User-Manage

概要

Linuxにおけるユーザ管理を習得する.
rootユーザの体験や，管理者権限の付与を行い，セキュリティを通したユーザ管理の重要性を習得する.

目次

1. ユーザ
2. グループ
3. rootユーザになる
4. コマンドの記法について
5. 自分のアカウントに管理者権限を付与
6. 管理者権限によるコマンドの実行
7. コマンドの記法
8. ホームディレクトリ
9. 新しいグループを作成
10. 新しいユーザを作成
11. ユーザの違いを体感
12. グループの違いを体感

ユーザ

Linuxのユーザは，`UID`(`User ID`)という識別子が振られ，アカウント名と対応付けて区別される． `id` コマンドを用いることで，自らのUIDを確認することができる.
以下，自分のアカウントであり，本システムの管理者アカウントは， `itakehara` として作業する.
下記の例では， ユーザ `itakehara` のUIDは，1000である.
ちなみに，gid以降の表示は，グループと呼ばれるものであり，後述する.

==========

演習1

==========

idコマンドを実行せよ.
実行結果は以下のようになる.

```
[itakehara@s20g470 ~]$ id
uid=1000(itakehara) gid=1000(itakehara) groups=1000(itakehara)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

==========

また， `id` コマンドの後ろにアカウント名を指定すると，そのアカウント名の `id` 結果が表示される.

一般ユーザ

通常のユーザがログインに使用するアカウントである.
例えば, `hoge.eng.kagawa-u.ac.jp` には, 統合認証のID(`s20g470` など)でログインできるが, これらは一般ユーザとして用意されている.

システムユーザ

アカウントはアプリケーションにより, 個別に追加されることもある.
例えば, Webサービスを配信するアプリケーションである, `Apache` は, `apache` が登録されている.
つまり, `Apache` を運用しているサーバでは, 特別な変更をしない限り, `apache` ユーザが存在する. そのため, 悪意のあるものからログインを試みられることがある.

rootユーザ(スーパユーザ)

Linuxで, 最も強い権限を持つユーザは `root` であり, 特別なユーザである.
非常に強い権限を持ち, 他のユーザやシステムに関するデータの読み込み, 書き込み, 実行を行える.
システムで重要なファイルすら削除や, 編集ができる.
rootユーザは, 多くのLinuxシステムに存在する.
そのため, SSHなどの遠隔ログインを許している場合, パスワードの総当たり攻撃の標的とされる.

応用1

さて, SSH等でサーバを公開する上で, `root` には直接ログインできないようにすることが多い.
何故であるか？

グループ

ユーザは1つ以上のグループに属している.
複数のグループに所属でき, 主グループ(Primary Group)とサブグループと言われる.
グループもユーザと同様に, `GID` (`Group ID`)が振られている.
`id` コマンドを用いて確認できる.

```
(実行結果は上記と同じなので省略)
```

上記の実行結果より, ユーザ `itakehara` は, 主グループ `itakehara` に所属しており, GIDは1000である.

==========
演習2
==========
`itakehara` のグループを確認せよ.
==========

所属する部署やユーザの役割に適切なグループを割り当てることで, グループ単位での管理を行える.
例えば, `student` グループと `teacher` グループを作成する.
学生は, 他のユーザのファイルは読み込めず, 教員は全員のファイルを読み込むことができるようにする, などの運用が考えられる.
また, `hoge.eng.kagawa-u.ac.jp` では, 学生ならば, GIDが30000として, studentが割り当てられる.

ユーザと同様に，一般グループ，システムグループ，rootグループがある．
管理者権限を実行するグループもあり，ユーザを所属させることで管理者権限を発動できる．
詳しくは，後述する．

==========
余談であるが，TA竹原は過去に，自分用のサーバで，`apache` ユーザを管理者権限を実行するグループに所属させ，PHPによるWebアプリから管理者権限のコマンドを実行できるようにしたことがある．
==========

rootユーザになる

現在のユーザは一般ユーザであり，システムの運用は難しい．
そこで，一旦 rootユーザになって運用することが思いつく．
それでは，一般ユーザとrootユーザの違いを体験してみよう．

システムのログを記す， `/var/log/messages` ファイルを読み込もうとすると，権限エラーで弾かれる．

==========
演習3
==========
`cat` や `less` コマンドを用いて， `/var/log/messages` ファイルを読み込み，権限エラーを確認せよ．
==========

`su` コマンドを使うことでユーザを変更できる．
コマンドの概要や使い方は， `man su` コマンド実行したときの `NAME` と `SYNOPSIS` に示されてある．
以下は， `man su` の実行結果である．

