現在世界中のWebサーバの60%以上で稼動

## Webブラウザの歴史

- MOSAIC  
National Center for Supercomputing Applications(イリノイ大学)のMarc Andreessenにより93年に開発された世界初のブラウザ
- Netscape Navigator  
MOSAICの有料化に反対した開発スタッフがNetscape Communications社を興し、そこで開発されたブラウザ
- Internet Explorer  
Microsoft社がSpyglass社を経由して入手したMOSAICを基に開発したブラウザ

## WWWで用いられる基本技術

- HTML(Hyper Text Markup Language)  
ハイパーテキストを記述するための言語
- HTTP(Hyper Text Transfer Protocol)  
WWWクライアント/サーバ間のプロトコル
- URI(Universal Resource Identifier)  
さまざまなリソースに対する統一的な名前付けの手法

## Hyper Text Markup Language

- CERN(欧州合同素粒子原子核研究機構)で開発されたドキュメントシステム
- ドキュメントの中に他のドキュメントを参照できるドキュメントを埋め込むことでドキュメントの関連付けや閲覧の効率を向上

## Hyper Text Transfer Protocol

- アプリケーションレイヤ(OSI参照モデル)に相当する低負荷で高速なプロトコル
- 状態を保存しないオブジェクト指向型プロトコル
- メソッドの拡張により多くの用途に対応可能

HTMLのプロトコル

Method URI HTTP-Version  
例 GET /index.html HTTP/1.0

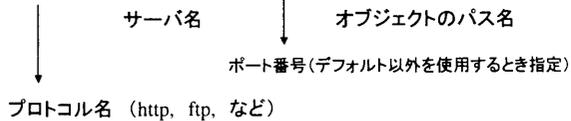
メソッド(HTTP/1.0)

GET URIで識別されたオブジェクトを取得  
HEAD URIで識別されたオブジェクトについてのメタ情報を取得  
POST サーバがURIで識別されるリソースに付属するオブジェクトを受け取る

## Uniform Resource Locator

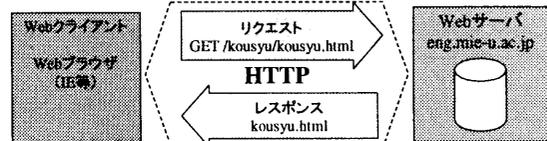
インターネット上におけるオブジェクト指定の形式

`http://www.mie-u.ac.jp:80/kousyu/kousyu.html`



## WWWの概要

URL `http://www.eng.mie-u.ac.jp/kousyu/kousyu.html`



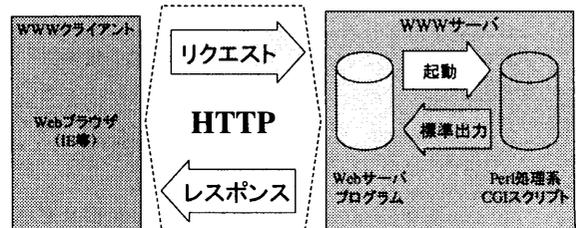
- ① WebブラウザからURLを入力すると、インターネット上のサーバを検索し、HTMLファイルを要求
- ② サーバ側はHTMLファイルを要求されると、その中身をインターネットを介してブラウザに送信
- ③ Webブラウザがデータを受け取り、HTMLの書式に従い表示する

## インタラクティブなWWW

- FORMタグ(HTML) ブラウザ/クライアント側からの入力が可能
- CGI(Common Gateway Interface) サーバ側にあるアプリケーションを実行しその結果をクライアントのブラウザに反映
- Javaアプレット オブジェクト指向言語 アプレット(実行モジュール)と呼ばれるアプリケーションをWebブラウザ内で動作
- JavaScript HTMLの機能を拡張するスクリプト HTMLの文書内にスクリプトを埋め込みクライアント側のブラウザで実行
- SSI(Server Side Includes) HTMLドキュメント内に記述 サーバ側で外部プログラムを実行し、その結果をクライアント側のブラウザの表示画面に反映させる

## CGIの概要

URL `http://www.eng.mie-u.ac.jp/kousyu/kousyu.cgi`



## Apacheの構築手順

- ソフトウェア(パッケージ)のインストール  
apache 1.3.12-5
- apacheの動作確認
- DNSサーバへの登録
- 設定ファイルの設定(編集)
- apacheの起動

## apacheの動作確認

- apacheの起動 `/etc/rc.d/init.d/httpd start`
- WWWブラウザより確認 `http://192.168.2.1/`



## DNSサーバへの登録

- DNSのゾーンファイル(eng)に登録

```
tech      IN      A       192.168.2.1
tech21    IN      A       192.168.2.10
```

```
www       IN      CNAME   tech
```

