

Linux Conference 2001

## オープンソースで作るPPPoEサーバの性能評価

http://www.ime.cmc.osaka-u.ac.jp/~h-masuda/LC2001/ (予定)

大阪大学サイバーメディアセンター

梶田 秀夫  
h-masuda@ime.cmc.osaka-u.ac.jp

## PPPoEの実装の選択

userlandで実現したもの(userモード)

- 通常アプリケーションとして実装
  - packet socket(Linux), BPF(BSD) 等
- コンテキストスイッチが多発 → 性能劣化

kernel内で実現したもの(kernelモード)

- カーネル組み込み
- コンテキストスイッチが少ない

→ 両方を比較

- RADIUSサーバによるCHAP認証を利用

## 背景

認証付情報コンセントを安価に構築したい

- クライアントPCの利用者が誰かを特定する
  - 認証前には外部と通信することが出来ない
  - 認証後別のパソコンへの入れ換えが困難
- サーバから利用者が容易に特定できる(ユーザ追跡)
- 実装はいくつか知られている
  - MAC address事前登録, LANA/WILL(大阪市大), PPTP(東京農工大)...
  - 認証VLAN(Omni S/R シリーズ等), EAP(Cisco)...

ちょっと違ったアプローチはないか?

→ PPPoE を使ってみよう

## PPPoEサーバの性能評価

- 転送レート(スループット)
  - PPPoEサーバのCPU使用量
  - PPPoEサーバのメモリ使用量
- 実際にシステムを組んで実測

比較項目

- kernelモード, userモード, 単純なルータ(DHCP)
- PPPoEサーバのCPU性能(クロック周波数)
- PPPoEクライアントの数

## PPPoE(PPP over Ethernet)

Ethernet上で仮想的なPPP通信路を作る(RFC2516)

- Ethernetによる大きな容量の媒体
- PPPによるユーザベースの認証
- フリーな実装多数(UNIX, Windows等)
- ADSLなどで利用
- Windows XPにも標準搭載
- IP over Layer2

認証つき情報コンセントに使えないか?

- 認証前には外部と通信することが出来ない
  - PPP接続前には通信不能
- 認証後別のパソコンに入れ換えられない
  - MAC addressとsessionIDで仮想通信路を識別

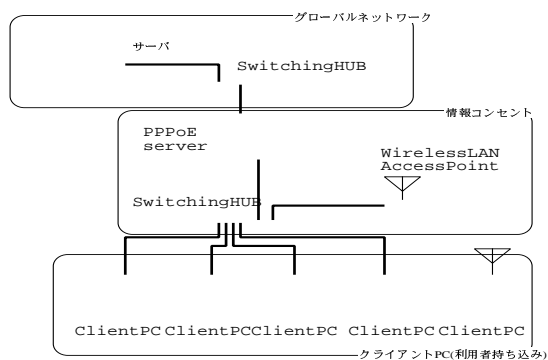
## PPPoEサーバ

	ServerA	ServerB	ServerC	ServerD
CPU	PentiumIII 1GHz	MMX Pentium 233MHz	Pentium 100MHz	PentiumIII 550MHz
Memory	256MB	128MB	64MB	896MB
NIC	Intel EtherExpress Pro/100 x 2			
OS	Linux2.4.2ac22(PPPoEオプション有り) + kernel level pppoe Linux2.4.2ac22(PPPoEオプション無し) + rp-pppoe (注) Linux2.2.1\$pre21 + rp-pppoe(Server Cのみ)			

表2: サーバ計算機のスペック

- 普通に手に入る程度のPC
- ネットワークカードは統一

## システム構成



実力や如何に?

## PPPoEクライアント

	ClientA	ClientB	ClientC	ClientD
OS	Windows2000 Pro	Windows98	Windows98	Linux2.4.2ac22 + kernel level pppoe
RASPPPOE version 0.95b				
CPU	Celeron 633MHz	Celeron 300MHz	Pentium 266MHz	Celeron 633MHz
Memory	256MB	192MB	128MB	256MB
NIC	SIS 900	Intel EtherExpressPro/100	3CCFE575BT (PCCARD)	SIS 900

表3: クライアントPCのスペック

- 普通に手に入る程度のPC
- MS-Windows, UNIX(Linux)を適当に混在

## 用意した外部サーバ／接続環境

### ftpサーバ

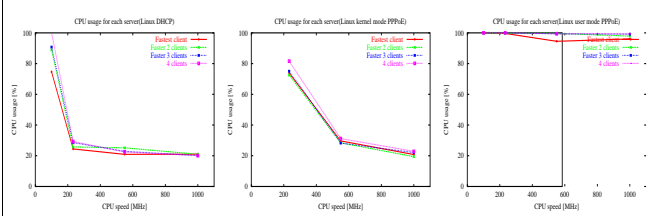
- PentiumII 450MHz, 128MB, IntelEtherExpressPro/100
- Linux2.4.2ac13

### HUB

- IBM8275-416(10/100baseTX自動切替16portHUB)

