7. 管理者の仕事

7.1 グループとユーザ

- Linux を利用するには，ユーザアカウントが必要.
  - 任意のユーザでログインすると，Linux を利用することができる.
    - ログインしたユーザが Linux というシステムの利用権限を持っているため.
- グループを使えば複数のユーザを束ねることができる.
- グループとユーザを適切に設定することで，ファイルやディレクトリ，任意のプログラムやシェルスクリプトなどを必要なユーザにのみ参照・編集する権限や実行する権限を与えることができる.
  - グループとユーザの作成・変更・削除は root ユーザで実行する必要がある.

ユーザ

- メモリやファイルなどのさまざまな資源を利用するために，ユーザという最小単位で権限を定義できる.
  - インストール時から用意されているユーザに加え，システム管理者が必要に応じてユーザを定義できる.
  - ユーザの定義は `/etc/passwd` ファイルに記述する.
- Linux では， `/etc/passwd` ファイルをエディタで直接編集する代わりにコマンドで操作することが推奨されている.
  - `useradd` : 新しいユーザの追加
  - `usermod` : ユーザの定義の変更
  - `userdel` : ユーザの削除

ユーザの作成

- 新しくユーザを作成するには， `useradd` コマンドを使う.
  - ubuntu では `useradd` でホームディレクトリが作られないので `adduser` を使う (c.f. Ubuntu でuseaddでホームディレクトリができない).
    - 対話式
- ユーザにはユーザ ID (数字) を割り振る.
- ユーザは必ずグループに所属する.
- 作成したユーザをログインユーザとして使用する場合は， `passwd` コマンドでパスワードを登録する必要がある.

ここでは，CentOS における実行例を載せる．

書式:

```
useradd [option] ユーザ名
```

option:

-c コメント
コメント（文字列）を指定．

-g グループ名
プライマリグループ名を指定．グループ名は /etc/group ファイルで定義したもの．

-G グループ名
補助グループを指定．

-d
ホームディレクトリを指定．

-s
シェルを指定．デフォルトで /bin/bash が指定されているディストリビューションが多い．
ログインしないユーザは nologin 指定するなどする．

-u ユーザID番号
ユーザID番号を指定．

実行例 (ユーザの作成):

```
[ai@localhost ~]$ sudo su
[sudo] password for ai:
[root@localhost ai]# useradd testuser
[root@localhost ai]# cat /etc/passwd | grep testuser
testuser:x:1001:1001::/home/testuser:/bin/bash
[root@localhost ai]# ls /home/
ai  testuser
```

実行例 (ユーザ ID を指定したユーザの作成):

```
[root@localhost ai]# grep 1002 /etc/passwd
[root@localhost ai]# useradd -g users -u 1002 guestuser
[root@localhost ai]# grep guestuser /etc/passwd
guestuser:x:1002:100::/home/guestuser:/bin/bash
[root@localhost ai]# ls /home
ai  guestuser  testuser
```

ユーザアカウントの変更

ユーザアカウントを変更するには， `usermod` コマンドを使う．

書式:

```
usermod [option] ユーザ名
```

option:

`-c` コメント
コメント（文字列）を変更する．

`-g` グループ名
プライマリグループ名を変更する．グループ名は `/etc/group` ファイルで定義したもの．

`-G` グループ名
補助グループを変更．

`-l` ユーザ
既存のユーザ名を変更．

`-u` ユーザID番号
ユーザID番号を変更．

実行例 (ユーザアカウントのコメントの変更):

```
[root@localhost ai]# grep guestuser /etc/passwd
guestuser:x:1002:100::/home/guestuser:/bin/bash
[root@localhost ai]# usermod -c "Osaka University" guestuser
[root@localhost ai]# grep guestuser /etc/passwd
guestuser:x:1002:100:Osaka University:/home/guestuser:/bin/bash
```

