

ネットワークアクセス可能なポータブルレポジトリの設計と実装

尾崎 哲[†] 神田 充[†]
嶋田 雄二郎^{††} 山田 竜也^{††}

近年、携帯電話によるネットワークサービスや公衆無線 LAN サービスのように、多くの情報に常にアクセスできる状態をいたいという要望を満たすためのサービスが盛んである。これら多様な情報を人間をの代わりに覚えておくための補助記憶装置(これをレポジトリと呼ぶ)の一例として、ユーザのメールを収集、蓄積し、ネットワーク経由でユーザの望む装置へ提示する、小型装置の試作を行った。この装置を持ち歩くユーザは、移動先の PC 等でも常に通常と変わらない操作が可能であり、情報の分散や整合性の確保を心配する必要がない。本報告では、本試作のソフトウェア設計と実装について述べる。

Design and Implementation of a Network Accessible Portable Repository

SATOSHI OZAKI,[†] MITSURU KANDA,[†] YUJIRO SHIMADA^{††}
and TATSUYA YAMADA^{††}

Recently, network services through cellular phone and public wireless access becomes very popular, which fulfills users' and that they always want to be accessible to lots of information. We have prototyped a small device as an example of auxiliary storage appliance called "repository", which holds lots of information the user accesses.

This repository fetches and stores the user's Email then delivers to the any device he/she wishes through the network. Therefore, if the user carries this repository, he/she can operate the PCs on the remote site just the same way as usual and he/she need not care about the distribution and synchronization of information.

In this paper, we will describe the software design and implementation of the prototype portable repository.

1. はじめに

近年、定額制のインターネット接続環境等により、飛躍的に多くの情報を容易に入手することが可能になった。これに伴い、多くの情報に常にアクセスできる状態をいたいと考える人が増えている。もちろん、全ての情報を覚えておくことは困難であるから、人間をアシストする補助記憶(これを本稿ではレポジトリと呼ぶ)を使うことになる。ネットワークアクセス機能付携帯電話の普及や、PDA に使いそうな情報をコピーして持ち歩く、といった行動は、このことを裏付けるものだと思う。

ここで、レポジトリをどこに置くのか、という観点からこの 2 つの例をみると、「ネットワーク上のストレージ」と「持ち歩く小型機器」、という 2 つのシナリオを示唆しているのが興味深い。

前もって要りそうな情報を選択しコピーしておくという、Palm や PocketPC にみられる Sync テクノロジーは、今までの技術水準では現実的な解であったが、この手法は、選択するのに手間がかかる、何を選択すべきか予想が付かない、という点で、扱う情報が増えようと破綻しがちである。

HDD、フラッシュメモリなどに代表されるストレージ機器の性能の飛躍的な伸びとコンパクト化、また、ADSL、ホットスポットなどに代表されるネットワーク帯域幅の大幅な増加とアクセス機会の増大を考えると、どちらも進歩の激しい分野であり、一昔前の 3 桁増のスペックなど当り前の世界である。こうなると、どちらのシナリオをとるにせよ、レポジトリに十分な容量を確保し、選択しないで全ての情報をレポジトリへ入れてしまう、という手法が現実味を帯びる。

[†] 株式会社東芝 研究開発センター 通信プラットフォームラボラトリー

Communication Platform Laboratory, Corporate Research and Development Center, Toshiba Corporation

^{††} 株式会社東芝 e-ソリューション社 SI 技術開発センター
Systems Integration Technology Center, e-Solution Company, Toshiba Corporation

この2つのシナリオは一長一短であるが、アクセスの容易さとコストの点から、今回は小型機器にレポジトリを置いて全部持ち歩くシナリオについて検討したい。

良い例として、ポータブル音楽プレイヤーで「小型機器にレポジトリを置いて全部持ち歩く」を実現したのが、iPod⁹⁾、GIGABEAT¹⁰⁾といったHDD内蔵型プレイヤーだと考えられる。

さて、持って歩くとなると機器の大きさが問題になる。程度問題もあるが、小さい方が望ましい。一方、情報に視覚的にアクセスするとなると、表示デバイスの大きさが問題になる。こちらは基本的に大きい方が望ましい。これを両立するために、近くにある大きな表示デバイスと接続、連絡するのがよいのではないだろうか。

ポータブル音楽プレイヤーの例で言えば、普段はヘッドフォンで聴くが、オーディオケーブルでカーコンポや家のステレオセットと接続することでより良い環境で聴くことができるのと同じ考え方である。

