

CS461/ECE 422 Midterm 2 Answers and Comments**Multiple choice (2 points each)**

1. Which memory protection mechanism protects the OS memory space from user programs, protects user programs from each other, and avoids fragmentation problems as the program memory use grows?
 - a) Fence register
 - b) Base and bounds registers
 - c) Segment tables
 - d) *Page tables*

2. In which case is salt most valuable as a deterrent to an attacker.
 - a) *When a bulk attack is performed against a large password file*
 - b) When an online attack is performed
 - c) When a targeted attack is performed
 - d) When a bulk attack is performed against a small password file

3. What key piece of information is targeted by the attacker to enable a stack smashing buffer overflow attack?
 - a) Frame pointer
 - b) Stack pointer
 - c) Branch target cache
 - d) *Return address*

4. Passwords are constructed from the 13 character Elbonian alphabet. All passwords are exactly 6 characters long. The attacker can make 5,000 guesses a second. Assuming the passwords are uniformly distributed and stored in the password file as a standard crypto hash, how long will the attacker need to work before he has a 50% chance of breaking any particular password?
 - a) 121 years
 - b) 37 weeks
 - c) 15 days
 - d) *483 seconds*

5. Which of the following is the best definition for trusted path?
 - a) A means of sending information confidentially using a formally evaluated code base.
 - b) *A means of contacting functionality in the trusted computing base that cannot be intercepted by non-trustworthy entities.*
 - c) A code sequence that has passed a high assurance evaluation process.
 - d) The mechanism responsible for mediating all access control requests.

6. Which design principle is the main goal of the Chinese Wall Policy?
 - a) Least Privilege
 - b) Complete Mediation
 - c) *Separation of Duty*
 - d) Psychological Acceptability

7. What is the best definition for the the Principle of Least Common Mechanism?
 - a) *Minimize subsystems shared between mutually distrusting users.*
 - b) The design of the mechanism should be as simple as possible.
 - c) Every access to every object must be checked for authority.
 - d) Base access decisions on permissions rather than exclusions.

Short answer

8. (16 points) Consider a Bell LaPadula label system with the following clearances:

Supreme > High > Middling > Low

and the following categories:

TOW (Take Over the World), Lunch, Experiments

Consider the subject Pinky with a label of *Middling:{Lunch}*

a) What is the label of a file that he could read but not write?

Low:{Lunch} or Middling:{} or Low:{}

b) What is the label of a file that he could write but not read?

Supreme:{Lunch} or any other label that can dominate Middling:{Lunch}

c) What is the label of a file that he could read and write?

Middling:{Lunch} only

d) What is the label of a file that he could neither read nor write?

Low:{TOW} or any other label that is incomparable (label doesn't dominate Middling:{Lunch} and Middling:{Lunch} does not dominate the label)

Consider the subject The Brain with a label of *Supreme:{TOW, Lunch, Experiments}*

a) What is the label of a file that he could read but not write?

*Middling:{Lunch} or any label that is not Supreme:{TOW, Lunch, Experiments}.
The Brain's label has the highest clearance level and all categories so it dominates all other labels.*

b) What is the label of a file that he could write but not read?

There is no such label. Since The Brain's label is at system high there are no other labels that dominate it but are not dominated by it.

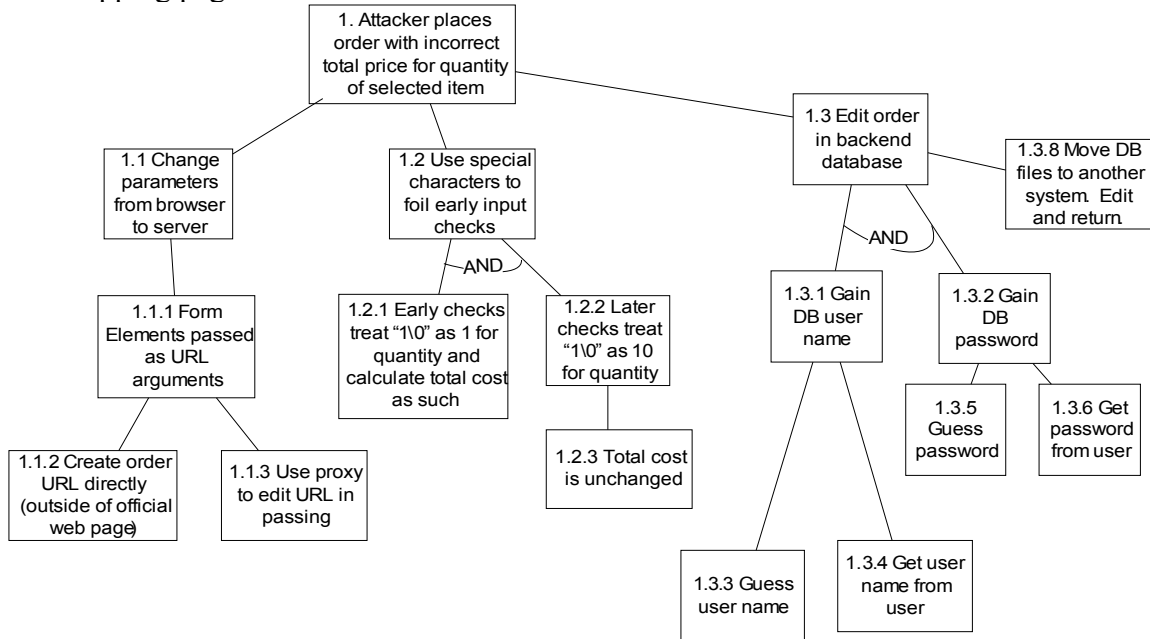
c) What is the label of a file that he could read and write?

