複数OS環境におけるユーザ管理

田中哲朗, 安東孝二，吉岡顕†

東京大学情報基盤センターの教育用計算機システムは，実ユーザ数2万人という大規模なシステムである．1999年4月に教育用計算機システムを導入するにあたっては，Windows Terminal Server Editionを大規模に導入し，端末にNC(Network Computer)を1600台採用するという他に例のない構成を採用した．

ユーザは1つの端末で，Unix, Windows, NC-OSの3種類の環境を使い分けることもできる．どの環境を選んで使ってもユーザが混乱することにないような環境を構築すると共に，管理の手間を増やさないことを設計目標とした．

NCは本来，ビジネスにおける定型業務などを目的に開発されているため，教育用として必要な機能のいくつかを補う必要があった．また，教育的な立場からプリンタの使用に関して制限を加える仕組みも導入した．

4月から運用を始めた新システムは個々の問題は生じているが，おおむね順調に動いている．このシステムは他の教育機関におけるシステムの設計においても，参考になる点が多いと思う．

User management in multiple OS

TETSURO TANAKA, KOJI ANDO, AKIRA YOSHIOKA†,†,†

The educational computer system of the Information Technology Center, the University of Tokyo, is a huge system, which users are over 20,000. In April 1999, we replaced the old system with the new one, which consists of many Unix servers, many Window TSE(Terminal Server Edition) servers and 1,600 NCs(Network Computers).

Users can use applications in three environments: Unix, Windows, NC. One of the goals of system design was not to confuse users in three environments, and an another goal was to reduce administration cost.

Since NCs were originally designed to be used in daily works in business, we added functions needed for educational use. And we added a mechanism which limit the use of printers for each user.

1. 教育用計算機システムの概要

東京大学の教育用計算機システムは登録ユーザ数3万人，実ユーザ数（1ヵ月に一度程度はシステムの一部を使うもの)2万人弱，端末は駒場地区，本郷地区の30ヶ所に分散して約1600台配置されるという大規模なシステムである．

今年の4月にシステムを更新するにあたっては，それまでのX端末を中心にしたUnix環境だけでなくWindows環境も充実してほしいという要求が大きくなってきた⋆．

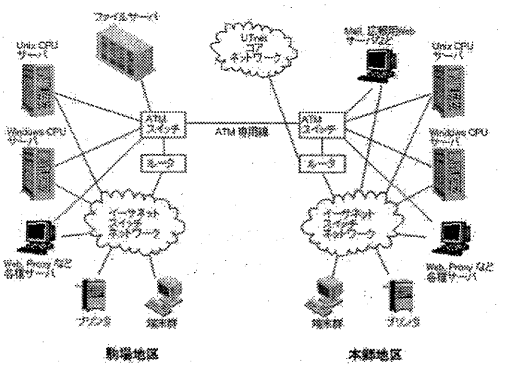

図1 教育用計算機システム

そのため，端末としてNC(Network Client)を採用して，Windows, Unix 両方の環境を使用可能にした．Windows環境，Unix環境のどちらを利用する場合も，NC本体ではなくサーバに

† 東京大学情報基盤センター
Information Technology Center, University of Tokyo
ktanaka@ecc.u-tokyo.ac.jp

⋆ それまでのシステムでも全体の30 %弱はPC端末だったが，OSがWindows 3.1でメモリ，CPUも貧弱だったので利用者は少なかった

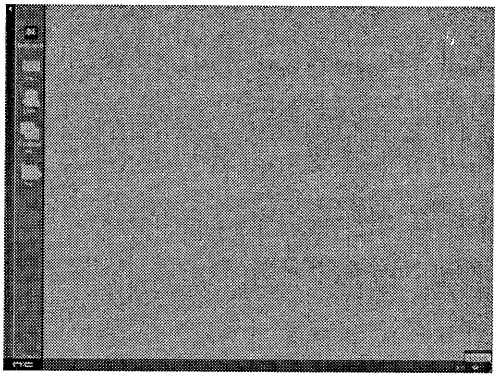

図 2　NC デスクトップ画面

ログインして，アプリケーション本体はサーバ上で実行して，画面表示と入力のみを端末で実行する形態を取る．このような形態は Unix 環境では一般的だったが，日本語 Windows 環境では 1998 年の Windows Terminal Server Edition ではじめて可能になった⋆．

センターの利用者は主に NC(Network Computer) と呼ばれる端末からシステムを利用する⋆⋆．学外者の利用を許して，無料のインターネットカフェにしないこと，ログを残す必要があることから，ユーザ名，パスワードによる認証をおこなっている⋆⋆⋆．