レポジトリと表示デバイスはIPベースのネットワークで接続することを考える。USB、IEEE1394なども接続方法の候補となるが、これらがPCベースあるいはAV機器ベースなどハードウェアへの依存性が高いのに対して、IPベースのネットワークならばハードウェア依存性が少ないこと、また、インターネット標準の通信プロトコルを使って運用することで、異機器間でのデータの相互運用性に優れることが期待できる、といったことが理由である。

以上のような背景をもとに、本稿ではネットワークアクセス可能なポータブルレポジトリを提案する。

以下、2章で、レポジトリに蓄えるデータとしてメールが重要であることを述べ、それに特化した「携帯マイクロサーバ」の設計を記す。3章において実装手法について述べ、4章で実験を通じて機能を検証する。

2. 携帯マイクロサーバの設計

メールは今やメッセージの伝達手段であるばかりでなく、それが蓄積されていくことで重要な資料となる。

会社の機器、自宅の機器、自分のモバイル機器といった複数の機器で別個にメールを受け取ってしまうと、メールが各機器に分散してしまう。メッセージを分散させず、いつでもどこでも整合性の取れたメールの運用を行うには、オリジナルのメールを持ち歩くことが確実な方法である。一箇所のレポジトリにメールを蓄積することにより、必要などきに必要な資料を取り出せるようになる。

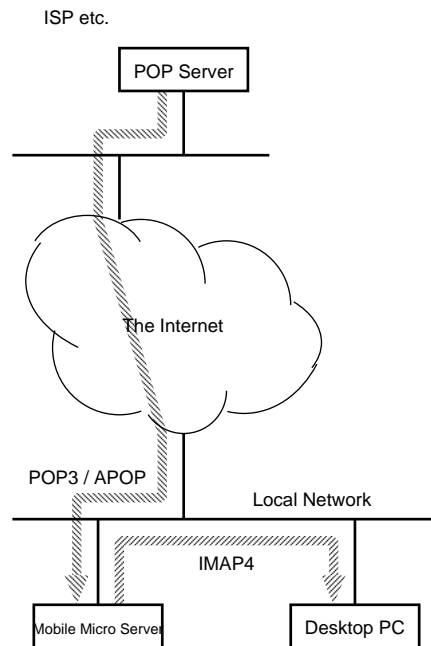


図1 携帯マイクロサーバにおけるメッセージの流れ：携帯マイクロサーバはISPのメールサーバに対してはPOPクライアントとしてふるまい、メッセージを取得する。一方ユーザがメールを閲覧するMUAに対してはIMAPサーバとしてふるまい、メッセージを提供する。

常に持ち歩き、必要な時には近くのPCの側に置き、PCの広い画面での使い慣れたメールクライアント(MUA, Message User Agent)でメールを読み書きする。そのようなメール用のパーソナルレポジトリを実現する機器を、携帯できる小さなサーバということで「携帯マイクロサーバ」と呼ぶことにし、その設計を試みた。

携帯マイクロサーバは、メール専用端末やPDAのような外観のハードウェアで、内部に比較的大容量のストレージと外部との通信インタフェースを持つ。

ユーザは、外出時には従来のメール専用機器同様に携帯マイクロサーバ単独で簡単な応答ができる。会社や自宅では普段使っているPCと同じネットワークに携帯マイクロサーバを接続して、PCからメールの読み書きをすることができる。

これを実現するため、携帯マイクロサーバには、次のような機能を組み込む。

メールを集める 携帯マイクロサーバはメールサーバに対してPOP¹⁾クライアントとしてふるまい、メールを手元を集める。メールサーバがファイアウォールの外にあるとき、携帯マイクロサーバはSOCKS³⁾プロトコルを用いてファイアウォールを越え、メールサーバにアクセスする。複数のメ

表 1 小羊ルータ ハードウェア仕様：小羊ルータは小型ながらも PC/AT 互換のアーキテクチャのハードウェアであり、ソフトウェアの試作が比較的容易である。

寸法	93 × 91 × 30 (mm)
重量	316g
消費電力	3W (max 6W)
アーキテクチャ	PC/AT 互換
CPU	AMD Elan520 (Am5x86 133MHz 互換)
RAM	32MB
HDD	IBM Microdrive (1GB), CF ソケット装着
I/F	Ethernet Port (10BaseT) × 2 ステータス表示 LED × 1, 押しボタン × 1 シリアルポート (Dsub9pin) × 1, CF Type2 × 1