- PPPoEサーバと外部サーバとの間は100Mbpsで通信可能
  - ftpを実際実施して確認

## 評価結果(CPU利用率)



- kernelモードではPentiumIII 550MHz以上では約30%程度
- userモードではCPUを使い切っている

## 評価方法(スループット/CPU利用率)

- 各クライアントPCで外部サーバから10Mbyteのftpを5回

転送レート = 同時に動くクライアントPCのftpの転送レートの合計値

CPU利用率 = vmstat 1 の記録を利用

→ 特にCPU転送レートは誤差が大き過ぎる。

## 評価結果(メモリ使用量)

### kernelモードの場合

- pppoed + pppd × クライアント数
  - pppoed = 540kByte
  - pppd = 1MByte

### userモードの場合

- pppoe-server + ( pppoe + pppd ) × クライアント数
  - pppoe-server = 550kByte
  - pppoe = 500kByte
  - pppd = 1MByte

kernelモード: 数十台@128Mbyte が最大利用可能台数

userモード: メモリよりもCPU利用率側がボトルネック

## 評価方法(スループット/CPU利用率)

- 各クライアントPCで外部サーバから10Mbyteのftpを5回

転送レート =  $\frac{\text{ネットワーク I/F の積算通信量} (/proc/dev/net)}{\text{積算 CPU 時間}}$

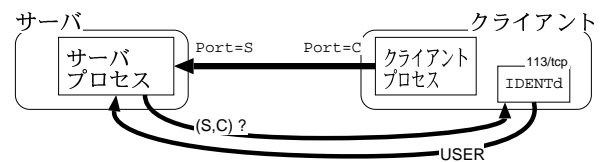
CPU 利用率 =  $\frac{\text{(ユーザ or システム)CPU 時間} (/proc/stat)}{\text{積算 CPU 時間}}$

- proc ファイルシステムを用いて sleep 1 を挟んで取得
  - 取得できる値は、open() 時に決定される

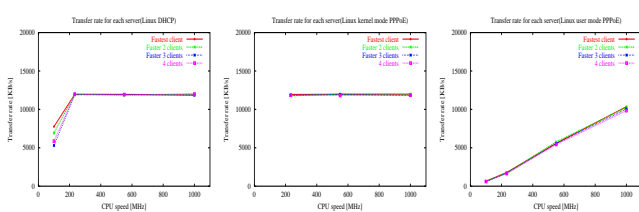
## ユーザ追跡機能

### IDENT(RFC1413)を使用する

- sendmail, apache, squid から利用



## 評価結果(スループット)

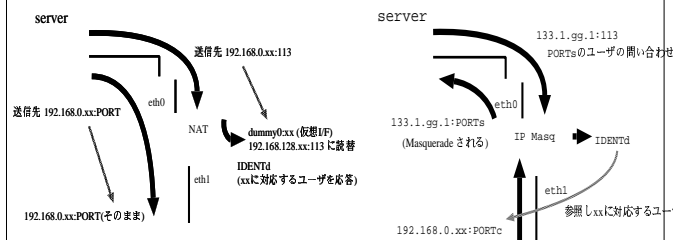


- 転送レートの低下は 1% 程度
- userモードはPentiumIII 1GHzでも 80% 程度
  - 1.3GHzあたりでネットワーク側が頭打ちと予測

## ユーザ追跡機能

### IDENT(RFC1413)を使用する

- sendmail, apache, squid から利用



NAT方式

→ NAT方式とMasq方式を比較

Masq方式

## 性能評価

- 転送レート(スループット)
- PPPoEサーバのCPU使用量
- PPPoEサーバのメモリ使用量
- 実際にシステムを組んで実測

### 比較項目

- IDENTなし、NAT方式、Masq方式
- PPPoEサーバのCPU性能(クロック周波数)
- PPPoEクライアントの数

## IDENT機能実現方式の比較

### NAT方式

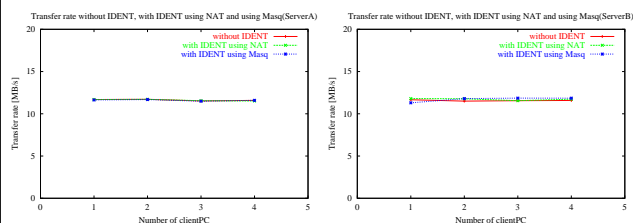
- IPアドレスがクライアント数の二倍必要
- 仮想I/Fが必要
- tcpserver がクライアントの数だけ起動される

### Masq方式

- クライアントプログラムに制約
  - IPアドレスを直接利用できない
- クライアントの区別が容易ではなくなる

→ NAT方式の方が運用上は好ましい

## 評価結果(スループット)



- IDENT機能による転送レートの低下はほとんど見られない
- クライアント数に対してもほとんど低下しない

## まとめ

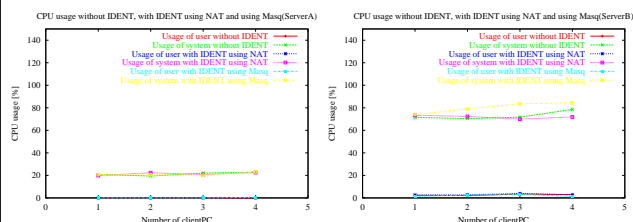
### PPPoEを使った認証つき情報コンセントの実装と評価

- PentiumIII 550MHz Linuxサーバで 100baseTX は確保
- クライアント数4までは、ネットワークボトルネックと推察
- kernelモードがよい
- IDENTを使ったユーザ追跡が可能
  - NAT方式とMasq方式

### 今後の課題

- クライアント数を増やした場合の評価
  - シミュレーション
- Gigabit Ethernet に増強した場合
- 実運用のノウハウ蓄積

## 評価結果(CPU利用率)



- サーバA(1GHz)で 20% ほどでほぼ横ばい
- サーバB(233MHz)では 80% ほどでほぼ横ばい

## 評価結果(メモリ使用量)

- プロセスサイズをtopを利用して1秒毎に取得する

### IDENT機能なしの場合

- pppoed + pppd × クライアント数
  - pppoed = 540kByte
  - pppd = 1MByte

### NAT方式の場合の追加分

- tcpserver × クライアント数 + identd × 問い合わせ数
  - tcpserver = 499kByte
  - NAT方式用identd = 1200kByte

### Masq方式の場合の追加分

- tcpserver × 1 + identd × 問い合わせ数
  - Masq方式用identd = 2299kByte