Final Exam Format
Security

file systems

block devices / I/O / DMA 4 out of 5

virtual memory

synchronization

selective assignment of privilege

- capability: give capability to user

  access to executable

  + privilege escalation

  1 super user

permissions

acls

<table>
<thead>
<tr>
<th>U1</th>
<th>U2</th>
<th>U3</th>
<th>U4</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>rwx</td>
<td>rw-</td>
<td>r--</td>
</tr>
<tr>
<td>F2</td>
<td>rwx</td>
<td>rw-</td>
<td>r--</td>
</tr>
<tr>
<td>F3</td>
<td>rwx</td>
<td>rw-</td>
<td>r--</td>
</tr>
<tr>
<td>F4</td>
<td>rwx</td>
<td>rw-</td>
<td>r--</td>
</tr>
</tbody>
</table>

Cannot be implemented

with user-group-world

permissions: more than three permission sets

Be able to translate a matrix into a acl. (Trivial):

F1 U1: rwx allow

U2: rw allow

U3: r allow

*: deny
Be able to take a matrix and be able to translate into an ACL with groups.

- Page tables and address translation
- Page fault handling

Assume 32-bit address space

4k pages

1 page directory

Key insight: Everything is a page. Smaller page = fewer entries with fewer things covered by each page.

Page table walks:

1. Load page table
2. Look up page in table
3. Load page directory
4. Load page table
5. Load page

The cache of virtual page number to physical page number.

offset A

offset B

offset C
Page fault handling

OS is fixing up the page table so the next time the virtual address is accessed it will go through.

Multiple faults are possible:
* instruction
* subroutines access ...

Things we can implement with page fault handling:
- demand loading
- paging
- easy allocation
- copy on write