表 2 主な利用ソフトウェア一覧：ソフトウェアの機能試作にあたっては、いくつかのオープンソースソフトウェアをそのまま、あるいは一部に手を入れて利用した。

OS	Red Hat Linux 7.1 ¹²⁾
IPv6 Stack	USAGI (2001/09/03 version) ¹⁵⁾
SLP	OpenSLP 1.0.4 ¹⁶⁾
IMAP	UW-IMAPD 2001 ¹⁷⁾
SMTP	sendmail 8.11.6 ¹⁸⁾
IMAP before SMTP	DRAC ¹⁹⁾

イルサーバにアカウントを所持している場合は次々と巡回アクセスしていき、各サーバに分散して届いているメールを集める。

メールの閲覧 携帯マイクロサーバは MUA に対して IMAP²⁾ サーバとしてふるまい、内部に保管されているメールに対する閲覧などの操作を受け付ける。

メールを送信する 携帯マイクロサーバは MUA に対して SMTP⁴⁾ サーバとしてもふるまい、MUA から送信されたメールを受け取る。受け取ったメールは、特定の SMTP サーバに送るのではなく、宛先に応じて DNS を引いて送り先サーバを決定し送信する。これにより、送信先 SMTP サーバと異なるネットワークに接続していると「SMTP リレー拒否」のため送信できない、という問題を回避する。

Plug and Play 携帯マイクロサーバを近くのネットワークにつないだ時、面倒な設定なしにコンフィグレーションできる機能である。

3. 携帯マイクロサーバの実装

携帯マイクロサーバの機能を検証するため、ワイルドラボ社製小羊ルータ¹¹⁾のハードウェアを利用して実装を行った。小羊ルータのハードウェア仕様を表 1 に、主な利用ソフトウェアの一覧を表 2 に示す。

小羊ルータは PC/AT アーキテクチャなので、ソフ

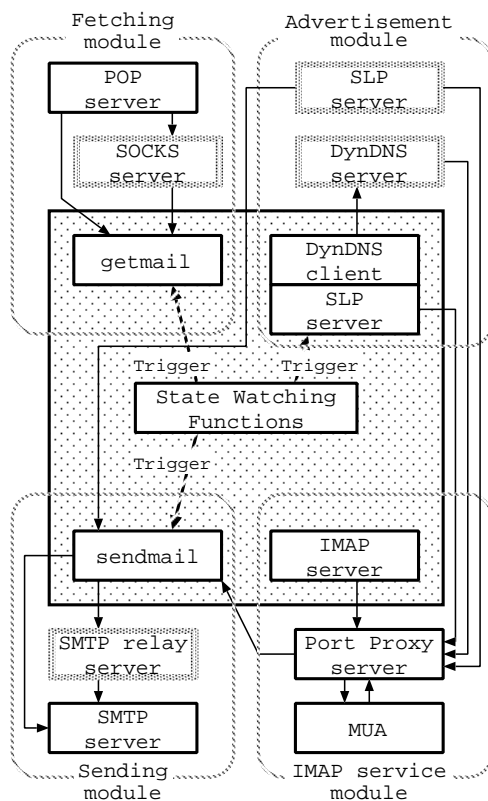


図 2 携帯マイクロサーバのモジュール構成：メール取得 (fetching)、位置情報広報 (advertisement)、メール送信 (sending)、IMAP サービス (IMAP service) の 4 種類の機能モジュールとそれらをコントロールする状態監視 (state watching) 機能モジュールから構成される。

トウェアは PC 上の小パーティション上にインストールし、これをマイクロドライブにコピーし、小羊ルータ上でブートさせる、という手順で組み込んだ。

IPv6 環境を完全にサポートするため、USAGI の IP スタックを用い、OpenSLP 等、既存のソフトウェアを IPv6 対応に改造した。

SOCKS クライアント、POP クライアント、IMAP クライアントの機能は RFC を元に独自に実装を行った。

モジュール構成を図 2 に示す。網掛けの部分小羊ルータのハードウェア上に組み込む機能で、その外側にあるのはネットワーク上の別のマシンに置かれる機能である。メール取得、位置情報広報、メール送信、IMAP サービスの 4 種類の機能モジュールを、状態監視機能モジュールがコントロールすることで全体を動作させている。これらの機能モジュールはいくつかの実行ファイルとスクリプトで構成される。それぞれの機能モジュールについて個別に述べる。

