

2007/3/16



2007 年 3 月 16 日開催
オープンソースカンファレンス 2007 Tokyo/Spring
Linux ハンズオンセミナー

『Linux における
ユーザ管理の仕組みを学習しよう！』

日本電子専門学校

杉松秀利

申 載雄

sugimatsu@mail.pcmarks.jp

sin@jec.ac.jp

Copyright(c) : Japan Electronics College All Rights Reserved.

Linux におけるユーザ管理の仕組み

- 本日のハンズオンセミナーでは、Linux システムではどのような仕組みを使って、ユーザ管理を行っているのかということについて学習します。
- なお、本資料は、CentOS 4.4とDebian GNU/Linux 3.1を使用して検証した結果に基づき作成しました。
- 本日は、CD のみで起動する Linux である、knoppix を使用してハンズオンセミナーを行いたいと思います。

準備：Linux の起動

- 1) PC の電源を入れて、CD をトレイに入れ、リブートしてください。
- 2) 左下に `boot:` と表示されたら、`knoppix 3` を入力してください。(30 秒以内に入力しないとデフォルト起動動作に移行されますので、最初のkの文字だけでも、まず入力してください。)
- 3) ターミナルのプロンプトの最後の記号(# または \$)によって、root 権限か一般ユーザか分かります。root ユーザでログインされたか確認してください。X-Windows が起動された場合は、一般ユーザでログインされていると思いますので再起動しなおしてください。

[~]#: root 権限 [~]\$: 一般ユーザ権限

I ユーザ管理用のファイルとディレクトリ

資料の前半部分では、Linux システムにおいて、ユーザ管理用に使用されている主要なファイルとディレクトリについて学習します。前半部分のキーワードは、**3つのファイルと1個のディレクトリ**です。

1. これがユーザ情報を管理するファイルとディレクトリです！

Linux システムにおいて、ユーザ情報がどのように管理されているのかを理解して頂くために、本日のセミナーでは次の3つのファイルと1個のディレクトリについて学習することにします。

◇ ユーザ管理に使用される主要なファイルとディレクトリ

ファイル名など	機能	適用
<code>/etc/passwd</code>	各ユーザアカウントの様々な情報が記録されているファイル。1行に1エントリずつ定義されている。	パスワードファイル
<code>/etc/shadow</code>	ユーザのアカウントに対する暗号化されたパスワード情報、およびオプションとしてパスワードの有効期限の	暗号化されたパスワードファイル

	情報が記されているファイル。1 行に 1 エントリずつ定義されている。	
/etc/group	グループにどのユーザーが所属しているかが定義されているファイル。1 行に 1 エントリずつ定義されている。	ユーザグループファイル
/etc/skel/	ホームディレクトリを作成する際にコピーされるファイルやディレクトリが置かれているディレクトリ。	ファイルの雛形が置かれているディレクトリ

注：上記のファイルのほかに、各グループの暗号化パスワードや、グループのメンバーシップなどの情報が記録された/etc/gshadow ファイルがあります。

2. 新しいユーザを作成するには

- それでは、実際に新しいユーザを作成して、これらのファイルがどのような役割を果たしているかを調べてみることにしましょう。
- 新しいユーザは、useradd コマンドを使って作成します。

(1) RedHat 系 Linux で新しいユーザを作成する場合には、次の構文を使用します。

```
# useradd 新規ユーザ名
```

(2) 新しいユーザを作成する操作例

```
[root@ws101 ~]# useradd sugimatsu
```

(3) Debian GNU/Linux(以下 Debian と言います)の場合には、次の構文を使ってください。

```
# useradd -m 新規ユーザ名
```

注：-m は、新規ユーザを作成する際に、もしホームディレクトリが存在しない場合には、同時にホームディレクトリを作成することを指定するオプションです。

(4) RedHat 系の Linux では、useradd コマンドの代わりに、adduser コマンドを使うこともできるようになっています。

```
[root@ws101 ~]# adduser 新規ユーザ名
```

注：adduser コマンドは、useradd コマンドのシンボリックリンクなので、使用する際のオプションなどはすべて同じです。

◎ RedHat 系 Linux と Debian では新規ユーザを作成する構文が違うのは何故？

- RedHat 系 Linux では、/etc/login.defs ファイルの中で、CREATE_HOME オプションが yes に設定されており、useradd コマンドの実行時に、デフォルトの設定として、新規ユーザのホームディレクトリが作成されるようになっています。
- これに対して、Debian の /etc/login.defs ファイルでは、CREATE_HOME オプションの定義がないため、新規ユーザを作成する際に、同時にそのユーザのホームディレクトリを作成したい場合には、-m オプションを指定して useradd コマンドを実行しなければなりません。
- つまり、2 種類のディストリビューションでは、/etc/login.defs ファイルにおける CREATE_HOME オプションの取り扱いが異なるために、新規ユーザを作成するための構文が違っているのです。

3. /etc/passwd ファイルの内容はどうなっているの？

- 新規のユーザが作成されたところで、それぞれのファイルの内容を調べていくことにしましょう。
- まずは、/etc/passwd ファイルです。

(1) /etc/passwd ファイルのパーミッション

実際にファイルを開く前に、/etc/passwd ファイルのパーミッションを確認しておきましょう。

◇ /etc/passwd ファイルのパーミッションを確認する操作

```
[root@ws101 ~]# ls -l /etc/passwd
-rw-r--r--  1 root root 2269  3月  9 13:13 /etc/passwd    ← パーミッションの表示
```

◎ 全ユーザに読み取り権限が与えられているのは何故？

- 書き込み権限は所有ユーザのみに設定されていますが、すべてのユーザに読み取り権限が与えられていることに注意しましょう。
- このようなパーミッションの設定になっているのは、/etc/passwd ファイルに記録されている情報を、さまざまなプログラムが利用できるようにするためなのです。

(2) /etc/passwd ファイルの内容を表示してみよう

- それでは、今作成したばかりの新規ユーザの情報が、/etc/passwd ファイルの中に、どのように記録されているのかを、確認してみることにしましょう。
- 今回は、-N オプションを指定した less コマンドを使って、/etc/passwd ファイルの内容を表示してください。
- なお、-N は各行の先頭に行番号を表示するために指定するオプションです。

① /etc/passwd ファイルを-N オプション付きの less コマンドで表示する操作

```
[root@ws101 ~]# less -N /etc/passwd
```

② /etc/passwd ファイルの表示例

```
42 dovecot:x:97:97:dovecot:/usr/libexec/dovecot:/sbin/nologin
43 postfix:x:89:89::/var/spool/postfix:/sbin/nologin
44 mailman:x:41:41:GNU Mailing List Manager:/usr/lib/mailman:/sbin/nologin
45 mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/bash
46 radiusd:x:95:95:radiusd user:/bin/false
47 sugimatsu:x:500:500::/home/sugimatsu:/bin/bash
```

(3) /etc/passwd ファイルの書式

- /etc/passwd ファイルでは、1 行に 1 エントリずつのユーザなどの情報が記録されており、各行はコロン(:)で区切られた 7 つのフィールドによって構成されています。
- 各フィールドに格納されるユーザ情報は、次のとおりです。

◇ /etc/passwd ファイルのエントリの記述例

```
sugimatsu:x:500:500::/home/sugimatsu:/bin/bash
```

① ② ③ ④ ⑤ ⑥ ⑦

◇ 各フィールドに格納される情報

	名称	格納される情報
①	第 1 フィールド	ログイン名
②	第 2 フィールド	パスワード
③	第 3 フィールド	ユーザ ID
④	第 4 フィールド	グループ ID (プライマリグループ)
⑤	第 5 フィールド	ユーザ名もしくはコメント
⑥	第 6 フィールド	ユーザのホームディレクトリ
⑦	第 7 フィールド	ユーザのデフォルトシェルプログラム