- WWWブラウザでの動作確認  
次のURLを用いて表示されるか確認する  
`http://tech.eng.mie-u.ac.jp`  
`http://www.eng.mie-u.ac.jp`

## 設定ファイル

- httpdの設定ファイル  
`/etc/httpd/conf/httpd.conf`  
注)以下のファイルで記述していた設定は現バージョンではhttpd.confに統一された/etc/httpd/conf/srm.conf  
`/etc/httpd/conf/access.conf`
- ファイルタイプの設定ファイル  
`/etc/httpd/conf/mime.types`  
`/etc/httpd/conf/magic`

## httpd.confの基本的な考え方

- どのようなホームページの公開を行うのか  
CGI等の動的コンテンツの導入の可否  
組織のトップページor個人のページ
- 誰がコンテンツ(htmlファイル)を管理するのか  
複数の限られた管理者or各個人
- セキュリティはどうするのか  
上記の形態の合わせて最良の方法を選択

## httpd.confの設定例

- 工学部や各学科のトップページのような組織が公開するようなホームページ
- htmlファイルの管理は複数の管理者が行う
- 最低限のセキュリティ対策を行う

## 各設定ファイルの位置を確認

- RPMのパッケージを使用する時は各設定ファイルの位置を確認する

```
rpm -ql apache
/etc/httpd/conf/access.conf
/etc/httpd/conf/httpd.bootopt
/etc/httpd/conf/httpd.conf
/etc/httpd/conf/magic
/etc/httpd/conf/mime.types
/etc/httpd/conf/srm.conf
/etc/logrotate.d/apache
/etc/rc.d/init.d/httpd
/var/log/httpd
/var/proxy
```

## httpd.conf

### Section 1: Global Environment

- HTTPサーバ全体に係わる設定  
RPMパッケージを使用しているときは特に変更の必要なし  
動作モードの指定  
ServerType standalone  
設定ファイルの場所を指定  
ServerRoot "/etc/httpd"  
PidFile /var/run/httpd.pid  
ScoreBoardFile /var/run/httpd.scoreboard  
接続時間の指定  
Timeout 300

## httpd.conf

### Section 1: Global Environment

- コネクションに関する設定  
KeepAlive On  
MaxKeepAliveRequests 100  
KeepAliveTimeout 15  
起動デーモンに関する設定  
MinSpareServers 5  
MaxSpareServers 10  
StartServers 5  
リクエスト数の設定  
MaxClients 150  
MaxRequestsPerChild

## httpd.conf

### Section 2: 'Main' server configuration

- Mainサーバに関する設定  
Port 80  
Listen 80  
Listen 443  
User nobody  
Group nobody  
ServerAdmin root@localhost  
DocumentRoot "/home/httpd/html"  
Options FollowSymLinks  
AllowOverride None  
Options Indexes FollowSymLinks MultiViews  
AllowOverride None  
Order allow,deny  
Allow from all

## httpd.conf

### Section 2: 'Main' server configuration

```
UserDir public_html
DirectoryIndex index.html
AccessFileName .htaccess
Order allow,deny
Deny from all
UseCanonicalName On
TypesConfig /etc/httpd/conf/mime.types
DefaultType text/plain
MIMEMagicFile /etc/httpd/conf/magic
HostnameLookups Off
```

## httpd.conf

### Section 2: 'Main' server configuration

```
ErrorLog /var/log/httpd/error_log
LogLevel warn
LogFormat "%h %l %u %t %r" "%s %b" "%{Referer}i" %
  {User-Agent}i"" combined
LogFormat "%h %l %u %t %r" "%s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
CustomLog /var/log/httpd/access_log common
```

httpd.conf  
### Section 3: Virtual Hosts

- バーチャルホストに関する設定を行う
- バーチャルホストとは1台のサーバで複数のドメイン名(URL)でのアクセスを可能にすること

## HTMLファイルの管理

- 新規のグループを作成し、そこにHTMLファイルの管理者を登録する  
www-admini:x:502:yamamoto
- htmlディレクトリのパーミッションを変更する  
drwxr-xr-x 5 root root 4096 Sep 22 01:06 html/  
↓  
drwxrwxr-x 5 root www-admi 4096 Sep 22 01:06 html/
- Html以下にあるHtmlファイルの属性を以下のように設定しておく  
-rw-rw-r-- 1 root www-admi 1474 May 16 20:16 index.html

## HTMLファイルの作成

- サーバにログインして直接テキストエディタで作成・編集
- クライアント機(windows等)にてテキストエディタで作成・編集を行い、サーバにそのファイルを転送する(ftp)

## httpdの起動設定

- サーバ起動時にhttpdが自動的に起動するよう設定する
- turboserviceを用いてhttpdデーモンの項を有効にする