ユーザの削除

ユーザを削除するには， `userdel` コマンドを使う．

書式:

```
userdel [option] ユーザ名
```

option:

```
-r
ホームディレクトリの削除.
```

実行例 (ユーザアカウントの削除):

```
[root@localhost ai]# grep testuser /etc/passwd
testuser:x:1001:1001::/home/testuser:/bin/bash
[root@localhost ai]# userdel -r testuser
[root@localhost ai]# grep testuser /etc/passwd
[root@localhost ai]# ls /home
ai  guestuser
```

グループ

- 複数のユーザの権限をまとめて扱うために，グループを用いる.
  - ユーザは必ず1つ以上のグループに属しており，主に所属するグループをプライマリグループと呼ぶ.
  - 最初から用意されているグループに加え，システム管理者が必要に応じてグループを定義できる.
  - グループの定義は `/etc/group` ファイルに記述する.
- Linux では， `/etc/group` ファイルをエディタで直接編集する代わりにコマンドで操作することが推奨される.
  - `groupadd` : 新しいグループの追加
  - `groupmod` : グループの定義の変更
  - `groupdel` : グループの削除

グループの作成

- 新しいグループの作成には， `groupadd` コマンドを使う.
- グループには数字のグループ ID を割り振る.

書式:

```
groupadd [option] グループ名
```

option:

```
-g グループID番号
グループID番号を指定する.
```

実行例 (グループの作成):

```
[root@localhost ai]# grep 1001 /etc/group
[root@localhost ai]# groupadd -g 1001 testgroup
[root@localhost ai]# grep testgroup /etc/group
testgroup:x:1001:
```

グループの登録情報の変更

グループの定義を変更するには， `groupmod` コマンドを使う．

書式:

```
groupmod [-g gid] [-n new-group-name] 変更対象のグループ
```

option:

```
-n
既存のグループ名を変更する場合に指定．

-g
既存のグループIDを変更．
100未満のグループIDはシステムで使われているので指定できない．
```

実行例 (グループ名の変更):

```
[root@localhost ai]# grep testgroup /etc/group
testgroup:x:1001:
[root@localhost ai]# groupmod -n test testgroup
[root@localhost ai]# grep test /etc/group
test:x:1001:
```

グループの削除

グループを削除するには， `groupdel` コマンドを使う．
`groupdel` コマンドでは，登録されているグループで，ユーザが所属していないものの情報を削除する．

書式:

```
groupdel グループ名
```

実行例 (登録したグループの削除):

```
[root@localhost ai]# grep 1001 /etc/group
test:x:1001:
[root@localhost ai]# groupdel test
[root@localhost ai]# grep 1001 /etc/group
```

7.2 パスワードとパスワードファイル

- グループの定義: `/etc/group`
- ユーザの定義: `/etc/passwd`
- パスワード: `/etc/shadow` に暗号化されて記録
    - パスワードの変更は `passwd` コマンドを使って行われる.

パスワードファイル (`/etc/passwd`)

ユーザの情報は `/etc/passwd` ファイルに保存され，1行に1ユーザの情報を `:` で区切って記述する．各行は次のようになっている.

```
account:password:UID:GID:GECOS:directory:shell
```

ここで，GECOS とは，General Electric Comprehensive Operating System の略．従来はパスワードファイルに暗号化されたパスワードが記述されていたが，多くのディストリビューションはセキュリティを考慮してシャドウファイルにパスワードを記述している.

パスワードファイルの内容:

項目	内容
account	そのシステムでのユーザ名．大文字を含まないようにする.
password	以前はユーザの暗号化されたパスワード．現在は'x'.
UID	ユーザID番号.
GID	ユーザが所属するプライマリグループID番号.
GECOS	ユーザの名前またはコメントのフィールド.
directory	ユーザのホームディレクトリ.
shell	ログイン時に起動されるユーザのコマンドインタプリタ.