注：各フィールドに格納される情報から、/etc/passwd ファイルは「ユーザ情報のデータベース」である、とされています。

(4) 第 2 フィールドに格納される値

- 第 2 フィールドには、以前はユーザのパスワードが平文で格納されていましたが、現在ではパスワードは暗号化されるようになりました。
- そして、暗号化されたパスワードは/etc/shadow ファイルへ格納されるようになっています。
- このため、現在、第 2 フィールドに格納される値は、次の 3 つです。

◇ 第 2 フィールドに格納される値

値	説明
x	<ul style="list-style-type: none"> • パスワードが暗号化され、/etc/shadow ファイルで管理されていることを表します。 • x が格納された passwd ファイルのことを、shadow 化された passwd ファイルと言います。
*	<ul style="list-style-type: none"> • ユーザアカウントを一時的に無効にすることを表します。 • *が設定されている間、そのユーザはログインできなくなります。
ヌル	<ul style="list-style-type: none"> • ヌル(null)とは、長さ 0 の文字列のことです。 • 何も値が設定されていないため、sugimatsu::500 というように、ダブルコロン (::) で表示されます。 • ヌルに設定した場合には、そのユーザはパスワードの入力なしでログインできます。

注：shadow 化された/etc/passwd ファイルでは、ユーザにパスワードが設定されている/いないに関係なく、第 2 フィールドには x という値が格納されています。

- 以前は、各ユーザのパスワード情報も、このファイルに格納されていました。
- しかし、たとえ暗号化されているとしても、すべてのユーザが閲覧できるファイルに、パスワードを格納しておくことは、セキュリティ対策上、決して好ましいことではありません。
- そこで、ユーザのパスワードに関する情報は、/etc/shadow ファイルで管理されるようになったのです。

(5) 第 3 フィールドに格納される値

- 第 3 フィールドには、ユーザ ID (UID) を表す値が格納されます。
- また、第 1 フィールドに格納されるログイン名は、root、システムアカウント名、一般ユーザのアカウント名に分類できます。
- この第 1 フィールドに格納されるログイン名の種類と、それぞれの種類のログイン名に割り当てられるユーザ ID の値の関係は、次のように整理できます。

◇ ログイン名の種類と割り当てられるユーザ ID の関係

ログイン名の種類	格納される値
root	0
システムアカウント	1 ~ 499
一般ユーザのアカウント	500 ~ 60000

◎ 用語の説明

アカウント(account)

Linux システム上で、ソフトウェアなどを扱うことのできる権限のことを、アカウントと言います。アカウントには、ユーザアカウントとシステムアカウントの 2 種類があります。なお、2 種類のアカウントとも、ユーザ ID によって識別されるようになっています。

ユーザアカウント(user account)

ユーザが Linux システムへログインして、システムを操作できる権限のことを、ユーザアカウントと言います。ユーザアカウントには、パスワードが設定されるとともに、それぞれのユーザのログイン用のディレクトリとして、ホームディレクトリが作成されます。通常の場合、単に「アカウント」と言う場合には、ユーザアカウントを指します。

システムアカウント(system account)

Linux システムが、特定のアプリケーションを実行するために利用する、具体的なユーザの存在しないアカウントのことを、システムアカウントと言います。システムアカウントには、特定のユーザが存在しないため、パスワードを使って Linux システムへログインする権限はなく、ホームディレクトリも作成されません。

① 一般ユーザのユーザ ID に割り当てられる値の範囲について

- 一般ユーザに割り当てられるユーザ ID の開始値と終了値は、`/etc/login.defs` ファイルの `UID_MIN` パラメータと `UID_MAX` パラメータの設定値によって決定します。
- RedHat 系の Linux では、`UID_MIN` パラメータには 500、`UID_MAX` パラメータには 60000 という値が設定されているため、一般ユーザに割り当てられるユーザ ID の値の範囲が、500~60000 ということになっているのです。

② Debian の場合は 1000~60000 です

- Debian の場合には、`/etc/login.defs` ファイルの `UID_MIN` パラメータには 1000、`UID_MAX` パラメ

一タには 60000 という値が設定されています。

- このため、一般ユーザに割り当てられるユーザ ID の値の範囲は、1000～60000 になります。

(6) 第 6 フィールドに格納される値

- 第 6 フィールドには、ユーザのホームディレクトリ名が、フルパス付きで格納されます。
- `useradd` コマンドを実行する際に、特別にホームディレクトリを作成するパスを指定しなければ、デフォルトの設定では、`/home/`ディレクトリ下に新規のユーザ名と同じ名称のディレクトリが、そのユーザのホームディレクトリとして、自動的に作成されるようになっています。
- このような処理が自動的に行われるのは、`useradd` コマンドのデフォルトの設定値を定義した `/etc/default/useradd` ファイルの中で、「`HOME=/home`」と指定されているためです。

(7) 第 7 フィールドに格納される値

- 第 7 フィールドには、ユーザのデフォルトシェルプログラム名が、フルパス付きで格納されます。
- `useradd` コマンドを実行する際に、特別な指定をしなければ、自動的に「`/bin/bash`」が格納されます。
- このような処理が自動的に行われるのは、`useradd` コマンドのデフォルトの設定値を定義した `/etc/default/useradd` ファイルの中で、「`SHELL=/bin/bash`」と指定されているためです。

4. `/etc/shadow` ファイルの中を覗いてみよう

- 続いて、パスワードが暗号化された Linux システムでは、`/etc/passwd` ファイルと 1 セットのファイルとなる `/etc/shadow` ファイルの内容について、調べることにしましょう。
- Linux システムでは、暗号化されたパスワードは、`/etc/shadow` ファイルに保存されます。

(1) `/etc/shadow` ファイルのパーミッション

`/etc/passwd` ファイルのときと同様に、ファイルを開く前に `/etc/shadow` ファイルのパーミッションを確認しておきましょう。

◇ `/etc/shadow` ファイルのパーミッションを確認する操作

```
[root@ws101 ~]# ls -l /etc/shadow
-r----- 1 root root 1342  3月 10 04:57 /etc/shadow ← パーミッションの表示
```

- ファイルの所有者である `root` だけに、読み込み権限のみが許可されているファイルであることがわかります。
- つまり、Linux システムによってのみ書き込み可能であり、`root` 権限のみが閲覧可能なファイルであるということになります。
- なお、他のファイルと同様に、`root` 権限によってパーミッションを変更し、`root` 権限に書き込み権限を与えることは可能です。

(2) `/etc/shadow` ファイルの内容を表示してみよう

- それでは、`/etc/shadow` ファイルの内容を、確認してみることにしましょう。
- 今回も、`-N` オプションを指定した `less` コマンドを使って、`/etc/shadow` ファイルの内容を表示して

ください。

① /etc/shadow ファイルを-N オプション付きの less コマンドで表示する操作

```
[root@ws101 ~]# less -N /etc/shadow
```

② /etc/shadow ファイルの表示例

```
42 dovecot:!:13581:0:99999:7:::
43 postfix:!:13581:0:99999:7:::
44 mailman:!:13581:0:99999:7:::
45 mysql:!:13581:0:99999:7:::
46 radiusd:!:13581:0:99999:7:::
47 sugimatsu:!:13581:0:99999:7:::
```

(3) /etc/shadow ファイルの書式

- /etc/shadow ファイルでは、1 行に 1 エントリずつのユーザアカウントに対する暗号化されたパスワード情報が格納されています。
- また、暗号化されたパスワード情報のほかに、オプションとしてパスワードの有効期限に関する情報なども格納されています。
- このファイルも、各行の内容はコロン(:)区切られており、9 つのフィールドによって構成されています。
- 各フィールドに格納される情報は、次のとおりです。