```
NAME
       su - run a command with substitute user and group ID

SYNOPSIS
       su [options...] [-] [user [args...]]
```

つまり， `su` コマンドは引数として，userを指定することで，そのユーザに成り代わり，コマンドを実行できる．
`root` ユーザならば `su root` であるし， `itakehara` ユーザならば `su itakehara` でユーザを変更できる．

実際に実行してみると，そのユーザのPasswordが聞かれる．
ここでは，rootユーザになろうとしているので，インストール時に決めたRoot Passwordを入力する．
いつものように，入力しても，画面上には変化がないため，気をつけること．

```
[itakehara@s20g470 ~]$ su root
Password:
[root@s20g470 itakehara]#
```

rootユーザになることができた．
先程と同様に `/var/log/messages` を読み込んで見ると，今度はmessagesファイルのログを見ることができ

る.

==========
演習4
==========
rootユーザになり，messagesファイルのログを確認せよ.
==========

自分のアカウントに管理者権限を付与

Linuxは，マルチユーザシステムであり，複数人の管理者がいる場合， rootユーザを共有することは，セキュリティ上望ましくない.
では，どのすれば良いだろうか.
限られた一般ユーザに管理者権限(root権限)を与え，個人アカウントの一般ユーザが管理者権限を発動する，ことが考えられる.
これにより，以下のメリットが生まれる.

1. 誰がログインやコマンドを実行したのか，ログに残る
2. パスワードが浪費したとしても，その個人アカウントを停止するだけで済む

さて，実際に自分のアカウントに管理者権限を与えよう.
CentOSでは，管理者権限を持つアカウントを `wheel` グループで管理している． 自分のアカウントのサブグループとして， `wheel` グループを所属させることで，管理者権限を付与できる.

rootユーザで `gpasswd` コマンドを用いることで，付与する.

```
# gpasswd -a itakehara wheel
Adding user itakehara to group wheel
```

グループの追加による適応には一度ログアウトする必要がある.
`exit` コマンドで `root` ユーザからログアウトを行い，一般ユーザに戻り，一般ユーザからもログアウトする.
つまり，2回 `exit` コマンドを実行する.
その後，一般ユーザに再度ログインする.
`id` コマンドを用いて，自身のアカウントのサブグループに， `wheel` グループが追加されていることを確認する.

==========
演習5
==========
管理者アカウントのサブグループに `wheel` グループを追加せよ.
==========

管理者権限によるコマンドの実行

さて，一般ユーザに管理者権限が付与された.
それでは，もう一度， `/var/log/message` ファイルを確認していみよう． そのままでは，やはり権限エラーにより，読み込むことができない.

それでは，どのように管理者権限を発動すればよいか.
答えは，`sudo` コマンドを使うことである.
`sudo` コマンドは，他のアカウントとしてコマンドを実行する，非常に強力なコマンドである.
以下のように，`-u` オプションの後ろにユーザを指定する.

```
$ sudo -u root cat /var/log/messages
```

すると，以下のような警告が表示される.

```
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for itakehara:
```

↓ (和訳)

```
あなたはシステム管理者から通常の講習を受けたはずです。
これは通常、以下の3点に要約されます:

    #1) 他人のプライバシーを尊重すること。
    #2) タイプする前に考えること。
    #3) 大いなる力には大いなる責任が伴うこと。

[sudo] itakehara のパスワード:
```

==========
演習6
==========
sudoコマンドを使い， `/var/log/messages` ファイルを確認せよ.
==========

個人アカウントのパスワードを入力すると，後ろのコマンド(`cat /var/log/messages`)が，管理者権限として実行できる.
ここで，気づいた点であるが，この管理者権限の発動には， rootユーザのパスワードは用いていない.
rootユーザのパスワードを秘匿したままで，管理者権限を発動できるというわけだ.

また，`/var/log/secure` ファイルを見てほしい.
以下のようなログが記録されている.

