Lecture

- In the lecture on May 18 we will discuss Distributed Systems. This covers three chapters of earlier or other editions from the course book. The corresponding material is not in the 9th edition of the International Edition of the course book. We uploaded the necessary material to blackboard. In the blackboard system you will find:
  - Chapter 17 from the US version of the 9th edition of the course book: Distributed Systems
  - Chapter 17 from the International version of the 8th edition of the course book: Distributed File Systems
  - Chapter 18 from the International version of the 8th edition of the course book: Distributed Synchronization
- Note, that there is an exercise class on Tuesday 10:15 as well as on Wednesday 12:15. In the exercises this week the TA will present even more details on how to solve the last mandatory assignment. Don't miss this opportunity.

Exercises

Note, as usual, that you find even more exercises including solutions here :
`http://codex.cs.yale.edu/avi/os-book/OS9/practice-exer-dir/index.html`

Prepare for the Tutorial Sessions this week:

14.2 The access-control matrix could be used to determine whether a process can switch from, say, domain A to domain B and enjoy the access privileges of domain B. Is this approach equivalent to including the access privileges of domain B in those of domain A?

14.5 Discuss the strengths and weaknesses of implementing an access matrix using access lists that are associated with objects.

14.6 Discuss the strengths and weaknesses of implementing an access matrix using capabilities that are associated with domains.

14.9 What is the need-to-know principle? Why is it important for a protection system to adhere to this principle?

14.10 Discuss which of the following systems allow module designers to enforce the need-to-know principle.

a. The MULTICS ring protection scheme

b. JVM's stack-inspection scheme

14.11 Describe how the Java protection model would be sacrificed if a Java program were allowed to directly alter the annotations of its stack frame.

14.12 How are the access-matrix facility and the role-based access-control facility similar? How do they differ?

14.13 How does the principle of least privilege aid in the creation of protection systems?

15.1 Buffer-overflow attacks can be avoided by adopting a better programming methodology or by using special hardware support. Discuss these solutions.

15.2 A password may become known to other users in a variety of ways. Is there a simple method for detecting that such an event has occurred? Explain your answer.

15.3 What is the purpose of using a "salt" along with the user-provided password? Where should the "salt" be stored, and how should it be used?

15.4 The list of all passwords is kept within the operating system. Thus, if a user manages to read this list, password protection is no longer provided. Suggest a scheme that will avoid this problem. (Hint: Use different internal and external representations.)

15.5 An experimental addition to UNIX allows a user to connect a watchdog program to a file. The watchdog is invoked whenever a program requests access to the file. The watchdog then either grants or denies access to the file. Discuss pros and cons of using watchdogs for security.

15.9 Make a list of six security concerns for a bank's computer system. For each item on your list, state whether this concern relates to physical, human, or operating-system security.

15.10 What are two advantages of encrypting data stored in the computer system?

15.11 What commonly used computer programs are prone to man-in-the-middle attacks? Discuss solutions for preventing this form of attack.

15.12 Compare symmetric and asymmetric encryption schemes, and discuss under what circumstances a distributed system would use one or the other.

15.13 Why doesn't $D_{kd,N}(E_{ke,N}(m))$ provide authentication of the sender? To what uses can such an encryption be put?

15.14 Discuss how the asymmetric encryption algorithm can be used to achieve the following goals.

a. Authentication: the receiver knows that only the sender could have generated the message.

b. Secrecy: only the receiver can decrypt the message.

c. Authentication and secrecy: only the receiver can decrypt the message, and the receiver knows that only the sender could have generated the message.

US17.1 What is the difference between computation migration and process migration? Which is easier to implement, and why?

US17.2 Contrast the various network topologies in terms of the following attributes:

a. Reliability
b. Available bandwidth for concurrent communications
c. Installation cost
d. Load balance in routing responsibilities

US17.3 Even though the ISO model of networking specifies seven layers of functionality, most computer systems use fewer layers to implement a network. Why do they use fewer layers? What problems could the use of fewer layers cause?

US17.5 What are the advantages of using dedicated hardware devices for routers and gateways? What are the disadvantages of using these de- vices compared with using general-purpose computers?

US17.12 Consider a distributed system with two sites, A and B. Consider whether site A can distinguish among the following:

a. B goes down.
b. The link between A and B goes down.
c. B is extremely overloaded and its response time is 100 times longer than normal.

17.1 What are the benefits of a DFS when compared to a file system in a centralized system?

17.3 Discuss whether AFS and NFS provide the following: (a) location transparency and (b) location independence.

17.8 AFS is designed to support a large number of clients. Discuss three techniques used to make AFS a scalable system.

17.11 Describe some of the fundamental differences between AFS and NFS.

18.3 Your company is building a computer network, and you are asked to write an algorithm for achieving distributed mutual exclusion. Which scheme will you use? Explain your choice.

18.4 Why is deadlock detection much more expensive in a distributed environment than it is in a centralized environment?

18.9 Derive an election algorithm for bidirectional rings that is more efficient than the one presented in this chapter. How many messages are needed for n processes?