◇ /etc/shadow ファイルのエントリの記述例

```
sugimatsu:!:13581:0:99999:7:::
```

① ② ③ ④ ⑤ ⑥⑦⑧⑨

◇ 各フィールドに格納される情報

	名称	格納される情報
①	第 1 フィールド	ログイン名
②	第 2 フィールド	MD5 により暗号化されたパスワード
③	第 3 フィールド	1970 年 1 月 1 日から起算して最後にパスワードが変更された日迄の日数
④	第 4 フィールド	パスワードが変更可能となるまでの日数(この期間を越えないと、パスワードは変更できない)
⑤	第 5 フィールド	パスワードを変更しなくてはならなくなる日までの日数(この期間を越えたらパスワードの変更が必要)
⑥	第 6 フィールド	パスワード有効期限が来ることをユーザに警告する日数
⑦	第 7 フィールド	パスワード有効期限が過ぎてから、アカウントが使用不能になるまでの日数
⑧	第 8 フィールド	1970 年 1 月 1 日からアカウントが使用不能になる日までの日数
⑨	第 9 フィールド	予約済みのフィールド

注：各フィールドに格納される情報から、/etc/shadow ファイルは「パスワード情報のデータベース」

である、とされています。

(4) 第 2 フィールドに格納される値

第 2 フィールドには、MD5 というアルゴリズムに基づき暗号化されたパスワードが格納されますが、次の 3 つの値も格納されるようになっています。

◇ 第 2 フィールドに格納される値

値	説明
!!	第 1 フィールドのユーザに、パスワードが設定されていないことを表します。
*	ログインできないアカウントであることを表します。
ヌル	ユーザがパスワードの入力なしでログインできることを表します。

注： Unix では、NP と *LK* で表される値に相当するものと推測されますが、実際に /etc/shadow ファイルに格納されている内容を分析してみると、Linux の場合にはこの 2 つの値が厳密に使い分けられていません。

(5) その他のフィールドに格納される値について

- 第 3 フィールドの 1970 年 1 月 1 日から起算して最後にパスワードが変更された日迄の日数については、自動計算された値が格納されます。
- 第 4 フィールドのパスワードが変更可能となるまでの日数については、/etc/login.defs ファイルの PASS_MIN_DAYS パラメータに設定されている値が格納されます。デフォルトでは、0 という値が格納されますが、これはいつでも変更が可能であることを意味します。
- 第 5 フィールドのパスワードを変更しなくてはならなくなる日までの日数については、/etc/login.defs ファイルの PASS_MAX_DAYS パラメータに設定されている値が格納されます。デフォルトでは、99999 という値が格納されますが、これは有効期限がないことを意味します。
- 第 6 フィールドのパスワード有効期限が来ることをユーザに警告する日数については、/etc/login.defs ファイルの PASS_WARN_AGE パラメータに設定されている値が格納されます。デフォルトでは、7 という値が格納されますので、7 日前から警告を発するという設定になります。

5. /etc/group ファイルも解剖しよう

- 今回のハンズオンセミナーで取り上げるユーザ情報を管理するファイルの最後は、/etc/group ファイルです。
- /etc/group ファイルは、グループにどのユーザが所属しているのかを定義するファイルです。

(1) /etc/group ファイルのパーミッション

例によって、まず最初に /etc/group ファイルのパーミッションを確認しておきましょう。

◇ /etc/group ファイルのパーミッションを確認する操作

```
[root@ws101 ~]# ls -l /etc/group
-rw-r--r-- 1 root root 763  3月 10 04:57 /etc/group ← パーミッションの表示
```

◎ 全ユーザに読み取り権限が与えられている

- /etc/passwd ファイルとまったく同じパーミッションが設定されていますね。
- 書き込み権限は所有ユーザにしか設定されていませんが、すべてのユーザに読み取り権限が与えられています。
- /etc/passwd ファイルと同様に、記録されている情報を、さまざまなプログラムが利用できるようにするために、このようなパーミッションの設定になっているのです。

(2) /etc/group ファイルの内容を確認してみよう

- /etc/group ファイルの内容を、確認してみましょう。
- 今回は、vi を使って /etc/group ファイルを開くことにします。

◇ /etc/group ファイルを vi で開く操作

```
[root@ws101 ~]# vi /etc/group
```

◎ vi で行番号を表示するには

- /etc/goup ファイルが開いたら、行番号を表示することにしましょう。
- vi は、起動直後はコマンドモードになっているので、キーボードから「:set nu」と入力した後に Enter キーを押してください。これで、行番号が表示されます。

◇ /etc/group ファイルの表示例

```
51 postdrop:x:90:
52 postfix:x:89:
53 mailman:x:41:
54 mysql:x:27:
55 radiusd:x:95:
56 sugimatsu:x:500:
```

◎ vi を終了させるには

- /etc/goup ファイルの内容の確認が終わったら、vi を閉じてください。
- vi を終了させる場合には、キーボードから「:q」と入力した後に Enter キーを押します。

(3) /etc/group ファイルの書式

- /etc/group ファイルも、1 行に 1 エントリずつの定義が記述されています。
- 各行はコロン(:)で区切られた 4 つのフィールドによって構成されており、次のような書式で記述されています。

◇ /etc/group ファイルの書式

```
グループ名:パスワード:グループ id:ユーザリスト
```

◇ 各フィールドに格納される情報

フィールド	格納される情報
第 1 フィールド	グループの名前
第 2 フィールド	パスワード → 実際には、/etc/passwd ファイルと同様に x が格納されている
第 3 フィールド	グループ ID の数値
第 4 フィールド	グループのメンバー全員のユーザ名。それぞれのユーザ名は、コンマ(,)で区切られている。

注：今回のセミナーでは取り扱いませんが、暗号化されたパスワードは、/etc/gshadow ファイルに格納されています。

(4) 第 3 フィールドに格納される値

- 第 3 フィールドには、グループ ID (GID) を表す値が格納されます。
- また、第 1 フィールドに格納されるグループ名は、root、システムアカウントのグループ名、一般ユーザアカウントのグループ名に分類できます。
- この第 1 フィールドに格納されるグループ名の種類と、それぞれの種類のグループ名に割り当てられるグループ ID の値の関係は、次のようになっています。

◇ グループ名の種類と割り当てられるグループ ID の関係

ログイン名の種類	格納される値
root	0
システムアカウントのグループ	1 ~ 499
一般ユーザアカウントのグループ	500 ~ 60000

① 一般ユーザのグループ ID に割り当てられる値の範囲について

- 一般ユーザのグループに割り当てられるグループ ID の開始値と終了値は、/etc/login.defs ファイルの GID_MIN パラメータと GID_MAX パラメータの設定値によって決定します。
- RedHat 系の Linux では、GID_MIN パラメータには 500、GID_MAX パラメータには 60000 という値が設定されているため、一般ユーザのグループに割り当てられるグループ ID の値の範囲は、500 ~ 60000 ということになります。

② Debian の場合は 100~60000 です

- Debian の場合には、/etc/login.defs ファイルの GID_MIN パラメータには 100、GID_MAX パラメータには 60000 という値が設定されています。
- このため、一般ユーザのグループに割り当てられるグループ ID の値の範囲は、100~60000 になります。

6. 作成したユーザアカウントでログインしてみよう

- みなさんが、先ほど作成したユーザには、まだパスワードが設定されていません。
- この状態で、2 つのテストをしてみることにします。
- 1 番目は、新規ユーザのパスワードが設定されていないままの状態、ログインできるかどうかを確認するテストです。

- そして、2 番目のテストとして、`/etc/passwd` ファイルを編集して、新規ユーザに関する定義の第 2 フィールドの値を `x` からヌルへ変更した上で、もう一度ログインできるかどうかを確認します。

(1) パスワードを設定しない状態でログインする

それでは、1 番目のテストを、次の手順で行ってください。

- ① `Alt` + `F2` キーを押して、新しいコンソールを開く。
- ② ログイン名として、新しいユーザ名を入力して Enter キーを押す。
- ③ Password:と表示され、パスワードの入力待ち状態になるので、何も入力せずに Enter キーを押す。
- ④ 再度、login:と表示され、ログイン名の入力待ちになることを確認する。
- ⑤ `Alt` + `F1` キーを押して、root 権限でログインしているコンソールに戻る。