```
Nov  8 02:07:43 s20g470 sudo: itakehara : TTY=pts/0 ; PWD=/home/itakehara ;
USER=root ; COMMAND=/bin/cat /var/log/messages
```

つまり， itakehara ユーザが /bin/cat /var/log/messages を実行したということが確認できる
このように， sudo コマンドの実行状況を記録できる．
grep コマンドやパイプ | を併用すると楽に見つけられる．

また，実際に root ユーザでシステムにログインし，コマンドを実行したときの /var/log/secure を確認する．
root がログインされたことはわかるが，誰が root となっているのかがわからない．

==========
余談であるが，Debianでは，sudo グループである．
TA竹原としては， sudo グループのほうがわかりやすいと感じている．
==========

応用2

なぜ，rootユーザの共有は望ましくないのだろうか？

応用3

なぜ，当初は wheel グループにユーザが追加されていないのだろうか．
セキュリティの観点から検討せよ．

応用4

sudo コマンドと su コマンドを組み合わせるとどうなるだろうか．

コマンドの記法

ここまでは，コマンドの実行は基本的に一般ユーザのみだったため， ls のように，コマンドのみを記していた．
しかし，今後は，一般ユーザで実行するコマンドとrootユーザで実行するコマンドを分ける場面がある．
これらは，一般的な記法として，$ と # で区別する． 例えば，

```
$ id
```

と書いていれば，一般ユーザで id コマンドを実効する意味であり，

```
# id
```

と書いていれば，管理者権限で実行する意味である．

つまり，

```
$ sudo -u root ls /root
```

と

```
# ls /root
```

は記法の上では同じ意味である.
ただし，`sudo` コマンドで `root` ユーザとして実行した場合と，実際に`root` ユーザは，パスやシェルなどが異なるので，注意すること.

もちろん，実際にシステムには， `$` や `#` はタイプしない.

ホームディレクトリ

アカウントにログインしたとき，最初に位置するディレクトリを ホームディレクトリ と呼ぶ.
一般的に，近年のLinuxでは， `/home/itakehara` などであるが，運用によって異なる.
例えば， `hoge.eng.kagawa-u.ac.jp` では， `/home/st2020/s20g470` がホームディレクトリである.

応用5

`root` ユーザのホームディレクトリを調査せよ

新しいグループを作成

`groupadd` コマンドを使うことで，グループを作成できる.
作成したグループは，`/etc/group` ファイルの末尾で確認できる.
フィールドの詳細は後述する.

==========
演習7
==========
`student` グループと `teacher` グループを作成せよ.
コマンドのオプションは調査すること.
一般ユーザからは作成できないため， `sudo` を使う.
==========

`id` コマンドを実行して，サブグループに追加されていることを確認せよ.

新しいユーザを作成

それでは，以下の条件の教員ユーザを新しく作成する.

- ユーザ名: 好きな教員名や名前(英数字). `teacher`は避ける．例 `prof`，`asst` 既にあるユーザ名と重複せず，ホスト名とも重複しないこと．ホスト名は， `hostname` コマンドで確認できる.
- 主グループ: `teacher`
- サブグループ: `student`, `adm` (システムグループ)

==========
演習8
==========

上記の情報を基に， `groupadd` と `useradd` コマンドを用いてユーザを作成せよ．
オプションは，コマンドを間違えて実行したときの，修正方法は自分で調査せよ．
特に記載が無いところ(UIDやGIDなど)は，考慮しなくて良い．
教員ユーザのログインパスワードを設定せよ．
`sudo` コマンドや `passwd` コマンドを用いること．

==========

次に学生ユーザを新しく作成する．

- ユーザ名: 好きな名前．`student` は避ける．例 `ikejiri` , `furuhama`
- 主グループ: `student`

==========

演習9

==========

学生ユーザを作成せよ．
ログインパスワードの設定を忘れずに．

==========

==========

余談であるが，同じくユーザを作成するコマンドに `adduser` というコマンドがある．
CentOSでは，以下のように， `useradd` コマンドへのシンボリックリンクとなっている．