3.1 メール取得機能

getmail POP もしくは IMAP クライアントとしてふるまい、設定ファイルに記述されているメールサーバからメールを取得する。SMTP クライアントとしてふるまい、取得したメールを localhost の sendmail に送出することで、ユーザのメールプールに格納する。メールサーバへの経路の途中にファイアウォールがある場合、SOCKS プロトコルを用いてこれを越える。C 言語で独自に実装。

3.2 IMAP サービス機能

UW-IMAPD UW-IMAPD を IPv6 対応 inetd 経由で起動している。ユーザのメールプールにあるメールをネットワーク上の IMAP クライアントに開示する。

3.3 メール送信機能

sendmail sendmail を IPv6 対応 inetd 経由で起動している。

resetsmtp 携帯マイクロサーバが接続しているネットワークに応じて sendmail.cf を書き換えることにより、メールの配送方法を適宜変更する。

slpcs SLP⁽⁷⁾⁽⁸⁾ を使用し、携帯マイクロサーバが接続されているネットワークで設定ファイルに記述されたサービスが広報されているかどうかを調べる。シェルスクリプトとして実装。

getstatic ネットワークインタフェースについている IP アドレス情報を元に、設定ファイルに書かれたそのネットワークの SMTP サーバを調べる。C 言語で実装。

IMAP before SMTP 携帯マイクロサーバがユーザ以外からの SMTP リレーを受け付けないよう、SMTP サービスを有効にするためには IMAP での認証を済ませていることを確認する。UW-IMAPD を修正して DRAC を組み込むことで実現している。

3.4 位置情報広報機能

ifup-local 広報する位置情報の更新を実行する。mcinit を呼ぶシェルスクリプトとして実装。

mcinit SLP サーバで使用する SLP 設定ファイルを更新し SLP サーバを再起動することで、更新した位置情報を SLP でネットワークに広報する。シェルスクリプトとして実装。

mc_rwfile コマンドラインで渡された情報を基に SLP サーバで使用する SLP 設定ファイルを書き換える。mcinit から使われる。C 言語で実装。

SLP サーバ 携帯マイクロサーバが SLP で位置情報

を広報する際の SLP サーバとして OpenSLP を IPv6 対応に修正したものを使用する。基本的には携帯マイクロサーバ内に内蔵するが、サイト内に SLP サーバ置かれていれば、位置情報の広報と共に、サイト内の SMTP 中継サーバを携帯マイクロサーバ内の sendmail に知らせる役割も担う。

dnsupdate 携帯マイクロサーバのネットワークインタフェースに付いた IPv6 グローバルアドレスを Dynamic DNS Update⁽⁶⁾ により DNS サーバに登録する。BIND の nsupdate を呼び出すシェルスクリプトとして実装。

rtosol ネットワークに RS (Router Solicitation) パケットを送出し、RA (Router Advertisement) パケットを受け取ることで、接続したネットワークにおける IPv6 アドレスプレフィックス情報を取得する。C 言語で実装。

3.5 状態監視機能

ifmond クライアントからの要求により、ネットワークインタフェースに関する情報を提供する。C 言語で実装。

click 携帯マイクロサーバの押しボタンスイッチの状態を監視する。製造元 (ワイルドラボ) 提供のソースコードを基に、押しボタンのクリック、ダブルクリック、長押し of 判別などを C 言語で実装。

3.6 他のマシン上に設置する機能

MUA ユーザが利用するメールクライアント。IMAP をサポートしていればユーザの好みのものを利用できる。実験では、Windows XP 上の Winbiff⁽¹³⁾ と Red Hat Linux 上の Sylpheed⁽¹⁴⁾ を利用した。

POP/IMAP server オフィスや ISP 上で提供される POP/IMAP サーバ。実験では UW-IMAP およびこれに付属する POP サーバを利用。

PortProxy server SLP で通知される携帯マイクロサーバのアドレスを MUA に伝えるために用いる。MUA のプログラムは必ずしもソースコードが入手できるとは限らないため、MUA の設定を動的に変更することは、時に困難である。そこで、SLP で携帯マイクロサーバのアドレスを入手し、携帯マイクロサーバ上の IMAP や SMTP のポートを localhost にフォアワードするプログラム mproxy を MUA を動作させるマシンと同じマシンで動かしておく。MUA の設定には localhost を記述しておけばよい。C 言語で実装。SLP のライブラリを利用。