グループファイル (`/etc/group`)

グループの情報は `/etc/group` ファイルに保存され，1行に1グループの情報を `:` で区切って記述する．各行は次のようになっている.

```
group_name:password:GID:user_list
```

グループファイルの内容:

項目	内容
group_name	グループの名前
password	以前は暗号化されたグループのパスワード，パスワードが不要なら空欄.
GID	グループID番号.
user_list	グループに所属するユーザ名のリスト．各ユーザ名は ',' で区切られる.

パスワード

ユーザのパスワードを登録・変更するためには， `passwd` コマンドを使う.
パスワードの変更は `root` ユーザであることが求められる.

書式:

```
passwd [ユーザ名]
```

実行例 (ユーザの追加とパスワードの設定):

```
[root@localhost ai]# cat /etc/passwd | grep 1002
guestuser:x:1002:100:Osaka University:/home/guestuser:/bin/bash
[root@localhost ai]# passwd guestuser
Changing password for user guestuser.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

CentOS の場合， `passwd` コマンドでパスワードを設定するとき，例えば以下のような場合に注意がなされる.

- ユーザ名と同じパスワードの設定
- パスワードの長さが8文字以下

シャドウファイル

ユーザのパスワードは，シャドウファイル (`/etc/shadow`) に保存される．シャドウファイルに登録された1つのユーザ (1行) の内容は以下のようになる.

```
account:password:last_changed:may_be_changed:must_be_changed:warned:expires:disabled:reserved
```

シャドウファイルはエディタで直接編集すべきではない.

項目	内容
account	ユーザ名
password	暗号化されたパスワード
last_changed	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
may_be_changed	パスワードが変更可能となるまでの日数
must_be_changed	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
warned	パスワード有効期限が来る前に，ユーザが警告を受ける日数
expires	パスワード有効期限が過ぎ，アカウントが使用不能になるまでの日数
disabled	1970年1月1日からアカウントが使用不能になるまでの日数
reserved	予約フィールド

7.3 用意されているユーザとグループ

一般のユーザとグループ

- アカウントを作成すると，ユーザ名と同様の名前のグループが作られ，ユーザはそのユーザグループに所属しているとシステムに登録される．
- グループはユーザをまとめるためにある．
  - 個別のユーザを所属部署などの単位でグループ化することができる．
  - 特定のグループにのみ権限を与えるといったことも可能となる．

root ユーザ

- システム設定の変更や，プログラムのインストールや削除，ユーザの作成・削除などに制限がない特別なユーザ．
  - アクセス権に関係なくすべてのユーザディレクトリへのアクセス，コンテンツの読み書きができる．

`su` コマンド

- `su` コマンドは，すでに別のユーザでログインしているユーザが，一時的に他のユーザになるためのコマンド．
  - オプションとしてユーザを指定しない場合は，root ユーザでシェルを起動する．
- オプションを付けずに `su` コマンドを実行した場合は，カレントディレクトリを変更せずに root ユーザでログインする．
  - カレントディレクトリを root ユーザのホームディレクトリに変更してログインするためには， `su -` または `su - root` とする．
- root ユーザでログインすると，システム管理用のコマンドを実行出来る．

書式:

```
su
su - [ユーザ]
```

option:

```
su - (or su - root)
root ユーザになる

su - user
指定したユーザになることができる．
```

`sudo` コマンド

- `sudo` コマンドを使えば，スーパーユーザ (root) 権限でコマンドを実行できる．
- `-u` オプションを付けて `sudo` コマンドを実行すると，任意のユーザでコマンドを実行できる．
- オプションを付けずに `sudo` を実行すると， `root` 権限でコマンドを実行する．
- CentOS では，初期設定のままでは `sudo` コマンドを利用することができない．
  - wheel グループという `root` 権限を持つグループに登録する必要がある．
- `sudo` の設定は， `/etc/sudoers` ファイルを編集することで変更できる．
  - `visudo` コマンドを実行すると編集できる．