```
$ which adduser | xargs ls -lh
lrwxrwxrwx. 1 root root 7 Nov  7 00:38 /usr/sbin/adduser -> useradd
```

しかし，Debianでは，以下のように別コマンドとなっている．

```
$ sudo which adduser| xargs ls -lh
-rwxr-xr-x 1 root root 34K Sep 16  2018 /usr/sbin/adduser
$ sudo which useradd| xargs ls -lh
-rwxr-xr-x 1 root root 128K Jul 27  2018 /usr/sbin/useradd
```

これらは，ディストリビューションごとの工夫であると言えるだろう．

==========

さて，現在のユーザとグループの一覧を確認しよう．
まず，現在のユーザは， `/etc/passwd` ファイルを見ることで確認できる．

```
(~省略~)
itakehara:x:1000:1000:itakehara:/home/itakehara:/bin/bash
prof:x:1001:1001::/home/prof:/bin/bash
ikejiri:x:1002:1002::/home/ikejiri:/bin/bash
```

==========
余談であるが，`:` で区切った2カラム目に `x` と書いているが，古いUNIXでは，ここに暗号化したパスワードが書いてあった．
==========

次に現在のグループの一覧は， `/etc/group` ファイルを見ることで，確認できる．

```
(~省略~)
adm:x:4:prof
(~省略~)
wheel:x:10:itakehara
(~省略~)
itakehara:x:1000:itakehara
teacher:x:1001:
student:x:1002:prof
```

上記の2つより以下がわかる．

- `itakehara` はUID1000で，主グループはGID1000の `itakehara` に所属しており，サブグループはGID10の`wheel`に所属している．
- `prof` はUID1001で，主グループはGID1001の `teacher` に所属しており，サブグループはGID4の `adm` とGID1002の `student` に所属している．
- `ikejiri` は，UID1002で，主グループは，GID1002の `student` に所属している．

また，`id` コマンドで確認できる．

応用6

`adm` グループとはなんであろうか

- 23.10. Journal の使用 Red Hat Enterprise Linux 7 | Red Hat Customer Portal
  https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/7/html/system_administrators_guide/s1-using_the_journal

ユーザの違いを体感

==========
演習10
==========
`su` コマンドを使い，`prof` ユーザになる．
`id` コマンドで，UIDや，グループの確認をせよ．

`cd` コマンドで，ホームディレクトリに移動し，ホームディレクトリの場所を調べよ．
環境変数にて確認しても良い．

`itakehara` ユーザのホームディレクトリ(`/home/itakehara`)を見ようとする．
ホームディレクトリに自分以外のアクセス権限がないため，見れないことがわかる．
==========

これは，`ls -l /home` にて，権限を確認できる．

グループの違いを体感

`ikejiri` ユーザで，`journalctl` コマンドを使おうとすると，以下のエラーで，使えない．

```
Hint: You are currently not seeing messages from other users and the system.
      Users in the 'systemd-journal' group can see all messages. Pass -q to
      turn off this notice.
No journal files were opened due to insufficient permissions.
```

`prof` ユーザで，`sudo` コマンドを使おうとすると，以下のエラーで，使えない．

```
prof is not in the sudoers file.  This incident will be reported.
```

しかし，`prof`ユーザでは，`journalctl` コマンドを使うことができる．
これは，サブグループとして `adm`に所属しているからである．

応用7

なぜ，`sudo` コマンドが使えないのであろうか． なぜ，`prof` ユーザは，`journalctl` コマンドが使えるのであろうか．

参考文献

- Linuxシステム管理標準教科書 ダウンロード LinuCレベル2対応 | Linux技術者認定試験 リナック | LPI-Japan https://linuc.org/textbooks/admin/ , 2020/11/07 .
- 岡崎正一. UNIX -基本操作から実践活用まで-, 啓学出版株式会社, 1992年12月30日第2版発行


```
Name     竹原 一駿 ( Ichitoshi TAKEHARA )
所属     香川大学大学院 工学研究科 信頼性情報システム工学専攻 最所研究室 M1
メールアドレス   itakehara@fw.ipsj.or.jp
```


```
--------------------------------------------------------------------------------
2020/12/03 初版
2021/05/25 2版
--------------------------------------------------------------------------------
```