

学生用ファイルサーバのユーザ管理についての一試み

A Study on User Management of Server for Student

瀬戸 博幸

Hiroyuki Seto

1. はじめに

鹿児島女子短期大学のインターネット関連のサーバが2006年9月に更新され、その概要についてはすでに報告した。⁽¹⁾今回は、学生用ファイルサーバへの利用者登録の自動化について検討を行ったので、以下それについて報告する。

2. 学生用ファイルサーバ

鹿児島女子短期大学には児童教育学科、生活科学科、および教養学科に一学年およそ500名の学生がおり、教養学科の学生およそ80名は2年間を通して一人に一台のノートパソコンが貸与されているが、他の学科の学生は60台のデスクトップコンピュータ教室を共用している。教養学科以外の学生には半年または一年間、週に1コマ程度の情報機器関連の演習が課せられているが、60台のコンピュータには教室共用を考慮し環境復元ソフトを導入しているため、処理結果を保存するストレージの確保が大きな問題となった。つまり、各コンピュータには各自のストレージ用としてMOドライブを設置しており、USBインターフェースも存在するので、演習の処理結果を保存するためにはMOメディアまたはフラッシュメモリ等を各自用意すれば簡単に解決すると想定していたが、週1度の演習に際してメディアの持ち込み忘れ、退出時の放置などがかなりの頻度で発生し、それらの学生は演習に支障をきたす結果となった。

そこで、2006年の更新時に学生用サブネットのDHCPサーバにLinuxとSambaを用いて学生用ファイルサーバを構築し⁽³⁾、Windows XPの学生用コンピュータからは、各自のユーザIDとパスワードを使用し、ネットワークドライブの割り当てを行うことで、MOメディアやフラッシュメモリを必要としない各自専用のストレージの確保が可能となった。また、同サーバにApacheを用いてWebサーバを構築し、学生各自は自分のストレージにWeb専用のフォルダを確保することで簡単に学生各自のホームページを公開することも可能となった。⁽¹⁾

このWebサーバはコンピュータ教室を共用している学生だけでなく、ノートパソコンを貸与され各自のストレージには困らないであろう教養学科の学生にも、ウェブデザイン実務士資格の必須項目であるホームページ公開の手段として、また、簡便で安全なファイル交換の手段として有効である。

3. ファイルサーバのユーザ管理

このように学生にとって有効な学生用ファイルサーバであるが、サーバ管理者には大きな負担が存在する。それはユーザの登録やメンテナンスなどのユーザ管理である。

学生用ファイルサーバを利用するためには、Linux ユーザおよび Samba ユーザとして各学生を登録しなければならない。一般にユーザ登録するには

- ①groupadd コマンドでグループ名やグループ ID をシステムに登録
- ②useradd コマンドでユーザ名やユーザ ID をシステムに登録
- ③chmod コマンドでユーザのホームディレクトリのアクセス権を調整
- ④passwd コマンドでユーザの初期パスワードを設定

以上で、Linux ユーザとして登録が完了し、ユーザはシステムを利用できるようになる。

しかし、上記は手作業で1ユーザを登録する方法であり、500名もの新規ユーザを上記方法で登録するには時間もかかり、当然間違いの発生する可能性も高くサーバ管理者にとって大きな負担となっていた。

よって、ユーザ登録を一括して行うシェルスクリプト（以下スクリプト）の作成を試みたので以下、それについて報告する。

4. ユーザを一括登録するためのスクリプトの作成

今回作成したスクリプトを図1に示す。

```
#!/bin/sh
## このスクリプトの原型は Aaron Malone 氏が作成し, Meredydd Luff, Peter Samulson, Kathryn Hogg
## の各氏が修正したものを参照した(2)が, 今回の試み用に, 瀬戸が改変を加えた.
## このスクリプトは Free Software Foundation の発行する GNU General Public License の条件で
## 再配布や改変を行ってもかまわない.
## 使用法:
## #sh mass_useradd < 入力ファイル
## 入力ファイルのデータ形式は 1 行に ユーザ I D:ユーザの氏名 のテキストファイル
PATH=/usr/local/sbin:/usr/sbin:/sbin:/usr/local/bin:/usr/bin:/bin:$PATH
## カンマで区切られたデータ形式の場合は IFS=":$IFS"を IFS=",$IFS"に変更する
while IFS=":$IFS" read username realname; do
    case "$username" in
        ''|\#*) continue;
    esac
    id=$( { getent passwd; getent group; } | cut -f3 -d: |          ## 改変した
        awk ' max<$0 { max=$0 }                                ## 改変した(4)
            END { max++; print max }' )                          ## 改変した
    print $id                                                    ## 追加した
    groupadd -g $id $username
    useradd -m -c "$realname" -g $username -u $id $username
    chmod 700 /home/$username
    ( echo "$username:password" | chpasswd )                      ## 追加した
done
```