NC にログインすると，図 2のような画面が出て，WWW ブラウザ (Netscape Navigator)，メールクライアント (ICEmail) などのインターネットアプリケーションをローカルに使うことができる．

Unix サーバ，Windows サーバと接続するためのメニューも用意してある．Unix サーバ，Windows サーバ共に X プロトコルで接続する．ユーザの混乱を避けるために，どちらも Xnest を利用して NC のウィンドウの一つが X 端末，Windows 端末であるかのように見せている．

サーバは以下のような構成になっている．

Unix サーバ　Sun Enterprise 450 x 50 台

- CPU は Ultra SPARC 300MHz 4 プロセッサ
- OS は Solaris 2.6
- メモリは 1GB
- 1 サーバあたり，同時 40 名の使用を想定
- NC のブートサーバ，プリンタサーバを兼用

Windows サーバ　NEC Express 5800/140Ha x 25 台

- CPU は Pentium II Xeon 450MHz 4 プロセッサ
- OS は Windows NT Terminal Server Edtition に Citrix Metaframe, NCD WinCenter を付加
- メモリは 2GB
- 1 サーバあたり，同時 40 名の使用を想定．ライセンスによる管理

これらのサーバに接続をする際は，再度ユーザ名，パスワードによる認証をおこなう．

他に認証を必要とするサーバとしてメールサーバがある．メールサーバは，NC, Unix, Windows のどのメールクライアントを使い分けても同じように使えるように，IMAP プロトコルによるアクセスを基本としている．IMAP サーバとしてのスケーラビリティを考慮して，フリーの IMAP サーバではなく商用の SIMS (Sun Internet Mail Server) を用いている．

2. アカウント管理と認証

教育用計算機システムの利用者は以下のような範囲になっている．

- 学部学生，大学院生
  学生証番号に基づいたユーザ名をつけて自動登録（授業などと関連づけて登録されるわけではない）
- 教員
  申請に応じて登録．所属に基づいた一文字を先頭にした任意のユーザ名．授業を目的とした利用者の他に，メールサーバのみの利用者もいる．
- 職員等申請に応じて登録．所属に基づいた一文字を先頭にした任意のユーザ名．メールサーバのみの利用が多い．

学生，教員は NC, Windows, Unix の 3 種類の環境を使うことができる．これらの環境でアカウント名やパスワードが異なると，ユーザに混乱を与えることになるので，統一する．パスワードの

⋆ 英語 Windows 環境では以前からあった

⋆⋆ Unix システムや，メールサーバはインターネットから使うことができるが，主に想定している使い方ではない

⋆⋆⋆ NC-OS 自体には IC カードを用いた認証の機能があるが，IC カード発行の手間，コストなどを考慮して，IC カードリーダのない NC を採用した

変更インタフェースを限ることにより，複数のパスワード認証システムを使用することは可能だったが，ネットワーク上での認証システムとして実績のあるNISを3種類のシステムの認証に用いることとした

職員は，NCは利用できるが，Unix, Windowsシステムの利用はできない．これは，NISのドメインを複数作成し(1つのNISサーバで管理できる)，1つのマスターファイルからフィルターを通して複数のドメインの map を作成する方法を取る．Unix(SunOS 5.6)では，NISにおる認証はデフォルトで行うことができるが，その他の環境では工夫が必要になる．

- NC
  NCの認証はデフォルトでは，ブートサーバ上の pcnfsd でおこなう．pcnfsd がNISの認証をおこなえば認証できるが，ブートサーバである Unix システムの利用する NIS ドメインと NC 用の NIS ドメインが異なっているので，pcnfsd に手を加えないと使えない．そのため，NCの開発元に直接 NIS 認証を行うよう変更してもらった．
- Windows
  Windows NT TSE には元々，NISを使ってlogin認証をおこなう機能は用意されていない．NCD Wincenter を付加すると NIS validation 機能が使用可能になるので，そのようにした．
- メールサーバ
  メールサーバのソフトウェア SIMS は認証にLDAP(Lightweight Directory Access Protocol) を使っている．SIMS 3.5 付属の LDAP サーバには NIS サーバに直接問い合わせる機能がないため*，ldapsync コマンドを使って一日一回 NIS 用の元データを LDAP のデータベースに変換する作業をおこなっている

NISは，以下のような欠点を持つ**．

(1) セキュリティレベルが低い
Unix のパスワードは暗号化されているものの，辞書アタックや専用ハードウェアに対しては十分な強さをもっているとは言えない．したがって，暗号化されたパスワードであっても，知られるのは危険である．NISでも，特定のIPアドレスの root 権限を有するクライアント以外からの要求しか受け付けないよう設定は可能だが，IPアドレスの偽造，盗聴に対しては無力である．