当然のことではあります。上記の操作によって、パスワードを設定しない場合、デフォルトの `/etc/passwd` ファイルの定義のままでは、ログインできないことが確認できました。

(2) `/etc/passwd` ファイルを編集した後にログインする

次に、`/etc/passwd` ファイル内の新規ユーザに関する定義の第 2 フィールドの値を、`x` からヌルへ変更した後に、ログインできるかどうかを確認するテストを、次の手順で行ってください。

- ① `/etc/passwd` ファイルのバックアップファイルを作成する。

```
[root@ws101 ~]# cp /etc/passwd /etc/passwd.org
```

- ② `vim` コマンドを実行して、`/etc/passwd` ファイルを開く。

```
[root@ws101 ~]# vim
```

- ③ `i` キーを押して、編集モードに切り替える。
- ④ 最終行に定義されている新規ユーザの第 2 フィールドの値 `x` を削除する。
- ⑤ `Esc` キーを押して、コマンドモードに切り替える。
- ⑥ `:wq` または `:x` と入力して Enter キーを押す、編集した内容を保存した上で、`vi` を終了させる。
- ⑦ `/etc/shadow` ファイルの内容も、同様に編集するかどうかの確認メッセージが表示されるので、`n` キーを入力して Enter キーを押す、編集作業を終了する。

◇ 日本語モードで表示している場合のメッセージの表示

```
このシステムではシャドウパスワードが使われています。
今すぐ /etc/shadow を編集しますか [y/n]? n
```

◇ 英語モードで表示している場合のメッセージの表示

```
You are using shadow passwords on this system.
Would you like to edit /etc/shadow now [y/n]? n
```

- ⑧ `Alt` + `F2` キーを押して、新しいコンソールを開く。
- ⑨ ログイン名として、新しいユーザ名を入力して Enter キーを押す。
- ⑩ パスワードを入力せずに、ログインできたことを確認する。
- ⑪ `logout` または `exit` と入力し、Enter キーを押して、ログアウトする。
- ⑫ `Alt` + `F1` キーを押して、root 権限でログインしているコンソールに戻る。

上記の操作によって、`/etc/passwd` ファイル内のユーザに関する定義の第 2 フィールドの値を、`x` から `ヌル` へ変更すると、そのユーザはパスワードを入力せずに、ログインできることが確認できました。

(3) `/etc/passwd` ファイルの定義を元に戻す

それでは、テスト用に変更した `/etc/passwd` ファイル内の定義を元の記述に戻して、このテストを終了します。

- ① `vipw` コマンドを実行して、`/etc/passwd` ファイルを開く。

```
[root@ws101 ~]# vipw
```

- ② `i` キーを押して、編集モードに切り替える。
- ③ 最終行に定義されている新規ユーザの第 2 フィールドの値として、`x` を挿入する。
- ④ `Esc` キーを押して、コマンドモードに切り替える。
- ⑤ `:wq` または `:x` と入力して `Enter` キーを押し、編集した内容を保存した上で、`vi` を終了させる。
- ⑥ `/etc/shadow` ファイルの内容も、同様に編集するかどうかの確認メッセージが表示されるので、`n` キーを入力して `Enter` キーを押し、編集作業を終了する。

◇ 日本語モードで表示している場合のメッセージの表示

```
このシステムではシャドウパスワードが使われています。  
今すぐ /etc/shadow を編集しますか [y/n]? n
```

◇ 英語モードで表示している場合の場合のメッセージの表示

```
You are using shadow passwords on this system.  
Would you like to edit /etc/shadow now [y/n]? n
```

(4) Debian で `vipw` コマンドを実行した場合

- Debian で `vipw` コマンドを実行した場合、編集した後に `:wq` または `:x` と入力して `Enter` キーを押し、編集した内容を保存した上で `vi` を終了させたときには、`/etc/shadow` ファイルの内容も同様に編集するかどうかの確認メッセージは表示されず、そのまま `vi` が終了します。
- トラブルが発生して、確認メッセージが表示されないというわけではないことを、覚えておいてください。

7. 新しいユーザのパスワードを設定するには

- 新規ユーザにパスワードを設定する場合には、`passwd` コマンドを使用します。
- パスワードを設定/変更する構文は、次のとおりです。

◇ パスワードを設定/変更する構文

```
# passwd ユーザ名
```

(1) 新規ユーザのパスワードを設定する

それでは、新規ユーザのパスワードを設定する操作を実行してください。

◇ 新規ユーザのパスワードを設定する操作例

```
[root@ws101 ~]# passwd sugimatsu ← パスワードを設定するステートメントの実行
```

```
Changing password for user sugimatsu.
```

```
New UNIX password:          ← 設定するパスワードの入力(入力した文字列は非表示)
```

```
Retype new UNIX password:   ← 設定するパスワードの再入力(入力した文字列は非表示)
```

```
passwd: all authentication tokens updated successfully.      ← 設定成功のメッセージ
```

```
[root@ws101 ~]#
```

(2) パスワード設定後のファイルの内容の確認

新規ユーザのパスワードを設定したことによって、`/etc/passwd`ファイルと`/etc/shadow`ファイルの内容が、どのように変更されたのかを確認しておきましょう。

① `/etc/passwd` ファイルの確認

`tail` コマンドを使って、パスワード設定後の`/etc/passwd`ファイルの内容を確認してください。

◇ `tail` コマンドによる`/etc/passwd`ファイルの表示例 (パスワード設定後)

```
[root@ws101 ~]# tail /etc/passwd
named:x:25:25:Named:/var/named:/sbin/nologin
canna:x:39:39:Canna Service User:/var/lib/canna:/sbin/nologin
wnn:x:49:49:Wnn Input Server:/var/lib/wnn:/sbin/nologin
cyrus:x:76:12:Cyrus IMAP Server:/var/lib/imap:/bin/bash
dovecot:x:97:97:dovecot:/usr/libexec/dovecot:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
mailman:x:41:41:GNU Mailing List Manager:/usr/lib/mailman:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/bash
radiusd:x:95:95:radiusd user:/:/bin/false
sugimatsu:x:500:500::/home/sugimatsu:/bin/bash
[root@ws101 ~]#
```

◇ パスワード設定前の`/etc/passwd`ファイルの新規ユーザのエントリの定義例

```
sugimatsu:x:500:500::/home/sugimatsu:/bin/bash
```

`/etc/passwd`ファイルの定義は、パスワードを設定しても変更されないことを、確認してください。

② `/etc/shadow` ファイルの確認

パイプライン機能を使って、`cat` コマンドと `grep` コマンドを組み合わせで使用し、パスワード設定後の`/etc/shadow`ファイルの内容を確認してください。

◇ `cat` コマンドと `grep` コマンドによる`/etc/shadow`ファイルの表示例 (パスワード設定後)

```
[root@ws101 ~]# cat /etc/shadow | grep sugimatsu
sugimatsu:$1$4etJPDCP$h3BFfd7uCRsoKeCHE70Ib/::13583:0:99999:7:::
[root@ws101 ~]#
```

◇ パスワード設定前の/etc/shadow ファイルの新規ユーザのエントリの定義例

```
sugimatsu:!!:13581:0:99999:7:::
```

/etc/shadow ファイル内の新規ユーザのエントリの第 2 フィールドの値が、!!から暗号化されたパスワードの値に変更されていることを、確認してください。

8. ホームディレクトリを確認します

- 新規に作成したユーザのホームディレクトリは、自動的に/home/ディレクトリ下に、新規ユーザ名と同じ名称で作成されます。
- /etc/passwd ファイルの内容を調べた際に説明したように、このような処理が自動的に行われるのは、useradd コマンドのデフォルトの設定値を定義した/etc/default/useradd ファイルの中で、「HOME=/home」と設定されているからです。
- そして、/etc/default/useradd ファイルに定義されている「SKEL=/etc/skel」という設定によって、/home/ディレクトリ下に作成された新規ユーザ名と同じ名称のディレクトリに、/etc/skel/ディレクトリ下に置かれているファイルとディレクトリがコピーされます。
- このようにして、ホームディレクトリが自動生成されるのです。
- なお、skelというディレクトリ名は、骨格、骨組、必要最小限のものという意味を持つ skeleton(スケルトン)という用語に由来する名称です。

