CS526 Fall 2003 Qual1 supplemental, December 18, 2003
Prof. Chris Clifton, second reader Prof. Spafford

There are three questions given. You are expected to choose and answer only two of them. Please keep in mind that the qualifying exam tests your readiness to pursue a Ph.D. Think in terms of academically satisfying answers to the questions: "what are the underlying fundamentals?" rather than "how would I build this?" You might be better off choosing questions you think are harder, if it means you will better be able to display your mastery of the material.

1 Trusted Computing Base

Operating systems are growing larger and more inclusive. This has implications for security. In this question you will consider security-related tradeoffs between a monolithic all-inclusive operating system and a stripped-down kernel.

As you answer this question, try to ground your analysis in the fundamentals and concepts you have learned in the course. General intuitions, even if correct, will not be sufficient. Think of formal methods and models that you could use to design and evaluate systems, and describe how you would use them as part of your answers.

1.1 Security Issues with Monolithic Systems

Describe two or more potential security problems with a large, monolithic operating system. Give (possibly hypothetical) examples to describe the problems.

1.2 Risk Mitigation

Describe how you could mitigate the risks posed by one of the problems you describe above.

1.3 Security Issues with a Stripped-down Kernel

Describe two or more potential security problems that could occur in a system with a stripped-down kernel that would be less likely with a large, monolithic operating system. Give (possibly hypothetical) examples to describe the problems.

2 Role-Based Access Control

Role-Based Access Control (RBAC) allow subjects to assume different roles at different times, giving them different privileges. Assume an RBAC system where each subject has a set of roles they are allowed to assume, but they are only allowed to assume one role at a time. Your task is to model this formally, allowing us to reason about safety in the RBAC model.

For the rest of this question, choose either the Schematic Protection Model, Extended Schematic Protection Model, Access Control Matrix model, or Typed Access Control Matrix model.

2.1 Model

Describe how to model an RBAC system. E.g., given a set of roles, set of users (subjects), and set of privileges associated with each role, describe how to represent the RBAC system in terms of the chosen model. While you don't have to be complete, some level of formal detail is expected (e.g., describing commands and fully implementing one in ACM, or giving an example set of predicates/functions rights in SPM.)

2.2 Use of Model

Assume a user "Clifton" can be role "Useradmin" or "Printeradmin". A "Useradmin" is allowed to modify /etc/PAsswd, a "Printeradmin" is allowed to modify /etc/PRintcap. Using the model you defined, show how Clifton can modify first PAssword, then PRintcap.

2.3 Meaning of Model

What would the notion of "Safety" (defined for the HRU Access control matrix or SPM, whichever you are using) mean in the original RBAC model? In other words, in terms of the RBAC model, what does it mean to say a system is safe or unsafe, and how would you use the formal model (ACM/SPM) to evaluate this?

3 Hidden Code

Suppose you are writing a piece of code to detect and record malicious behavior (e.g., to aid in law enforcement). You obviously want this to remain undetected by the malefactor. This question deals with your ability to accomplish this.

This is not an intrusion detection question - you can assume the “suspicious behavior” has already been identified. Your goal is simply to undetectably record that behavior. You can think of this at the level of a command logger, or even keystroke logger.

3.1 Issues

Name at least one issue other than the malefactor discovering the code you have written that would enable the malefactor to discover that behavior is being logged. Discuss how you would address this issue.

3.2 Is this possible

Give a brief reason why you think it is or is not possible to write an undetectable method to record actions in each of the following situations. Best answers will be solidly grounded (e.g., you give or point to a theorem or other hard evidence that supports your statement.) However, other means are possible - for example, if the answer is yes, describing a design that would enable you to write an undetectable system.

- You are capturing the behavior on a “honeypot” – that is, you have complete control of the computer where the actions to be recorded take place

- The behavior takes place on a computer owned/managed by the malefactor, but most of the software (e.g., the operating system) is of a known design to which you have source code.

- The behavior takes place on a computer owned/managed by the malefactor, about which you have no information.

3.3 Concepts

What security concepts might help, and why? One example would be confidentiality - you'd want to develop code in a way that preserves confidentiality of the information collected. Provide one example other than confidentiality of a concept that might help, and a reason why.

3.4 Methods

What methods that you've learned might be useful, and why? A method might be something like formal verification that the code meets the requirements on preserving confidentiality. Come up with one other than what I've described.