(2) アップデートのコスト
NISでは，マップの一部だけに変更があった場合も，マップ全体を再構成する必要がある．NISでは，サーバのレプリカデータを持つスレーブサーバを置いて冗長性を上げることができるが，マップ全体を再送する必要がある．

1は深刻な問題である．特にユーザの目の前にあるNCもNISクライアントになるので，ユーザの持って来たノート PC にケーブルを繋ぎ変えて，同じIPアドレスに設定して通信できると，暗号化されたパスワードが取られてしまう．これに関しては，NCに繋がるケーブルは直接 switching HUB の1ポートに繋がり，そのポートにはMACアドレス登録によるポートセキュリティを設定するという方法で対処している．隅々まで自分で管理しているネットワーク内でしかNISは使ってはいけない

2に関しては，アップデートの頻度が高いパスワードマップに関しては1日1回だけ map の make をするようにした．また，スレーブサーバは1台しか置かない方式にした．2は NIS+ サーバに NIS サーバを兼ねさせることにより，ある程度解決は可能だが，NIS+ サーバが十分枯れていないということで，基幹として使うには不安が残り，今回は見送った．

NISで配布可能なユーザ情報としてグループがあるが，これは Unix 環境でしか使っていない．現在以下のようなグループ分けをしている．

- student
  学生を対象．
- teacher
  教官を対象．特定の教室の端末に login しているユーザを表示するコマンドなど，このグループに属するユーザしか実行できないコマンドがある．
- nconly 職員を対象．nconly グループは Unix

---

* 今後のバージョンでは改善されるとの話

** 今回は SunOS 上の NIS サーバの実装の問題にも悩まされた．NIS サーバで使われる dbm ライブラリはハッシュ値の衝突が頻発して，ページサイズを超えた場合は要素を入れられなくなるというバグ (あるいは仕様) があるが，3万5千ユーザ程度登録した際に，この制限により登録できなくなる問題に直面した．パスワードの GECOS フィールドを空にして，レコード長を短くすることにより解決した

は利用できないため，専らフィルタープログラムなどで使われるのみ．

ユーザ登録作業などは，導入業者の提案したDEVIASというシステムを使っている．これはUnix のNISサーバからデータを取ってきて，独自のデータベースを合わせてGUIプログラムで操作してからサーバに書き戻して，NISのmakeをさせるというものである．

3. ファイル共有

NC, Unix, Windowsそれぞれの環境で，個人用のファイル領域が必要になる．

Windows環境でのファイル共有法としては一般的にはSMBプロトコルが用いられる．Windowsサーバの数は25なので，トラフィック的にもファイルサーバ上でSMBプロトコルに対応する選択肢もあったが，管理者の過去の経験を生かせるNFSのみによる運用方式を取った．

Windows NT用のNFSクライアントソフトは多数あるが，Terminal Server Edition用のものは少なかったが，とりあえず仕様を満たすAT-NFSという製品を使っている．ただし，ファイルのロックなどで微妙な問題は生じている．

Windows上で作成したファイルは，漢字コードの場合も多いし，スペースを含むファイル名もよく使われる．Unix上でもこれらには対応可能だが，漢字コードを含むファイル名に関しては，よく使われるツールにも対応しないものもあるので，Windows用のユーザファイルを見せるのはユーザの混乱の元になる．また，Windows上ではファイル操作がGUIで初心者に容易になっているだけに，Unixの設定ファイルを誤って消す可能性もある．

そこで，ユーザのホームディレクトリをUnix用とWindows用に分け，Unix，Windows上で両方のホームディレクトリ以下を参照可能なものの，互いの環境用のホームディレクトリには書き込みできないようにした．

ホームディレクトリ用のディスクは，ハードウェアで18GBのディスク5台単位でRAID5を実現している．これを単位にして，約70GBのファイルシステムを約1000人分のホームディレクトリに使う．ファイルシステムは，パーフォーマンスを重視して，Veritas File Systemを使っているが，ユーザが小さいファイルを多く作ることを考慮して，標準的な8KBのブロックサイズではなく1KBのブロックサイズにしてある．

多人数でファイルサーバを共有する場合はquotaによる容量制限が不可欠である*．

そのため，Unixホームディレクトリ，Windowsホームディレクトリ合わせてのディスク容量を制限している．制限値は現時点では学生50MB，教官200MBである．

quotaの制限を逃れるために，友人のホームディレクトリの下に自分が書き込み可能なディレクトリを作って，そこの下に自分のファイルを作るというユーザが前システムから存在していた．これを防ぐために，