SOCKS server ファイアウォール越しに POP/IMAP サーバにアクセスする場合に使用。実験では、

SOCKS の参照実装を IPv6 に対応させたものを利用。

SMTP (relay) server オフィスや ISP 上で提供される SMTP サーバ。実験では sendmail を利用。

SLP server 必須ではないが、オフィスでのメールサーバのアドレスを提供することで、設定の自動化が可能。実験では OpenSLP を利用。

Dynamic DNS server 必須ではないが、携帯マイクロサーバのアドレス変更を受け付けることで、携帯マイクロサーバの現在のアドレスを広く知らせることができる。

DHCP server オフィスや ISP 上で提供される DHCP⁵⁾ サーバ。IPv4 だけをサポートするネットワークでは、動的に IP アドレスを取得するために DHCP サーバを利用する。IPv6 をサポートするネットワークでは、ルータが IP アドレスを配布するため DHCP を使わずに済む。

4. 検証実験と評価

4.1 運用評価概要

携帯マイクロサーバを携帯し、遠隔ネットワークからメールサーバに接続してメールを携帯マイクロサーバに取得し、携帯マイクロサーバに格納されたメールを同じ LAN に接続したメールビューア (PC) にて閲覧するという実験と、メールビューアから携帯マイクロサーバを経由してメールサーバにメールを送信するという実験を行った。

4.2 実験 1: IPv6 だけの環境

米国ミネアポリスで開催された 53rd IETF において、会場のネットワークを利用して携帯マイクロサーバの運用実験を行った。この実験では、FQDN (Fully Qualified Domain Name) による名前解決を行わずに、IPv6 だけのネットワークにおける携帯マイクロサーバの利用可能性を実験した。実験環境を図 3 に示す。

4.2.1 携帯マイクロサーバの設定

Dynamic DNS のサポートなしの実験のため、携帯マイクロサーバとメールビューア間は IPv6 アドレスを直接指定して通信した。

IP アドレスの設定 近隣のルータから RA (Router Advertisement) を受け、自動設定である。実験では、プライバシ拡張による IPv6 アドレスと、IEEE EUI-64 による IPv6 アドレスの 2 個が付与されたが、後者のアドレスだけを用了。また、このネットワークでは IPv4 も利用可能であるが、IPv4 の DHCP を受信しない設定にする。

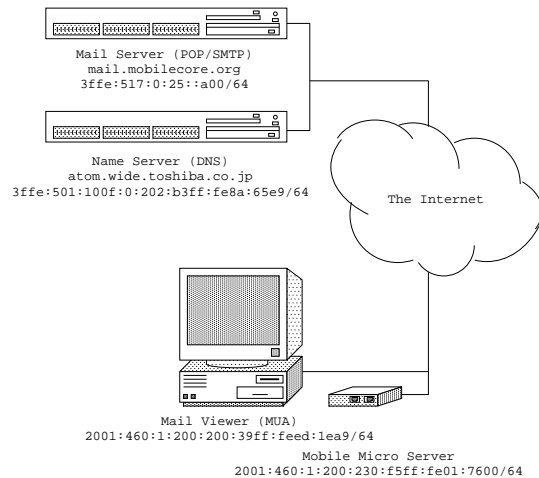


図 3 実験環境その 1: IPv6 の使える実験環境の一つとして、IETF 会場の IPv6 ネットワーク環境を利用して実験を行った。

リゾルバの設定 Red Hat Linux ではルータからの RA (Router Advertisement) を受けたときにはネットワークインタフェースにアドレスを付与する動作だけを行い、ネームサーバの設定は変更しない。移動先でも移動前に使っていたネームサーバを継続して利用する。移動先にもネームサーバは用意されていたが、ユーザの設定変更の手間を省いた。

POP クライアントの設定 設定ファイル getmailrc を記述しておく：

```
[DEMO]
mailserver=mail.mobilecore.org
username=ymd
password=hoge
*.mobilecore.org=DIRECT
```

この設定の意味は次の通り：
接続するメールサーバ mail.mobilecore.org
メールサーバのユーザ名 ymd
ユーザのパスワード hoge
mobilecore.org からは 直接接続