図1. ユーザー一括登録用スクリプト

図1のスキプトの動作を図2の流れ図で示す。

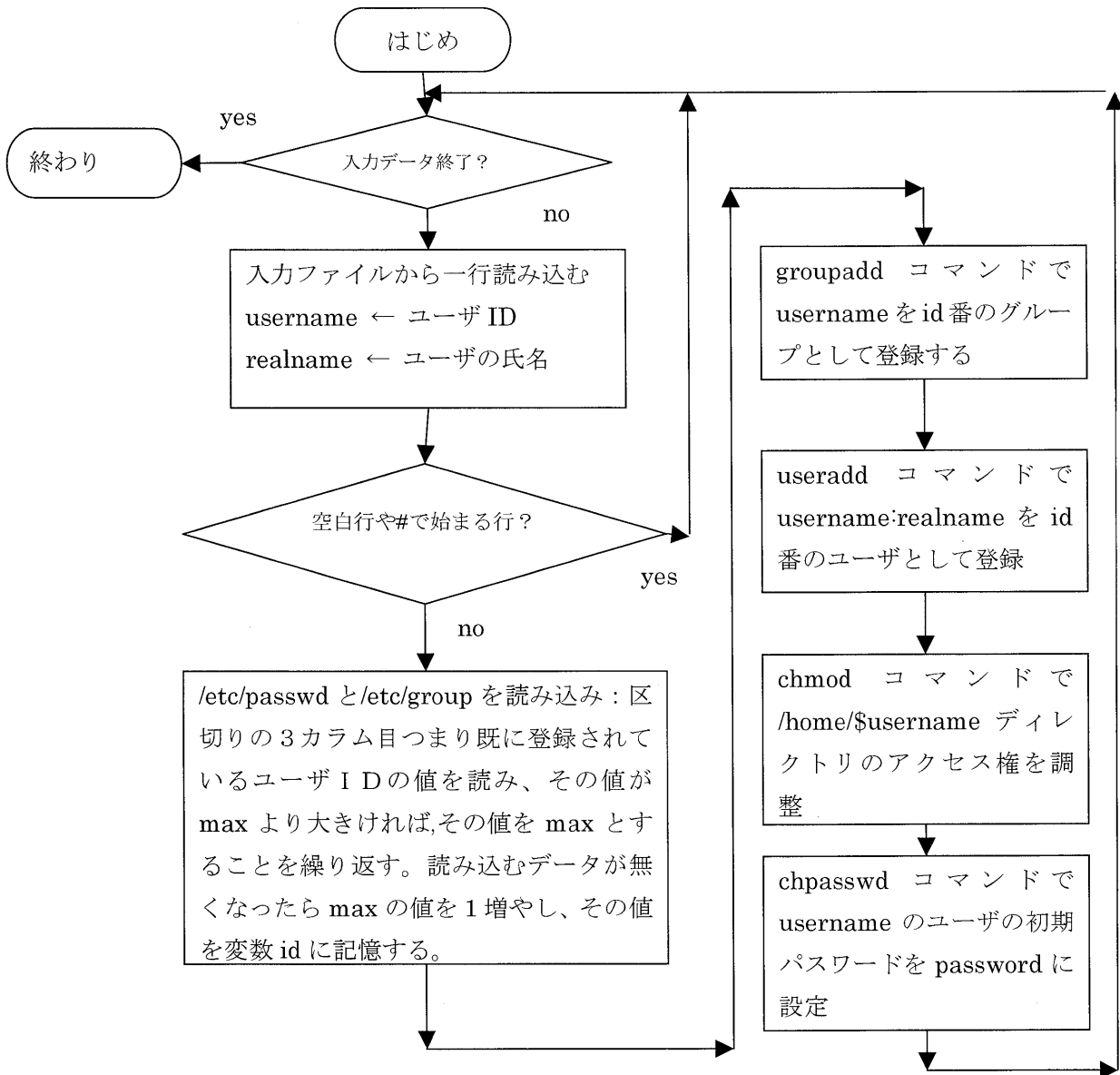


図2. スクリプトによる処理の流れ

なお、スクリプト内の chpasswd コマンド⁽²⁾ は passwd コマンドの一括処理版である。chpasswd コマンドの標準入力 は ユーザ ID : パスワード を一行の入力形式として要求するが、今回作成したスクリプトではスクリプト内で、その一行を作成しパイプを通して chpasswd コマンドに供給した。この chpasswd コマンドにより /etc/shadow に暗号化されたパスワードが作成され、ユーザはユーザ ID で学生用サーバにログインできるようになる。

