Permissions

process: object:
  UID - user ID  owner
  GID - group ID  group

permissions are given by: 
  rwx  rwx  rwx 
  user  group  other

Modern OS have process elements as UID & multiple GIDs. This provides flexibility/expressibility.

Alternative for permissions for an object is ACL:
Access Control Lists (ACL) -> used in firewalls.
  ordered list of:
    L-><identity> user/group
    L-><action> r/w/x
    L->allow/deny

Examples / list of rows:
  user: pjd  rw  allow
  group: faculty  rw  allow
  user: *  r  allow
    *  *  deny

Pros: flexibility
Cons: undefined length list.

Access control Matrix

<table>
<thead>
<tr>
<th></th>
<th>file 1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>user 1</td>
<td>rwx</td>
<td>r</td>
<td>rwx</td>
<td>rwx</td>
</tr>
<tr>
<td>2</td>
<td>rw</td>
<td>rw</td>
<td>r</td>
<td>rwx</td>
</tr>
<tr>
<td>3</td>
<td>r</td>
<td>*r</td>
<td>r</td>
<td>rw</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>-</td>
<td>r</td>
<td>r</td>
</tr>
</tbody>
</table>

multiple owners

file1: owner U1, group (1,2)
  user group other
  rwx  rw-  r--

file2: owner U2, group (1,2,3)
  rw-  r--  ---
Implementing U4 is easy using ACL

U1 : rwx allow
U2 : rw allow
U3 : r allow
* : * deny

} permissions for file 1

Using ACL

Root access is required for executing below commands
CHOWN : change owner
KILL : kill process
MK NOD : make node
NET RAW:
REBOOT :
load module:

AUTHENTICATION :

Implicit (privileged) access
something you.......
know : password
have : token
are } biometrics
do } e.g. voice recogn
iris recogn.

— login (text) [ssh]
— graphical login
— gdm/xdm
— windows winlogon/gina
— MAC
Implementation:

Username -> login -> database

SETUID

start environment.

- CHAP (Challenge Authenticate Protocol)
  user < nonce system hash(pw+nounce) } ? LDAP systems.

Methods of storing passwords:
- plain text file: bad!
- hashing
- hashed + nonce (random value)
- public/private key

Attacks:
- plain text file: chance of exposing whole list of passwords if file is leaked
- hashed: dictionary attacks
- CHAP: not susceptible to replay attacks