(1) ホームディレクトリ下のファイルとディレクトリを確認する

それでは、次の操作によって、新規に作成したユーザのホームディレクトリについて、確認してみることしましょう。操作手順は、次のとおりです。

- ① **Alt** + **F2** キーを押して、新しいコンソールを開く。
- ② ログイン名として、新しいユーザ名を入力して Enter キーを押す。
- ③ Password:と表示され、パスワードの入力が促されるので、先ほど設定したパスワードを入力して Enter キーを押す。
- ④ ログインに成功したら、ホームディレクトリ下のファイルとディレクトリを確認するために、次のステートメントを実行する。

◇ ls コマンドによるホームディレクトリ下のファイルとディレクトリの表示例

```
[sugimatsu@ws101 ~]$ ls -la
合計 52
drwx----- 3 sugimatsu sugimatsu 4096  3月 10 22:59 .
drwxr-xr-x  3 root      root      4096  3月 10 04:57 ..
-rw-----  1 sugimatsu sugimatsu   15  3月 12 04:40 .bash_history
-rw-r--r--  1 sugimatsu sugimatsu   24  3月 10 04:57 .bash_logout
-rw-r--r--  1 sugimatsu sugimatsu  191  3月 10 04:57 .bash_profile
-rw-r--r--  1 sugimatsu sugimatsu  124  3月 10 04:57 .bashrc
-rw-r--r--  1 sugimatsu sugimatsu 5619  3月 10 04:57 .canna
-rw-r--r--  1 sugimatsu sugimatsu  383  3月 10 04:57 .emacs
-rw-r--r--  1 sugimatsu sugimatsu  120  3月 10 04:57 .gtkrc
```

```
drwxr-xr-x  3 sugimatsu sugimatsu 4096  3月 10 04:57 .kde
-rw-r--r--  1 sugimatsu sugimatsu  658  3月 10 04:57 .zshrc
[sugimatsu@ws101 ~]$
```

以上の操作によって、ログインしたホームディレクトリ(ログインディレクトリ)下のファイル名とディレクトリ名を確認することができました。

(2) /etc/skel/ディレクトリ下のファイルとディレクトリを確認する

- 続いて、ホームディレクトリのコピー元(雛形)である/etc/skel/ディレクトリ下のファイルとディレクトリについて、確認することにします。
- 新規ユーザのアカウントでログインしたコンソール画面で、次のステートメントを実行してください。

◇ /etc/skel/ディレクトリ下のファイルとディレクトリの表示例

```
[sugimatsu@ws101 ~]$ ls -la /etc/skel/
合計 92
drwxr-xr-x  3 root root  4096  3月  9 13:01 .
drwxr-xr-x 96 root root 12288  3月 12 06:04 ..
-rw-r--r--  1 root root   24  8月 13  2006 .bash_logout
-rw-r--r--  1 root root  191  8月 13  2006 .bash_profile
-rw-r--r--  1 root root  124  8月 13  2006 .bashrc
-rw-r--r--  1 root root 5619  2月 22  2005 .canna
-rw-r--r--  1 root root  383  8月 13  2006 .emacs
-rw-r--r--  1 root root  120  8月 13  2006 .gtkrc
drwxr-xr-x  3 root root  4096  3月  9 12:46 .kde
-rw-r--r--  1 root root   658  8月 22  2005 .zshrc
[sugimatsu@ws101 ~]$
```

(3) 新規ユーザのホームディレクトリと/etc/skel/ディレクトリの比較

- 上記の(1)と(2)の実行結果を、比較してみてください。
- 2つのディレクトリには、ホームディレクトリ下の.bash_historyファイルを除き、まったく同じファイルとディレクトリが置かれていることが確認でき、/etc/skel/ディレクトリ下のファイルとディレクトリが、ホームディレクトリにコピーされていることが分かります。
- なお、ホームディレクトリ下の.bash_historyファイルは、bashシェルの使用履歴が保存されているファイルであり、自動的に生成されるファイルであるため、/etc/skel/ディレクトリ下には置かれていないのです。

9. ユーザ ID とグループ ID の機能を知るために

- これまでに学習してきたように、ユーザ名(ログイン名)とユーザ ID との関連付けは、/etc/passwd ファイルで定義されています。
- Linux システムの内部では、ユーザの識別はユーザ ID の番号を使って、そしてグループの識別はグループ ID によって行われています。

- そして、表示等を行う場合には、/etc/passwd ファイルを参照して、ユーザ ID をユーザ名に変換し、/etc/group ファイルを参照して、グループ ID をグループ名に変換して出力しているのです。
- そこで、このような仕組みを、一瞬にしてみなさんに理解して頂けそうなテストをしてみたいと思います。
- ハンズオンセミナーだからこそ許されるテストですので、皆さんはご自宅、学校、職場などでは絶対に行わないでください。それでは、禁断の実験を開始することにしましょう。

(1) /etc/passwd ファイルから新規ユーザのエントリ行を削除する

- 最初のテストは、/etc/passwd ファイルから新規ユーザのエントリ行を削除したら、新規ユーザのディレクトリやファイルの所有権限が、どのように表示されるかを確認する実験です。
- 操作手順は、次のとおりです。

- ① 新規ユーザでログインしているコンソールで、`logout` または `exit` と入力して Enter キーを押し、ログアウトする。
- ② `Alt` + `F1` キーを押して、root 権限でログインしているコンソールに戻る。
- ③ 次のステートメントを実行して、/etc/passwd ファイルのバックアップファイルを作成する。

```
[root@ws101 ~]# cp /etc/passwd /etc/passwd.org
```

- ④ vi で/etc/passwd ファイルを開く。
- ⑤ 新規ユーザのエントリ行にカーソルを合わせて、キーボードの D キーを 2 回押して、この行を削除する。
- ⑥ `:wq` または `:x` と入力して Enter キーを押し、編集した内容を保存した上で、vi を終了させる。
- ⑦ `ls -la` コマンドを実行して、ホームディレクトリ下のファイルとディレクトリの一覧を表示する。

◇ ls コマンドの実行結果例

```
[root@ws101 ~]# ls -la /home/sugimatsu/
合計 52
drwx----- 3 500 sugimatsu 4096 3月 10 22:59 .
drwxr-xr-x 3 root root 4096 3月 10 04:57 ..
-rw----- 1 500 sugimatsu 108 3月 12 08:30 .bash_history
-rw-r--r-- 1 500 sugimatsu 24 3月 10 04:57 .bash_logout
-rw-r--r-- 1 500 sugimatsu 191 3月 10 04:57 .bash_profile
-rw-r--r-- 1 500 sugimatsu 124 3月 10 04:57 .bashrc
-rw-r--r-- 1 500 sugimatsu 5619 3月 10 04:57 .canna
-rw-r--r-- 1 500 sugimatsu 383 3月 10 04:57 .emacs
-rw-r--r-- 1 500 sugimatsu 120 3月 10 04:57 .gtkrc
drwxr-xr-x 3 500 sugimatsu 4096 3月 10 04:57 .kde
-rw-r--r-- 1 500 sugimatsu 658 3月 10 04:57 .zshrc
[root@ws101 ~]#
```

- 上記の実行結果例で、ファイルとディレクトリの所有者の部分が、ユーザ名ではなくユーザ ID の番号で表示されていることに、注目してください。
- 表示上のこの変化は、/etc/passwd ファイルの新規ユーザのエントリ行を削除したために、ユー