4.2.2 メールビューアの設定

IP アドレスの設定 メールビューア用の PC は RA を受信して自動的に IPv6 アドレスを得る。このネットワークでは IPv4 も利用可能であったが、IPv4 は使わない実験のため、IPv4 の DHCP を利用しない設定にする。

リゾルバの設定 名前による通信は行わないので設定不要。

MUA の設定 MUA (Winbiff) のアカウントの設定を図 4 のように行った。ここで、デフォルトでは

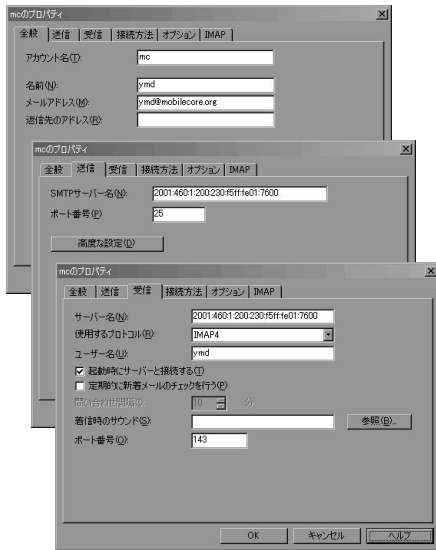


図 4 MUA (Winbiff) の設定画面 . Winbiff は Windows 上で動作する MUA で、IPv6 に対応している . この実験ではサーバ名の設定部分に IP アドレスを直接入力している .

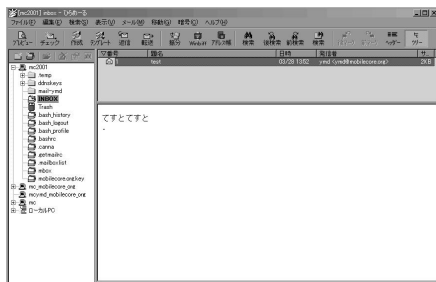


図 5 MUA (Winbiff) のメール閲覧画面 . 携帯マイクロサーバに取り込まれたメールがネットワーク経由でサブされ、MUA で正しく表示されることを確認した .

アカウント名が IPv6 アドレスになってしまうため、適当な名称に変更する . IMAP サーバおよび SMTP サーバのアドレスに携帯マイクロサーバの IPv6 アドレスを入力し、メールアドレスを設定すれば完了である .

4.2.3 運用

メールの取得 携帯マイクロサーバの外部に付いているボタンを 1 回押す .

メール閲覧 メールビューアから携帯マイクロサーバに IMAP4 で接続し、メールが読めることを確認 (図 5) .

性能測定 携帯マイクロサーバがメールサーバからメールを取得するのにかかる時間を time コマンドにて測定したところ、約 14 秒であった . なお、携帯マイクロサーバからメールサーバに対して ping6

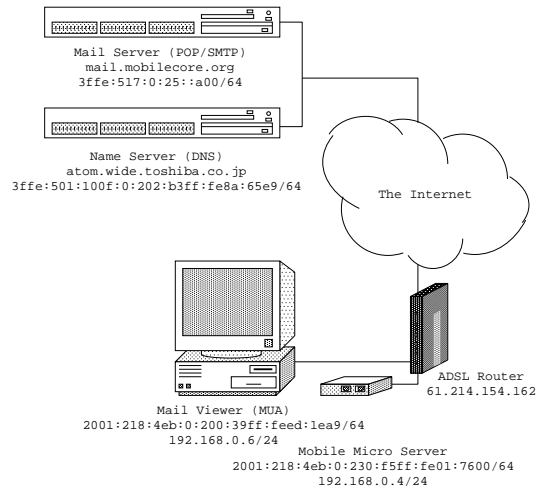


図 6 実験環境その 2 : 家庭のネットワーク環境の一例として、ADSL 回線を利用した環境で実験を行った .

の 5 回の平均は 322ms であった .

メール送信 Winbiff からメールの送信を正常に行えた .

携帯マイクロサーバの終了処理 携帯マイクロサーバを適切に終了するためには、サーバ側面のボタンを 3 秒以上押しシャットダウン処理を行わせる .

4.3 実験 2: IPv4, IPv6 混在環境

ここでは、IPv4, IPv6 混在環境下で DNS Dynamic Update を用いて携帯マイクロサーバのホスト名を動的にネームサーバに登録することによって、携帯マイクロサーバに付与された IP アドレスによらずにアクセスできるかを実験する .