5. 考察

このスクリプトを格納したファイル (ファイル名 mass_useradd) と実験用登録ユーザデータを、ユーザ ID : ユーザ氏名 の形式で格納した入力ファイル (ファイル名 username) を root のディ

レクトリに置き、

```
#sh mass_useradd < username
```

のコマンドを管理者権限で実行することで、1 ユーザあたり 1 秒以内の速度で登録が完了することを確認した。また、登録したユーザ ID と初期パスワード password を用いてユーザとしてログイン可能であることも確認した。

実験環境は Vine Linux 3.3.6-0v17, カーネルは Linux version 2.6.16-0v168である。

結果、500名の登録も上記コマンド一回の実行でおよそ10分以内に完了することが推測され、管理者の負担を十分軽減できると考えている。

さらに、登録用ユーザデータの作成であるが、入学時にユーザ ID とユーザ氏名の入ったテキストファイルを mail サーバへの登録用として作成している。この作業は学生部から入学生のデータを受け取り、Excel を使ってユーザ ID, 氏名その他 mail サーバに必要なパラメータを作成し、1 学生 1 行のデータとして csv 形式でテキストファイルに格納したものである。このファイルを使い、ユーザ ID とユーザ氏名を必要な形式に切り出すスクリプトを作成することで上記 username ファイルの作成は自動化できる。

つぎにユーザ管理のセキュリティについて述べる。ユーザ登録に際して初期パスワードを password としている点について、passwd コマンドを用いて一件ずつユーザを登録する場合、この password という値は『辞書に載っている簡単な値』として passwd コマンドによってパスワードの値として使うことを拒否される。つまりセキュリティ的に脆弱な値である。これを回避するために、たとえば z1x2c3v4などの値をランダムに発生し登録に用いればよいが、今回の場合は学生用ファイルサーバへのユーザ登録ということで password という値を採用した。(chpasswd コマンドはこの値を拒否しない)

その理由として学生用ファイルサーバはファイアウォールにより外部ネットワークから隔離されており、使用に際しては一斉授業でオリエンテーションを行いパスワードの変更を行うことができるため、セキュリティの脆弱性よりも一斉授業での説明のしやすさを考えてのことである。また、password の値はスクリプト mass_useradd のファイルに一元管理されており root のディレクトリに置くことで管理者として相応のセキュリティは確保できると考えたためである。しかし、一般的なサーバ管理として常にセキュリティの管理を行う必要があることは言うまでもない。同様に username ファイルも個人情報を含んでおり管理に気をつけなければならない。

今後について言えば、username ファイルを年度ごとに管理しておくことで、卒業後一定期間を過ぎサーバ利用が必要でなくなったユーザについて、年度毎に管理された username ファイルを用いて、一括してユーザを削除する処理も userdel コマンドを使い、図 1 に示したスクリプトを少し改変することで作成可能と考えている。

また、Linux の仕様としてユーザ ID は約65000番までしか管理できない。ユーザ数を考えると、長期使用でも問題なく使えることが予測できるが、この問題はスクリプトを調整して解決すべきか、ユーザ ID 使用に関するポリシーを策定することで解決すべきか今後様子を見ながら検討してゆきたい。上記スクリプト群をまとめユーザ管理ユーティリティを充実してゆくこともユーザ

管理の今後の方針となろう。

6. まとめ

学生用サーバに大量のユーザを登録するというユーザ管理の課題を，シェルスクリプトを作成する方法により解決することで，現実的・具体的な課題として捉えることができ今後の方向性も見えた。今後はユーティリティを充実し，どのような管理ポリシーが最適か検討してゆきたい。

参考文献

- (1) 瀬戸博幸 鹿児島女子短期大学インターネットサーバ更新の報告
鹿児島女子短期大学「紀要」第42号 (2007) pp.193-198
- (2) LINUX クックブック Carla Schroder 著 林 秀幸訳
オライリージャパン，オーム社 ISBN4-87311-248-6
- (3) LINUX ネットワーク管理 Tony Battus 他著 高橋浩和他監訳
オライリージャパン，オーム社 ISBN4-87311-247-8
- (4) AWK の第一歩
<http://lagendra.s.kanazawa-u.ac.jp/ogurisu/manual/awk/intro/awk-intro.html>

(2007年12月5日 受理)