Supreme:{TOW, Lunch, Experiments}

d) What is the label of a file that he could neither read nor write?

Again as in b there is no such label. Some of you noted that the introduction of a new category would create a non-comparable label, e.g. Middling:{new category}

9. (12 points) Consider the threat tree below. This represents a threat from a web based shopping page.



a) What are the attack paths in the tree?

1. 1 -> 1.1 -> 1.1.1 -> 1.1.2
2. 1 -> 1.1 -> 1.1.1 -> 1.1.3
3. 1 -> 1.2 -> 1.2.1 -> 1.2.2 -> 1.2.3
4. 1 -> 1.3 -> 1.3.1 -> 1.3.3 -> 1.3.2 -> 1.3.5
5. 1 -> 1.3 -> 1.3.1 -> 1.3.3 -> 1.3.2 -> 1.3.6
6. 1 -> 1.3 -> 1.3.1 -> 1.3.4 -> 1.3.2 -> 1.3.5
7. 1 -> 1.3 -> 1.3.1 -> 1.3.4 -> 1.3.2 -> 1.3.6
8. 1 -> 1.3 -> 1.3.8

Most people only identify the top three nodes in the tree, or described three or four classes of attacks through the tree.

b) Identify a set of controls that would address all attack paths?

Address 1.1.1 by changing application to not pass elements as URL elements. Or amend the set of arguments passed, so sufficient sanity checking can be performed on the server side. This deals with attack paths 1 and 2

Address 1.2 by improving input checking. Perhaps do white listing to only allow characters from a white list rather than trying to block bad characters. This deals with attack path 3.

Address 1.3 by greatly restricting direct access to the server machine. In addition you will probably want to layer controls to improve password access to the database 1.3.2

and detect out of bound database changes (1.3.8). This deals with the remaining attack paths.

c) Can this threat be completely controlled? Why or why not?

Paths 1 through 3 could be completely controlled by improving checking and parameter handling. The other paths are attacks on access that is needed by some entities, so while it can be reduced, you cannot completely control these attack paths without making normal, desirable behavior impossible.

Some of you noted as long as there are humans in the loop, we cannot guarantee that the threats are controlled (particularly in references to the paths that include the 1.3 node). Others of you noted that this threat tree was not likely exhaustive. In that case though I still wanted your opinion on whether the identified threats were adequately controlled.

10. (16 points) Consider the following access control matrix.

	System Binary	User Binary	Design Docs	Alice	Bob	Carol	Dave
Alice		X	RW				
Bob	WXA	WXA	A				
Carol	X	X	RW	D			D
Dave		X					

The rights are:

- R = read
- W = write
- X = execute
- A = edit file attributes, i.e., edit the rights for this object
- D = delegate, ability to create copy of a right held by the subject and pass to the object

a) Write the restrictions encoded in the access control matrix as access control lists.

System Binary ACL = (Bob:WXA), (Carol:X)

User Binary ACL = (Alice:X), (Bob:WXA), (Carol:X), (Dave:X)

Design Docs ACL = (Alice:RW), (Bob:A), (Carol:RW)

Alice ACL = (Carol:D)

Bob ACL =

Carol ACL =

Dave ACL = (Carol:D)

b) Write the restrictions encoded in the access control matrix as capabilities.

Alice Capabilities = (User Binary:X), (Design Docs:RW)

Bob Capabilities = (System Binary:WXA), (User Binary:WXA), (Design Docs:A)

Carol Capabilities = (System Binary:X), (User Binary:X), (Design Docs:RW), (Alice:D), (Dave:D)

Dave Capabilities = (User Binary:X)

(two more parts on the next page)

- c) Identify one benefit and one problem with each of the access control list and capability mechanisms.

ACL Benefit: If many users have the same rights, can you group users for right assignments and don't need to duplicate.

ACL Problem: Harder if not impossible to temporarily augment a user.'s process' rights to a files. If process ever needs the object right, it will be marked on that object's ACL. Could be used by another process under the same user ID in a way you did not expect.

Capability Benefit: Can very precisely pass along only the capabilities that are needed for the requested operation .

Capability Problem: Can be difficult to partially recall a capability. Performance problems. Can be difficult to assess the allowed access in general, so might be easy to get difficult to debug access errors in some cases.

There were many possible benefits and problems. Some of you picked very simple duals for the benefits of one and the problems of the other, e.g. Easy to find all rights per vs easy to find all rights per subject. These simple duals are really addressing the same issue an only got partial credit.