ザ ID をユーザ名に変換できなくなったために起こったことなのです。

(2) /etc/group ファイルから新規ユーザのグループに関するエントリ行を削除する

- 続いて行うテストは、/etc/group ファイルから新規ユーザのグループに関するエントリ行を削除したら、新規ユーザのディレクトリやファイルの所有権限が、どのように表示されるのかを確認する実験です。
- 操作手順は、次のとおりです。

① 次のステートメントを実行して、/etc/group ファイルのバックアップファイルを作成する。

```
[root@ws101 ~]# cp /etc/group /etc/group.org
```

② vi で/etc/group ファイルを開く。

③ 新規ユーザに関するグループのエントリ行にカーソルを合わせて、キーボードの D キーを 2 回押して、この行を削除する。

④ :wq または :x と入力して Enter キーを押し、編集した内容を保存した上で、vi を終了させる。

⑤ ls -la コマンドを実行して、ホームディレクトリ下のファイルとディレクトリの一覧を表示する。

◇ ls コマンドの実行結果例

```
[root@ws101 ~]# ls -la /home/sugimatsu/
合計 52
drwx----- 3 500 500 4096 3月 10 22:59 .
drwxr-xr-x 3 root root 4096 3月 10 04:57 ..
-rw----- 1 500 500 108 3月 12 08:30 .bash_history
-rw-r--r-- 1 500 500 24 3月 10 04:57 .bash_logout
-rw-r--r-- 1 500 500 191 3月 10 04:57 .bash_profile
-rw-r--r-- 1 500 500 124 3月 10 04:57 .bashrc
-rw-r--r-- 1 500 500 5619 3月 10 04:57 .canna
-rw-r--r-- 1 500 500 383 3月 10 04:57 .emacs
-rw-r--r-- 1 500 500 120 3月 10 04:57 .gtkrc
drwxr-xr-x 3 500 500 4096 3月 10 04:57 .kde
-rw-r--r-- 1 500 500 658 3月 10 04:57 .zshrc
[root@ws101 ~]#
```

- いかがですか？、ファイルとディレクトリの所有者のユーザ ID に続いて、今度は所有者が所属するグループの部分も、グループ ID の番号で表示されるようになったことが、確認できたと思います。
- この表示上のこの変化も、/etc/group ファイルの新規ユーザに関するグループのエントリ行を削除したために、グループ ID をグループ名に変換できなくなったのが原因です。
- 少々危険なテストでしたが、ファイルシステム等はユーザ ID とグループ ID で管理されており、それらの ID 番号は/etc/passwd ファイルと/etc/group ファイルによって、ユーザ名とグループ名に変換されていることが、体験的に理解して頂けたのではないかと思います。

10. ユーザを削除するには

- それでは、資料の前半部分の最後の操作として、ユーザを削除する方法と、ユーザを削除することによって、ユーザ情報を管理する3つのファイルの内容がどのように変化するかを確認したいと思います。
- /etc/passwd ファイルと/etc/group ファイルを、それぞれのバックアップファイルを使って元の内容のファイルに戻した上で、新規に作成したユーザを削除するという手順で、進めていくことにします。
- 操作手順は、次のとおりです。

- ① 次のステートメントを実行して、バックアップファイルを使って、/etc/passwd ファイルと /etc/group ファイルを復元する。

```
[root@ws101 ~]# rm /etc/passwd          ← 編集した/etc/passwd ファイルの削除
rm: remove 通常ファイル `'/etc/passwd'? y
[root@ws101 ~]# rm /etc/group           ← 編集した/etc/group ファイルの削除
rm: remove 通常ファイル `'/etc/group'? y
[root@ws101 ~]# mv /etc/passwd.org /etc/passwd ← /etc/passwd ファイルの復元
[root@ws101 ~]# mv /etc/group.org /etc/group ← /etc/group ファイルの復元
[root@ws101 ~]#
```

- ② userdel コマンドを使って、新規に作成したユーザを削除する。

◇ userdel コマンドにより新規に作成したユーザを削除する操作例

```
[root@ws101 ~]# userdel -r sugimatsu
```

注： -r は、ユーザのホームディレクトリとメールスプールも同時に削除するためのオプションです。

- ③ cat コマンドを使って、/etc/passwd ファイルを表示し、新規ユーザのエントリ行が削除されていることを確認する。

◇ cat コマンドによる/etc/passwd ファイルの表示例

```
[root@ws101 ~]# cat -n /etc/passwd
<前略>
42 dovecot:x:97:97:dovecot:/usr/libexec/dovecot:/sbin/nologin
43 postfix:x:89:89::/var/spool/postfix:/sbin/nologin
44 mailman:x:41:41:GNU Mailing List Manager:/usr/lib/mailman:/sbin/nologin
45 mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/bash
46 radiusd:x:95:95:radiusd user:/:/bin/false
[root@ws101 ~]#
```

注： -n は、行番号を表示するオプションです。

- ④ less コマンドを使って、/etc/shadow ファイルを表示し、新規ユーザのパスワードに関するエントリ行が削除されていることを確認する。

◇ less コマンドによる/etc/passwd ファイルの表示例

```
[root@ws101 ~]# less -N /etc/shadow
 42 dovecot:!!:13581:0:99999:7:::
 43 postfix:!!:13581:0:99999:7:::
 44 mailman:!!:13581:0:99999:7:::
 45 mysql:!!:13581:0:99999:7:::
 46 radiusd:!!:13581:0:99999:7:::
(END)
```

注：-N は、行番号を表示するオプションです。

- ⑤ cat コマンドを使って、/etc/group ファイルを表示し、新規ユーザのグループに関するエントリ行が削除されていることを確認する。

◇ cat コマンドによる/etc/group ファイルの表示例

```
[root@ws101 ~]# cat -n /etc/group
<前略>
 51 postdrop:x:90:
 52 postfix:x:89:
 53 mailman:x:41:
 54 mysql:x:27:
 55 radiusd:x:95:
[root@ws101 ~]#
```

ユーザ情報を管理するすべてのファイルで、新規に作成したユーザに関する情報が削除されたことが確認できます。

Ⅱ マニュアル操作による新規ユーザの作成

- 資料の後半部分では、パスワードの設定以外はコマンド操作を行わずに、ファイルの編集操作だけで、新規のユーザを作成することによって、これまでに学習してきた内容の理解を、より一層深めていきたいと思います。
- 後半部分のキーワードも、やはり **3つのファイルと1個のディレクトリ**です。
- つまり、Linux システムにおけるユーザ管理の仕組みのポイントは、これまでに学習してきました3つのファイルと1個のディレクトリに尽きると言ってよいでしょう。

1. 新規ユーザの作成に必要な操作

実際に操作を始める前に、ファイルの編集操作だけで、新規のユーザを作成するための基本的な手順を整理しておきます。

(1) バックアップファイルの作成

不測の事態が発生したときに、直ちに原状復帰が出来るようにするために、マニュアル操作で

新規ユーザを作成するために編集する/etc/passwd ファイル、/etc/shadow ファイル、そして/etc/group ファイルのバックアップファイルを作成します。

(2) パスワードファイルへの追加

新規に作成するユーザに関するユーザ情報を、/etc/passwd ファイルに追加します。

(3) shadow ファイルへの追加

- 新規に作成するユーザのパスワード情報を、/etc/shadow ファイルに追加します。
- ただし、このファイルで管理される暗号化されたパスワード自体は、マニュアルでの新規ユーザの作成処理がすべて終了した後に、passwd コマンドを実行することによって格納することになります。

(4) グループファイルへの追加

必要に応じて、新規に作成するユーザに関するグループ情報を、/etc/group ファイルに追加します。今回の事例では、/etc/group ファイルへの追加を行うことにします。

