DSV SU/KTH *IV2018(part 2)/Säk1b/IV2024*
Information and Computer Security
Autumn Term 2008

Alan Davidson

Very Brief Suggested Solutions to the Exam 2008-12-06 and Comments on the Marking

The answers suggested here may be briefer than would normally be expected from students. I have tried to summarise the most important aspects of problems so that students may compare the content their own answers to these. Lack of time prevents me from doing more.

Problem 1

> Though the access control matrix is a powerful model, it is seldom used in practice for implementing access control in computer systems. Explain what access control matrices represent and suggest reasons why they are not usually implemented. Describe a commonly implemented access control mechanism, as well as how it is related to the access control matrix.

An ACM represents the rights that subjects have over objects in a secure system. Since there can be very many separate subjects and objects in a system, and very many different types of rights, the access control matrix can potentially become a very computationally expensive data structure to use, and a very costly structure to manage.

One common mechanism is the Access Control List, which is a data structure associated with each object. It is commonly associated with file objects, in which case it normally represents what users may do with each file. If we view the ACM as a table of rights with subjects listing the rows and objects listing the columns then an individual ACL for an object represents a single column of the ACM. Rather than listing the rights for each individual user, such an ACL will very often introduce a generalisation of subjects, such as the groups of unix type system. This lessens space needed as well as lessening the difficulty of managing rights, with some loss of expressive power. The ACL can therefore be viewed as an abbreviated version of the ACM column.

Several answers gave the impression that the ACM only represents mappings between files and users. Some effort has been spent on the course (as well as in the course literature) in viewing the ACM as a general model that applies to all types of subjects and objects. Such answers were therefore marked as weak.

Problem 2

> The course book describes four general classes of threats:
>
> > ...*disclosure,* or unauthorised access to information;
> > *deception*, or acceptance of false data;
> > *disruption*, or interruption or prevention of correct operation;
> > and *usurpation*, or unauthorised control of some part of a system. [Bishop05,pp4-5]
>
> Exemplify each of of these classes of threat with a description of a tool or method that was used during the first laboratory assignment of this course. Where a class of threat was not exemplified in the practical assignment you should instead illustrate it with a description of a realistic scenario. Good answers will discuss a variety of tools and methods.

A brief list of some of the tools that could have been discussed are...

Disclosure: Key Logger, Password revealer, sniffer, nmap

Deception: Arp spoofing, password dialogue spoof

Interruption: Netbus, shut down system

Usurpation: netbus

Problem 3

> Describe each of the following IT security related terms. Also, clearly relate each of your descriptions to a closely connected IT security concept of your own choosing, and give an example of an application of these tools/threats/concepts:
>
> - One-time pad.
> - Biba
> - Biometrics
> - SQL Injection

Since this is a shortened exam solution, rather than provide descriptions of the terms here I refer to the descriptions of these terms in Wikipedia.

Some concepts that could suitably be related to these with lengthy discussions are: the Vernam Cipher, Bell LaPadula, "What the entity is", The Principle of Complete Mediation. Examples that could be elaborated upon could include: How a spy uses a physical one-time pad, how a document that needs its integrity protected is treated in a four level Biba model, fingerprints, and inserting ' OR 1=1--' into an SQL authentication script.

Several answers suggested that one-time pads are an authentication mechanism. That is a very limited view of them. They should primarily be regarded as an encryption method.

Despite rewording this exam question (relative to previous exams) to be as clear as possible, very many students did not answer this question properly. It is clearly stated that each concept should be related to another, and an example given, yet many only provide a description. I suspect this may be because students have studied earlier exam papers and solutions where only descriptions were required, and are following them rather than answering the question as it is in current exams. If good and full answers are given for several concepts, then I can allow a pass even if one of the concepts is missed or wrongly answered. Only giving descriptions implies a very high risk of failing this exam question.

Problem 4

> Imagine that you are a member of a software development project that is designing client side software that is required to communicate securely with your company's server. One of your programmers says that she is confident that by putting together a very complex mix of different substitutional and transpositional ciphers that she can create a sufficiently strong symmetric cryptographic algorithm that can secure the communications. It is proposed that each copy of the client software that is sold will be prepared with its own unique cryptographic key, hidden within the software. On the basis of this algorithm and key, the client will be automatically authorised when sending properly encrypted messages to the server, and thereby allowed to make sensitive updates to a server side database.
>
> Given this description, (quite apart from any practical problems that you may be able to guess at) identify and argue for ways in which the project may be heading for security problems where they could instead be observing sound security design principles.

The complexity suggests that economy of mechanisms is lacking. The fact that the key is hidden in software indicates that the secrecy of this design will be of great importance, i.e., the principle of open design cannot be applied. That only this mechanism is sufficient for authentication indicates that the principle of separation of privilege is not adhered to.

Other ideas that were viewed favourably:

Though it is not one of Saltzer and Schroeder's principles we are aware from the cryptography part of the course that is it not a good idea to design your own cryptographic algorithms unless you have background as a cryptographer and mathematician. There is nothing in the text to suggest that the programmer does not have this background, but I guess the odds are against it.

It seems from the text as though the programmer is on her own, which would suggest that it is difficult to apply separation of duty (one again not a Saltzer and Schroeder principle, but nothing

says that the principles discussed have to be S&S). It is probably a good idea to have another person check that the design and coding works. Note however that the question is about the design of the software, not the management of the development process, so it does not really fit the question.

Some said that the fact that the client was allowed to update important information server side indicates that the principle of least privilege is not being followed. But if the client needs to update important information, then it will need the privileges to do so. If the text had said that the client was given the privileges to update important information even though it only needed to fetch a time-stamp, that would have been indication of ignoring least privilege.

Problem 5

> If you have to select a single practical method* for assuring the quality of security of software explain which one would you select and why. Provide also a motivation in comparison with the other methods.
>
> * Note that in this exam question general assurance schemes such as The Common Criteria and SSE-CMM are assumed not to be single practical methods, and may therefore not be included as part of your answer.

This question was by Albin Zuccato as a test of the subject that he lectured on, i.e. a discussion on inspection, testing and verification. The addition was by Alan in order to clarify that this was not to be mixed up with the aspect of assurance that was covered on other parts of the course. Despite the clarification many examinees apparently were unable to associate the subject matter with Albin's assurance lecture, and nevertheless gave answers that were associated with assurance schemes rather than practical development methods. I was finally very lenient in the marking of answers that had any thread of sensible answers, but for good marks only practical methods were acceptable.

Some answers unfortunately interpreted the question from the perspective of a customer, i.e., how do I know that the software I am buying has good security. This is an interpretation of the question that was not intended, but is admittedly possible, so this kind of answer was accepted. In this interpretation answers will unfortunately be more difficult and far more vague than the intended interpretation, which meant that it was difficult to gain good marks.

There is of course no given answer to whether inspection, testing or verification should be selected, but answers should show an understanding of each of them and a reasonable motivation for choosing one of them over the others.

Reference

Bishop05 Matt Bishop, *Introduction to Computer Security*, Addison Wesley, 2005.