- 自分のホームディレクトリの存在するファイルシステム以外のquota値を0にする．

という対策を行っている．

Veritas File Systemを使っているということから，NFSクライアントからのquota要求に対応していないという欠点がある．

UnixやWindowsアプリケーション等では，書き込みが失敗したかどうかのチェックをきちんと行っていないものがある．そのため，ディスク容量が溢れても，ユーザが気がつかないことよくある．また，よくあるのは設定ファイルを上書き保存しようとして，空のファイルを作ってしまうことである．

WWWブラウザ，メールクライアントなどよく使われるツールの一部に関しては，起動時にquotaが溢れているかどうかをチェックして，溢れている時は設定ファイルをchmodして書き込めなくするという対策をしている．

4. ログ管理

教育用計算機システムでは以下のような観点からログを管理することが必要になる．

- 課金
  教育用計算機システムは，過去の計算機センターのような接続時間，CPU使用時間に応じた課金はおこなっていないが，ユーザの所属部局に対して実ユーザ数(1年間に一度でも使ったユーザ)に応じた利用負担金を1年単位で請求している．そのため，利用の記録を取る必要がある．
- 障害への対処

* 容量を監視して，個人的に注意するという方法は大規模システムでは限界がある．

ユーザ固有の障害 (パスワード変更のミス) などに素早く対応するために，パスワード変更などの記録は取っておく必要がある．

- 利用者の特定
  システムに対して害を及ぼす行為をしたり，不適切な情報発信をおこなったりした場合，特定の時間に特定のマシンを利用していたのが誰かを調べることにより特定できるようにしたい．

各種サーバ類 (WWW サーバ，WWW proxy サーバ，メールサーバ) などについては標準的なログを残し，Unix や Windows についても OS で提供されるログを残す形で十分だったが，NC の利用記録に関しては問題があった．

NC-OS 上では，PC-NFSD によるログイン認証がデフォルトになっている．認証に PC NFSD を利用するときは，認証作業を行う際に PC NFSD により記録される．しかし，本システムでは，

- 直接 NIS で認証するため，通信相手は NIS サーバだけで，認証の成功，失敗の記録も残らない．
- ログインの記録だけでなく，どの時点でログオフしたかの記録も取りたい．

ということから，別の記録方法を取る必要があった．

NC 側に新たなサーバ等を導入するのはセキュリティホールとなりうるので，SNMP を利用して，ログを取ることにした．NC は，SNMP に関しては以下のような機能を有している．

- login, logoff, poweron, poweroff 時にサーバに trap を送る
- snmpd が各種の情報 (login ユーザ名等) に答える．

当初は trap のみでログイン記録を取ることができると思われたが，poweroff 時にトラップを送るのが，ログイン画面からシャットダウンを選んだ場合だけで，いきなり NC 本体の電源スイッチを操作した時にはトラップを送る暇もなく電源が切れてしまうことがわかった．

そのため，ブートサーバである Unix サーバ上に 30 秒に一度，login ユーザを snmp で問い合わせて，変化が生じたときに /var/adm/wtmpx に書き込むサーバプログラム nclogd を作成した．このプログラムは CMU SNMP ライブラリを利用したので，新たに書いたのは 400 行ほどである．[⋆]

/var/adm/wtmpx に残す以外に，以下のような要求があるため，現在の利用者を知るインタフェースを用意した．

- 授業の出席を演習室での NC の利用で取る．
- 各演習室の混雑状況を見て，自習用に開室する部屋を決める．

このような目的には，finger やその拡張版である GNU finger，あるいは rwho などが用いられることが多いが，Unix サーバ上で走る nclogd が 1 台の管理用マシンで動くサーバに通信して情報をまとめるという方法を取った．このサーバプログラムは Java で記述したマルチスレッドプログラムで，100 行ほどである[⋆⋆]．

このサーバを利用したコマンドは教官のみに使用を許しているが，情報をプライバシーに関するガイドラインを満たすクライアントを作る学生には伝えている[⋆⋆⋆]．

5. WWW インタフェース

Unix を利用した uniform なシステムでは，

- passwd コマンドによるパスワード変更
- /.forward の記述によるメール転送の指定

などが，ユーザに抵抗無く受け入れられるが，授業の受講者以外特に Unix 環境を使う必要のない本システムでは，この手のユーザ認証を必要とするサービスは，どの環境からも (教育用計算機システム外部からも)WWW インタフェースを通じて受けられるようにした．

本人確認のために教育用計算機システムを入力させることがあるため，途中の経路を考えると暗号化が必須となる．そこで，SSL を使った通信で接続するようにしている．CGI を使う際に，パスワード等は POST で行うなどは常識である．現時点で，以下のようなサービスを WWW