(5) 新規ユーザのホームディレクトリの作成

新規に作成するユーザのホームディレクトリ(ログインディレクトリ)を作成します。

(6) 基本的なファイルやディレクトリのホームディレクトリへのコピー

作成したホームディレクトリに、/etc/skel/ディレクトリ下のすべてのファイルとディレクトリをコピーします。

(7) ホームディレクトリ内のファイルとディレクトリ所有権限の変更

/etc/skel/ディレクトリからホームディレクトリコピーしたすべてのファイルやディレクトリの所有者とグループを、新規に作成するユーザとグループに変更します。

(5)~(7)の新規ユーザのホームディレクトリを作成する処理は、実はさまざまな操作方法があります。今回のセミナーでは、(5)と(6)の処理を1回で行ってしまうという簡略化した操作で行うことにします。

2. 新規ユーザと新規グループの作成

それでは、具体的な情報に基づいて、マニュアル操作による新規ユーザの作成を始めるところにしましょう。

(1) 作成するユーザとグループに関する情報

新規に作成する2ユーザと1グループに関する情報は、次のとおりです。

① /etc/passwd ファイル関連情報

/etc/passwd ファイルに記述する情報は、次のとおりです。

◇ /etc/passwd ファイル関連情報

フィールド	ユーザ 1	ユーザ 2
ログイン名	user1	user2
パスワード	ヌル	ヌル
ユーザ ID	701	702
グループ ID	700	700
コメント	test_user1	test_user2
ユーザホームディレクトリ	/home/user1	/home/user2
デフォルトシェルプログラム	/bin/bash	/bin/bash

② /etc/shadow ファイル関連情報

/etc/shadow ファイルに記述する情報は、次のとおりです。

◇ /etc/shadow ファイル関連情報

フィールド	ユーザ 1	ユーザ 2
ログイン名	user1	user2
パスワード	ヌル	ヌル
1970 年 1 月 1 日から最後にパスワードが変更された日までの日数	13588	13588
パスワードが変更可能となるまでの日数	0	0
パスワードを変更しなくてはならなくなる日までの日数	99999	99999
パスワード有効期限が来ることをユーザに警告する日数	7	7
パスワード有効期限が過ぎてから、アカウントが使用不能になるまでの日数	ヌル	ヌル
1970 年 1 月 1 日からアカウントが使用不能になる日までの日数	ヌル	ヌル
予約済みのフィールド	ヌル	ヌル

③ /etc/group ファイル関連情報

/etc/group ファイルに記述する情報は、次のとおりです。

◇ /etc/group ファイル関連情報

フィールド	グループ 1
グループ名	userx
パスワード	x
グループ ID	700
ユーザリスト	ヌル

(2) 編集するファイルのバックアップファイルを作成する

- 設定ファイルなどを編集する際に、まず最初に行う操作は、操作の対象となるファイルのバックアップファイルを作成することです。
- バックアップファイルさえ作成しておけば、不測の事態が発生した場合でも、直ちに原状復帰が可能だからです。
- それでは、編集操作を行う/etc/passwd ファイル、/etc/shadow ファイル、そして/etc/group ファイル

イルについて、次のようなステートメントを実行して、バックアップファイルを作成してください。

◇ 3つのファイルのバックアップファイルを作成する操作例

```
[root@ws101 ~]# cp /etc/passwd /etc/passwd.org
[root@ws101 ~]# cp /etc/shadow /etc/shadow.org
[root@ws101 ~]# cp /etc/group /etc/group.org
[root@ws101 ~]#
```

(3) /etc/passwd ファイルへ新規ユーザ情報を追加する

vi で/etc/passwd ファイルを開き、/etc/passwd ファイルの関連情報に基づき、新規の 2 ユーザを作成するためのエントリの記述を、ファイルの末尾に追加してください。

① vi で/etc/passwd ファイルを開く。

```
[root@ws101 ~]# vi /etc/passwd
```

② 新規に作成する 2 ユーザのエントリの記述を、次のようにファイルの末尾に記述する。

```
user1::701:700:test_user1:/home/user1:/bin/bash
user2::702:700:test_user2:/home/user2:/bin/bash
```

③ :wq または :x と入力して Enter キーを押し、編集した内容を保存した上で、vi を終了させる。

(4) /etc/shadow ファイルへ新規ユーザのパスワード情報を追加する

- vi で/etc/shadow ファイルを開き、/etc/shadow ファイルの関連情報に基づき、新規の 2 ユーザを作成するためのエントリの記述を、ファイルの末尾に追加してください。
- なお、/etc/shadow ファイルを編集する場合には、パーミッションに注意してください。
- 既定のパーミッションは 400 であり、ファイルの所有者である root だけに、読み込み権限のみが許可されています。
- このパーミッションのままでは、root に書き込み権限がないため、このファイルを編集することができません。
- そこで、一旦、パーミッションを 600 に変更して、root にのみ読み込み権限と書き込み権限を与えた上で編集操作を行い、新規の 2 ユーザを作成するためのエントリの記述の追加が終了したのちに、既定のパーミッション 400 に戻すことにします。

① /etc/shadow ファイルのパーミッションを 600 に変更する。

```
[root@ws101 ~]# chmod 600 /etc/shadow
```

② vi で/etc/shadow ファイルを開く。

```
[root@ws101 ~]# vi /etc/shadow
```

③ 新規に作成する 2 ユーザのエントリの記述を、次のようにファイルの末尾に記述する。

```
user1::13588:0:99999:7:::  
user2::13588:0:99999:7:::
```

④ `:wq` または `:x` と入力して Enter キーを押し、編集した内容を保存した上で、vi を終了させる。

⑤ `/etc/shadow` ファイルのパーミッションを、既定の設定である 400 に変更する。

```
[root@ws101 ~]# chmod 400 /etc/shadow
```

(5) `/etc/group` ファイルへ新規グループ情報を追加する

vi で `/etc/group` ファイルを開き、`/etc/group` ファイルの関連情報に基づき、新規の 1 グループを作成するためのエントリの記述を、ファイルの末尾に追加してください。

① vi で `/etc/group` ファイルを開く。

```
[root@ws101 ~]# vi /etc/group
```

② 新規に作成する 1 グループのエントリの記述を、次のようにファイルの末尾に記述する。

```
userx:x:700:
```

③ `:wq` または `:x` と入力して Enter キーを押し、編集した内容を保存した上で、vi を終了させる。

(6) 作成した新規ユーザでログインしてみよう(中間テスト)

お待たせしました。それでは、ここまでの操作で設定した内容で、新規のユーザによってログインできるかどうかを、テストしてみることにしましょう。

- ④ `Alt` + `F2` キーを押して、新しいコンソールを開く。
- ⑤ ログイン名として、`user1` を入力して Enter キーを押し。
- ⑥ パスワードを入力せずに、ログインできたことを確認する。

- これまでのファイル編集操作で、入力ミスなどがなければ、`user1` というログイン名を入力すると、パスワードを入力せずにログインできます。
- ただし、ログインには成功しましたが、いつものログイン後の画面とは違いますね。

◇ `user1` のログイン前後の画面

```
CentOS release 4.4 (Final)  
Kernel 2.6.9-42.EL on an i386
```

```
ws101 login: user1
```

```
No directory /home/user1!
```

```
Loggin in the home = "/".
```

```
-bash-3.00$ _
```

← `/home/user1` ディレクトリがありませんよ！

← `/` ディレクトリをホームディレクトリとしてログインしたよ

← プロンプトの表示もいつもと違いますね

- ログイン後の画面が、このように表示されるのは、まだ user1 のホームディレクトリを作成していないのが原因です。
- このため、user2 でログインしても、同じ現象が起こるはずですよ。
- それでは、一旦ログアウトした後に、今度は user2 でログインして、同じ画面が表示されることを確認してみてください。

⑦ **logout** または **exit** と入力し、Enter キーを押して、ログアウトする。

⑧ ログイン名として、user2 を入力して Enter キーを押す。

⑨ パスワードを入力せずに、ログインできたことを確認する。

◇ user2 のログイン前後の画面

```
CentOS release 4.4 (Final)
Kernel 2.6.9-42.EL on an i386

ws101 login: user2
No directory /home/user2!           ← /home/user2 ディレクトリがありませんよ！
Loggin in the home = "/".          ← しかたがないので、/ディレクトリでログインしましたよ
-bash-3.00$ _                       ← 当然、いつもと同じプロンプトは表示できませんからね
```