4.3.1 携帯マイクロサーバの設定

リゾルバの設定 DHCPv4 を有効にしていると Red Hat Linux の起動時に /etc/resolv.conf を IPv4 のもので上書きされてしまうため、起動後に手動で設定しなおす必要があることがわかった . これは、dnsupdate コマンドを IPv6 だけのアドレス割当てで動作するように修正するか、Red Hat Linux の起動スクリプト中で /etc/resolv.conf の内容を上書きしないように修正することによって解決する .

不正経路問題 携帯マイクロサーバでは起動時の初期状態で IPv4 のデフォルト経路に不正なものが混入しているため、このままでは携帯マイクロサーバ以外にデータを送信できない . この問題を解決するためにデフォルト経路を削除する必要がある .

なお、IPv6 に関してはこの問題は発生しない .
DNS Dynamic Update クライアントの設定 クラ

イアント用の鍵 *.key ファイルと *.private ファイルを同じディレクトリに用意しておき、dnupdate コマンドがこれを参照するように指定する。

4.3.2 メールビューアの設定

IP アドレスの設定 Microsoft Windows XP は IPv6 をサポートしているが、名前解決は IPv4 トランスポートによってしか行えないので、DHCPv4 で IPv4 アドレスを設定する。インターネットに接続している ADSL ルータにグローバル IPv4 アドレス (65.214.154.162) が付与されており、それを NATP によって 192.168.0.0/24 のネットワークアドレスに変換している。携帯マイクロサーバとは IPv6 で通信するため、RA を受信し、設定された IPv6 アドレスを使用する。

リゾルバの設定 この実験では FQDN によって通信するので、ネームサーバのアドレスを設定する必要がある。実験環境は ADSL によってインターネットに接続されており、ADSL ルータ (192.168.0.1) が名前解決を代行するため、メールクライアントのネームサーバの設定は DHCPv4 によって自動設定される。

メールクライアントの設定 IMAP サーバのアドレスと SMTP サーバのアドレスとして携帯マイクロサーバの FQDN を指定する。その他のアカウントの設定は実験 1 と同様である。

4.3.3 運用

4.2.3 と同様に運用を行い、結果も同様に、アドレスの取得、メールの受信、閲覧、送信が問題なく動作した (図 5)。

4.4 その他の構成での実験

実験 1, 2 の他に、MUA として sylpheed を利用した実験、SOCKS を使ってファイアウォール越しにメールを取得する実験、SLP を用いて携帯マイクロサーバのアドレスを取得し port proxy によって MUA へ仲立ちする実験、などを行い、設計通りの動作を確認した。

4.5 考察

実験後、我々は実際に携帯マイクロサーバを生活用のメールアカウントに対して適用し、使い勝手などについて意見を集めた。挙げた項目と、その対応案を示す：

セットアップが面倒 リモートログインしてテキストファイルの設定ファイルを書きかえるのは難しい
Web 経由など、設定用 UI を作り込む
デバックが困難 設定が誤っていた時、なぜ動作しな

いかの原因がわかりにくい 設定用 UI で入力値のチェックをする。Web 経由で動作ログをとれるようにする

さらなる自動化 ボタンを押さずにすませられないか
ネットワークインタフェースを自動でセンス
リンクの up/down を検知して動かす

小型化 本気で持ち歩くならもっと小さくないと 本試作のハードウェアは既存の物の流用。製品にするなら小さく作り込むなり、PDA や携帯電話の一機能として組み込むなど工夫する

電源 電源をつながなければならぬのは手軽さをスポイルする 充電電池を内蔵し、クレイドルで充電するなど工夫する

無線 近くに置くだけで済むようにネットワークを無線にすると便利かも

補助プログラム 携帯マイクロサーバのアドレスを自動的に知るために MUA 側にプログラムを入れる必要があるが、他人の PC にインストールするのは困難 IMAP だけでなく WebMail のインタフェースを設け Web ブラウザから閲覧する

メール以外 メールだけでなく、Web キャッシュなど、ポータブルレポジトリとして汎用性を高めるとよいかも

5. まとめ

ネットワーク接続可能なポータブルレポジトリの一例として、携帯マイクロサーバを設計、実装し、IPv4 および IPv6 ネットワーク環境下においてほぼ設計通りの動作を確認した。