[⋆] このサーバを走らせた当初，ネットワークに大量のブロードキャストが発生し，tftp が fail することがあるという症状が出た．原因を調べてみたところ，NC の電源が入っていない時に，その IP に向けて SNMP の問い合わせをするために arp を発生していたためと分かった．NC の arp 表をスタティックに設定することにより解決した．

[⋆⋆] Java 言語でサーバを作成する時に気になるのは，速度とメモリだが，このサーバプログラムは 7ヶ月連続運用して，CPU 時間は 550 分程度，使用メモリも当初 8MB 程度だったものが 10MB に増えた位で，問題なく動いている

[⋆⋆⋆] 使用中の友達を捜すというツールは，禁止していても学生は作成するし，コントロールしないとネットワークや CPU に負担をかける粗悪なものが広まってしまうので，コントロールの効く範囲で作成を認めている．

で利用可能にしている．

- パスワード変更
  導入業者作成の perl script．内部では Unix の passwd を使っている．NIS map のアップデートを1日1回に限っているため，このインタフェースでも1日1回以上の実行を防ぐ仕組みを入れてある．
  Unix の passwd コマンド，Windows のパスワード変更機能は共存可能ではあるが，使用不能にしてある．*
- メール転送
  SIMS に用意されているユーザ設定用の WWW インタフェースのうち，メール転送だけを公開している．
- ファイル容量の確認
  パスワード認証を終えた後に，ファイルサーバに対して rsh をかけ，それを見易いように変換するセンター作成の Perl スクリプトである．
- プリンタ出力記録
  パスワード認証を終えた後に，プリンタのログを取っているサーバに対して rsh をかけ，それを見易いように変換するセンター作成の Perl スクリプトである．
- WWW 情報発信実験参加申し込み
  パスワード認証を終えた後に，WWW サーバに対して rsh をかけ WWW サーバにユーザを登録する．同時に，申込受け付けのメールを送る．

これらのインタフェースは Perl で簡単に書けるので，必要に応じて機能を付け加えるのは容易である．しかし，安易に作成するとセキュリティホールとなりうるので注意が必要である．

6. プリンタ出力制限

NC, Unix, Windows の3つの環境から出力する関係でプリンタは Postscript プリンタと決まった．

プリンタの出力する紙の枚数を数えるためには，プリンタのジョブを一旦サーバに spool しておいて，そこで枚数を数える方法が考えられる．しかし，Postscript プリンタの場合，Postscript エンジン以降でないと出力枚数を正確には決定できない．今回のプリンタは Postscript エンジンがプリンタ側にあるため，この方法は使えなかった．

そこで，ヘッダに setjobinfo をつけてプリンタ内のハードディスクに記録する方法を取った．メーカの協力により，こちらの用意したプログラム以外でヘッダを偽造したジョブを作成しても印刷しない仕組みも導入した．

ハードディスクに蓄えられたログは，1時間に1回，あるサーバが収集する．そのログを Perl で書かれたスクリプトで集計し，ユーザごとの月と年の出力の表を作成する．

プリンタの使用量は，費用の問題と教育用効果から新システムから制限するという方針になった．学生ユーザは月 50 枚，年 300 枚の出力制限値がデフォルトになっている．

講義等で学生の制限値を緩めたいという希望がある場合は教官からの申請で，ユーザ名のリストを送ってもらい制限値を緩めることも可能になる．

1時間に1回のログの集計で，出力制限値を超えたユーザのリストができるが，これは NIS サーバに転送される**．NIS の printer.deny という map でこのリストを全 Unix サーバに配布する．NC, Windows, Unix のすべてのプリント要求はすべて一旦 Unix サーバに蓄えられるが，そのサーバの「/etc/lp/interfaces/ プリンタ名」のスクリプトの中で printer.deny をチェックして，ジョブのユーザ名が map に含まれている時は，ジョブをクリアする．

7. ま と め

4月から運用を始めた新システムは個々の問題は生じているが，おおむね順調に動いている．このシステムは他の教育機関におけるシステムの設計においても，参考になる点が多いと思う．

参 考 文 献

1) 安東孝二，吉岡顕，田中哲朗: 大規模計算機センターのセキュリティ対策事例，情報処理学会分散システム / インターネット運用技術研究会報告 DSM16-3(1999).

* 使用不能にしたつもりが，Sun の recommended patch を当てた結果，いつのまにか使えるようになっていたという事件もあった

** 制限値を超えた時には，ユーザにメールを自動的に出して知らせる