- 画面の表示は納得いかないかも知れませんが、ここまでの操作に誤りがなければ、このように表示されるのが正常なのです。だから、中間テストなんですよ。
- それでは、ログインできることが確認できたら、すみやかにログアウトして、root 権限でログインしているコンソールに戻り、新規のユーザを作成する操作を続けることにしましょう。

⑩ **logout** または **exit** と入力し、Enter キーを押して、ログアウトする。

⑪ **Alt** + **F1** キーを押して、root 権限でログインしているコンソールに戻る。

以上で、中間テストは終了です。

3. 新規ユーザのホームディレクトリの作成

- マニュアル操作で新規のユーザを作成する作業も、いよいよ大詰めに迎えます。
- ここからの操作が、最終段階の作成作業になりますので、ゆっくりで結構ですから、ひとつひとつの操作内容の意味を確認しながら、ホームディレクトリの作成を行ってください。

(1) 新規ユーザのホームディレクトリを作成する

- 新規に作成する 2 ユーザのホームディレクトリを、/etc/passwd ファイルの第 6 ディレクトリに指定したディレクトリとして作成します。
- 今回は、ホームディレクトリの作成操作を簡略化するために、格納されているファイルやサブディレクトリを含めて、/etc/skel/ ディレクトリ全体を、/home/user1/ ディレクトリ、そして

/home/user2/ディレクトリという名称でコピーすることによって、新規の2ユーザのホームディレクトリを作成することにします。

- 実行するステートメントは、次のとおりです。

◇ /etc/skel/ディレクトリ全体をコピーして、新規の2ユーザのホームディレクトリを作成する

```
[root@ws101 ~]# cp -r /etc/skel/ /home/user1/  
[root@ws101 ~]# cp -r /etc/skel/ /home/user2/
```

注: -r は、ディレクトリを再帰的にコピーするために指定するオプションです。

(2) ホームディレクトリ全体の所有権限を変更する

- マニュアル操作で、新規ユーザを作成するための最後の処理は、たった今作成したばかりの2ユーザのホームディレクトリの所有権限を、ホームディレクトリ下に格納されているファイルやサブディレクトリも含めて、一括して変更する作業です。
- このような処理を行う場合には、所有権限を再帰的に変更するために、chown コマンドに-R オプションを指定して実行します。
- 実行するステートメントは、次のとおりです。

◇ 新規に作成する2ユーザのホームディレクトリの所有権限を再帰的に変更する

```
[root@ws101 ~]# chown -R user1:userx /home/user1/  
[root@ws101 ~]# chown -R user2:userx /home/user2/
```

注: 所有権限を指定する場合は、所有ユーザとグループをドット(.)で連結させても構いません。

(3) 作成した新規ユーザによるログインしてみよう(最終テスト)

- 以上ですべての作業が、終了しました。残る操作は、最終テストだけです。
- 最終テストは、新たに作成した2ユーザによるログイン操作を行います。
- ログイン後も、通常の画面が表示されれば、マニュアル操作による2ユーザの作成は成功した、ということになります。
- 中間テストでも行った操作ですが、今度はこれでクリアすれば、本当に終わりですよ、というテストとして行うことにします。

- ① **Alt** + **F2** キーを押して、新しいコンソールを開く。
- ② ログイン名として、user1 を入力して Enter キーを押す。
- ③ パスワードを入力せずに、ログインできたことを確認する。

- user1 というログイン名を入力すると、パスワードを入力せずにログインできますね。
- そして、次のような画面が表示されれば、大成功ですよ。

◇ user1 のログイン前後の画面

```
CentOS release 4.4 (Final)  
Kernel 2.6.9-42.EL on an i386
```

```
ws101 login: user1
Last Login: Mon Mar 12 17:57:04 on tty2
[user1@ws101 ~]$ _
```

- 正常にログインできることが確認できたら、一旦ログアウトして、今度は user2 でログインして、同様の確認を行ってください。

④ **logout** または **exit** と入力し、Enter キーを押して、ログアウトする。

⑤ ログイン名として、user2 を入力して Enter キーを押す。

⑥ パスワードを入力せずに、ログインできたことを確認する。

◇ user2 のログイン前後の画面

```
CentOS release 4.4 (Final)
Kernel 2.6.9-42.EL on an i386

ws101 login: user2
Last Login: Mon Mar 12 17:58:12 on tty2
[user2@ws101 ~]$ _
```

- いかがでしたか。新規に作成した 2 ユーザとも正常にログインできましたか。
- 残念ながら、何らかのトラブルが発生した場合には、ひと作業ずつ逆戻りしながら、設定内容のチェックを行ってください。
- それでは、ログアウトして、root 権限でログインしているコンソールへ戻ってください。

⑦ **logout** または **exit** と入力し、Enter キーを押して、ログアウトする。

⑧ **Alt** + **F1** キーを押して、root 権限でログインしているコンソールに戻る。

以上で、最終テストは終了です。

終わりに

ユーザ管理という操作は、たとえ Linux ユーザとしては初心者であっても、ご自分の Linux マシンを使用する場合には、欠かすことのできない操作のひとつになります。

今回のハンズオンセミナーでは、Linux システムにおけるユーザ管理の仕組みを学習することを通して、Linux というすばらしい OS への理解を深めて頂きたいと考えまして、コマンド操作から少し離れて、Linux の膨大な機能のほんの一部分に、皆さんと一緒にアプローチしてみました。

今回のセミナーに参加されて、Linux に対する皆さんの興味が、ほんの少しでも大きいものになってくれたならば、今回のセミナーは大成功だったと言ってよいと思います。

またいつの日か、短い時間であったとしても、皆さんと一緒に Linux について学習できる機会に恵まれるならば、これほど幸せなことはないと思っております。

2007 年 3 月 16 日

ハンズオンセミナー・スタッフ一同

Appendix ユーザ管理コマンドの設定ファイル

今回のハンズオンセミナーの中で、主要なファイルの既定値や useradd コマンドのデフォルトのオプションが指定されている設定ファイルとして、頻繁に登場しました/etc/login.defs ファイルと、/etc/default/useradd ファイルの内容を、付録として掲載しておきます。今後の Linux の学習や運用の際に、参考にしてください。

1. /etc/login.defs - shadow パスワード機能の設定

◇ /etc/login.defs ファイル (CentOS 4.4)

```
# *REQUIRED*
#   Directory where mailboxes reside, _or_ name of file, relative to the
#   home directory.  If you _do_ define both, MAIL_DIR takes precedence.
#   QMAIL_DIR is for Qmail
#
#QMAIL_DIR  Maildir
MAIL_DIR    /var/spool/mail
#MAIL_FILE  .mail

# Password aging controls:
#
#   PASS_MAX_DAYS   Maximum number of days a password may be used.
#   PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#   PASS_MIN_LEN    Minimum acceptable password length.
#   PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS  99999
PASS_MIN_DAYS  0
PASS_MIN_LEN   5
PASS_WARN_AGE  7

#
# Min/max values for automatic uid selection in useradd
#
UID_MIN        500
UID_MAX        60000

#
# Min/max values for automatic gid selection in groupadd
#
```

```
GID_MIN          500
GID_MAX          60000
```

```
#
# If defined, this command is run when removing a user.
# It should remove any at/cron/print jobs etc. owned by
# the user to be removed (passed as the first argument).
#
#USERDEL_CMD      /usr/sbin/userdel_local

#
# If useradd should create home directories for users by default
# On RH systems, we do. This option is ORed with the -m flag on
# useradd command line.
#
CREATE_HOME yes
```

2. /etc/default/useradd - デフォルト情報

◇ /etc/default/useradd ファイル (CentOS 4.4)

```
# useradd defaults file
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
```

以 上