- d) Identify two transitions in the system using the existing rights assignments which give Dave read access to the Design Documents.

- 1. Bob has the edit attributes right for the file. He uses that right to give Dave read access to Design Documents.*
- 2. Carol can Delegate to Dave. She has read access on Design Documents, so she can delegate that right to Dave.*

11. (4 points) Identify two problems that we face with security testing above and beyond regular functionality testing.

- 1. Security errors often involve using features in unexpected ways. Not only do we need to test that features operate when used as expected, the security tester must consider how features can be misused.*
- 2. Security is often expressed as an absence, e.g., the absence of unauthorized access or the absence of improper information disclosure. Regular feature testing tests for the presence of an action. Security testing must test for the absence, which is a much bigger potential space to explore.*

Some people identified general problems with testing, e.g. it takes a lot of time, which did not get full points.

12. (8 points) Assume you were designing a program that is looking for programs infected by one set of viruses.

a) Identify two ways you would attempt to detect infected programs.

1 Pattern match for key code segments of the known virus.

2. Compare the file sizes of programs against the installed file size.

b) Consider the attacker's point of view. For each of your detection techniques, how could the attacker foil your detection attempts with his next generation virus?

- 1. If the system is pattern matching, I'd adjust my virus code to not match by changing instructions to other equivalent instructions, reorder instructions in situations that don't affect final result, break up code into blocks situated throughout the program with jumps between the blocks.*
- 2. If the system is comparing file sizes, I could implement a stealth virus which would hook the file size system calls and return the original size for the case of my infected programs.*

13. (4 points) Identify and describe two of the major relationships maintained by the certification and enforcement rules defined by the Clark-Wilson model.

Certified relationship – Identify which datasets a particular transaction procedure (program) should be able to manipulate

Allowed relationship – The triple of user, program, and data sets. Identifies which users can invoke which programs on which data sets.

14. (6 points) The Ring model implemented by the x86 processor family uses labels to implement access control between high privilege (low ring number) and low privilege (high ring number) processes and memory. Labels are also used in a number of other trusted policies and models such as Bell LaPadula, Biba, and Type Enforcement.

a) Is the Ring Model a mandatory mechanism? Why or why not?

Yes, it is mandatory. There is no way for an unprivileged user to move segments between rings or avoid the ring privilege checks.

b) Does the Ring Model implement a confidentiality policy, an integrity policy, or something else? Why?

It is neither a pure integrity or confidentiality model. The high privilege ring can read and write lower privilege data. So it could leak information to a lower privilege process by writing. And it could reduce integrity by reading lower privilege data.

15. (12 points) You are shopping for an operating system for use in a critical aspect of your organization. As such the reliability, uptime, and general trustworthiness of the system is of the utmost importance. You are in conversations with two vendors. Judging from the vendor's data sheets both products have adequate functionality for your needs, so your remaining investigations are focused on the products' assurance.

- a) Vendor A says that their software engineers are so good that very few bugs were found during the product development cycle, but all discovered bugs were fixed. Vendor B says that their quality assurance test team is so good, that 1,000's of bugs were identified and fixed during the product development cycle. Which statement impresses you the most? What additional information would help you evaluate these statements?

If Vendor A did a thorough job of testing, I'd be more impressed by them. Otherwise, Vendor B at least did a fair bit of testing.

I'd like to see the test plans from both Vendors to understand how thorough their testing strategy was.

- b) Vendor A says that they hosted an “attack the OS” contest on the Internet, and no one was able to access the sensitive information stored on the system. Vendor B says that they paid for an external penetration testing group to attack their system. They fixed all problems identified by the penetration testing group. Which statement impresses you the most? What additional information would help you evaluate these statements?

I'd be more impressed by Vendor B's penetration testing group. A professional penetration testing group is probably more thorough. Relying on ad hoc outside testing (as in the case of Vendor A's contest) might be very thorough if many people participated. But most of them probably tested the same paths.

I'd like to see statistics from Vendor A on how many attacks there were. And from both Vendors I'd like to see information on the breadth of the penetration attempts.

(one more part on the next page)

- c) Both vendor products have a Common Criteria evaluation. Vendor A's product is evaluated at EAL5 against a specialized security target. Vendor B's product is evaluated at EAL2 against a security target based on the LSPP protection profile. Which evaluation impresses you the most? What additional information would help your evaluate the evaluations?

I'd probably be more impressed by the higher assurance evaluation of Vendor A. However, I'd be concerned whether the evaluated features match the features I care about. So I would like to review the security target from Vendor A.

16. (8 points) Authentication mechanisms are frequently classified by something you know, something you are, or something you have. Give an example of an authentication mechanism that meets each of these classifications.

- a) Something you know

Password system

- b) Something you are

Fingerprint reader

- c) Something you have

Physical key, e.g., a USB or PCCard based security card.

- d) Give an example of a multi-factor authentication.

Entering a pin (something you know) into a one time password card (something you have).