1. Aspects.
   - How do aspects support least privilege?
     
     ans. aspects allow users to authenticate the level of operations they wish to be able
     to perform. By specifying a minimal level, least privilege is provided.
   
   - Do aspects have only to do with authentication or are do they also have to do
     with authorization? (explain)
     
     ans. aspects are an authentication mechanisms. But by limiting where an user
     can activate an aspect from, it provides authorization based on the source system.
   
   - Give a set of example aspects and a possible ordering. Explain why this is rea-
     sonable.
     
     ans. u.notebook < u.home < u.work < u.machineRoom. The machine room
     is the most secure, being always locked and with 24 hour operators. The work
     environment has perimeter protection and requires authentication to enter the
     premises. The home is sometimes empty but has an alarm and uses a dedicated
     line to communicate to the office. Finally, the notebook is highly portable and can
     be easily lost or stolen.

2. Consider a milSec problem with levels unclassified and secret where secret has three
   compartments \{A\} \{B\}, \{A, B\}. Give the security card implementation.

   ans. There are four security cards labeled null, a, b, and ab

   | Identifier: null                  |
   | Groups: everyone                 |
   | Privileges:                      |
   | Security Method:                 |
   | case \langle r|w, a\rangle: switch a; |
   | case \langle r|w, b\rangle: switch b; |
3. Can any lattice model be implemented with Security Cards. If so, sketch how. If not, explain why not.

   ans. Yes, any lattice model can be implemented with security cards. In the lattice model, every node is associated with a label. In KernelSec, there is one security card per node in the lattice. Every access allowed by that node in the security card is a permission in the security card. For every node outside the permission set there is a case statement which goes to the appropriate security card.

4. In the lecture notes, chinese wall was implemented by explicitly selecting a company. Another way of implementing chinese wall is to implicitly select the company by accessing files of that company. Show how to implement that with security cards.

   ans. We give the example below of access card ford from the autos

5. Group management in Kernel Sec.
• Describe all the information to create a group set where all user are in exactly one of group A, B, or C.

ans. group template abc
  – Group tags: Atag, Btag, Ctag
  – Group info: A,1,\{u, Atag\}; B,2,\{u, Btag\}; C,3,\{u, Ctag\};
  – New group/Existing group unspecified

group set description
  – global unique group set: 10
  – name: myABC
  – template: abc

• Describe all the information to create a group set with supervisors and workers, which are mutually exclusive and employees which contain both.

ans. group template sw
  – Group tags: WorkerTag, SupervisorTag
  – Group info: worker,1,\{u, WorkerTag\}; supervisor,2,\{u, SupervisorTag\};
          employee,3,\{u, WorkerTag, SupervisorTag\};
  – New group/Existing group unspecified

group set description
  – global unique group set: 32
  – name: myWorkforce
  – template: sw


requisition • project − leader ↓ x
prepare • clerk
agree • project − leader ↓ x
issue • clerk

Identifier: requisition
Groups: a-singleton
Privileges: \langle rl, \langle a, ProjectLeader \rangle, \langle a, ProjectLeaderPToA \rangle \rangle
Security Method:
case \langle w, \langle *g, Requisition \rangle \rangle
    relabelTag(groupObject(*g,u), ProjectLeaderPToA);
    switch writeRequisition;

3
Identifier: requisitionToPrepare
Groups: a-singleton
Privileges: \(\{rl, \langle a, Clerk \rangle, \langle a, ClerkRToP \rangle\}\)
Security Method:
case \(\langle rl, \langle *g, Requistion \rangle, \langle *g, Prepare \rangle \rangle\)
    relabelTag(groupObject(*g,u), ClerkRToP);
    forceRelabelTag(obj, Agree);

Identifier: prepareToAgree
Groups: a-singleton
Privileges: \(\{rl, \langle a, Clerk \rangle, \langle a, ClerkRToP \rangle\}\)
Security Method:
case \(\langle rl, \langle *g, Prepare \rangle, \langle *g, Agree \rangle \rangle\)
    relabelTag(groupObject(*g,u), ProjectLeaderPToA);
    forceRelabelTag(obj, Agree);

Identifier: agreeToIssue
Groups: a-singleton
Privileges: \(\{rl, \langle a, Clerk \rangle, \langle a, ClerkAToI \rangle\}\)
Security Method:
case \(\langle rl, \langle *g, Agree \rangle, \langle *g, Issue \rangle \rangle\)
    relabelTag(groupObject(*g,u), ClerkAToI);
    forceRelabelTag(obj, Issue);