携帯マイクロサーバは、次のような特徴を持つ：

メール情報を一元管理 携帯マイクロサーバは会社や自宅のメールアカウントを一括で管理する。携帯マイクロサーバさえあれば、複数のアカウントのメールにいつでもアクセスできる。

その場の環境を活用 近くに PC があれば、PC から携帯マイクロサーバを操作できる。PC の持つ大きな表示画面や使いやすいキーボードが活用でき、使い慣れた MUA が使える。

同期作業なしで整合性確保 携帯マイクロサーバ内には常にオリジナルのメールデータが入っており、これを持ち歩くため常に最新の情報を操作できる。同期作業は必要なく、コピー同士の整合性に気を使う必要がない。

簡単アクセス 面倒な接続設定は一度だけ。様々なネットワークを渡り歩いても自動で認識し設定を切り替える。メールサーバへの経路にファイアウォー

ルがあれば乗り越える。

高度な相互運用性 携帯マイクロサーバはインターネット標準プロトコルを使用。データの互換性の問題に悩まされることがない。標準をサポートしているアプリケーションであれば、ベンダーやプラットフォームに関わらず使い慣れたアプリケーションが使える。将来にわたってメイルデータのポータビリティを確保する。

一方、実際に実装および実験を行った結果、いくつかの問題点も明らかになった。今後の課題は、これらの問題点を解決するとともに、機器の小型化、あるいは、携帯電話や PDA といった実際に持ち歩く機器にこの機能を内蔵すること、メイルに限らず、ユーザが処理する様々なデータに対応すること、などが挙げられる。

謝辞 本研究の一部は、通信・放送機構 (TAO) による平成 12 年度補正予算関連「成果展開等研究開発」事業 (アプリケーション開発基盤システム及びネットワーク構築システム) に係る研究助成によるものである。

SOCKS5 の相互接続性確認実験のため、Permeo Technologies, Inc. (<http://www.permeo.com/>) より SOCKS 参照実装コードを提供頂いた。

本研究にあたり、東芝 SI 技術開発センターの島津伸行氏、倉富修氏、岡光秋氏、栄光宏氏、岡本利夫氏、および東芝研究開発センターの井上淳氏に有益な議論を頂いた。

参 考 文 献

- 1) J. Myers, and M. Rose, Post Office Protocol - Version 3, RFC1939 (Standard), 1996.
- 2) M. Crispin, Internet Message Access Protocol - Version 4rev1, RFC2060 (Proposed Standard), 1996.
- 3) M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas and L. Jones, SOCKS Protocol Version 5, RFC1928 (Proposed Standard), 1996.
- 4) J. Klensin, Editor, Simple Mail Transfer Protocol, RFC2821 (Proposed Standard), 2001.
- 5) R. Droms, Dynamic Host Configuration Protocol, RFC2131 (Draft Standard), 1997.
- 6) P. Vixie, Ed., S. Thomson, Y. Rekhter and J. Bound, Dynamic Updates in the Domain Name System (DNS UPDATE), RFC2136 (Proposed Standard), 1997.
- 7) E. Guttman, C. Perkins, J. Veizades and M. Day, Service Location Protocol, Version 2, RFC2608 (Proposed Standard), 1999.
- 8) E. Guttman, Service Location Protocol Modifications for IPv6, RFC3111 (Informational),

2001.

- 9) アップルコンピュータ, iPod, <http://www.apple.co.jp/ipod/>.
- 10) 東芝, GIGABEAT, <http://www.toshiba.co.jp/mobileav/audio/>.
- 11) Wildlab., xDSL/CATV インターネット接続用超小型 IP ルータ/LinuxBox 小羊ルータ LAMB, <http://www.wildlab.com/>.
- 12) Red Hat Japan, <http://www.jp.redhat.com/>.
- 13) インターネットメールクライアント Winbiff, <http://www.orangesoft.co.jp/Winbiff20/>.
- 14) Sylpheed - a GTK+ based, lightweight, and fast e-mail client -, <http://sylpheed.good-day.net/>.
- 15) USAGI Project - Linux IPv6 Development Project, <http://www.linux-ipv6.org/>.
- 16) OpenSLP Project, <http://www.openslp.org/>.
- 17) University of Washington, UW-IMAP, <http://www.cac.washington.edu/>.
- 18) Sendmail Consortium, <http://www.sendmail.org/>.
- 19) Gary Mills (University of Manitoba), Dynamic Relay Authorization Control, <http://mail.cc.umanitoba.ca/drac/>.