DSV SU/KTH

IV2018(part 2)/Säk1b/IV2024
Information and Computer Security
Autumn Term 2009

Alan Davidson

Very Brief Suggested Solutions to the Exam 2009-12-07 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

> Explain the purpose of a security policy from several different perspectives, for example, with a formal definition and a pragmatic viewpoint.

Bishops definition is:

"A *security policy* is a statement that partitions the states of the system into a set of *authorized*, or *secure*, states and a set of *unauthorized*, or *nonsecure*, states." [Bishop05, p45].

Though students may not be expected to memorize such definitions by heart for the exam, close approximations would be good as a answer to the formal part.

Some security policies will be directed towards the members of an organisation, specifying what the are allowed and not allowed to do if the organisation is to remain secure. It will therefore be written in a high level language that should be easy for people to understand and apply. There will also be measures to support this policy, such as regular education about it, and even punishment systems for not adhering to it. The purpose is to ensure that members of an organisation cooperate in upholding a secure organisation.

Some security policies will be specified in a lower level, formal language. The purpose of these is to carefully specify details of how to uphold security in order that these details could be effectively applied to a mechanism that supports the policy. One such example might be a policy that specifies what Internet traffic should be allowed in a language that is easily translated to the configuration parameters of a firewall.

Problem 2

> A student in higher education might expect to be given space on a file server where files for e.g. working with hand-in assignments can be kept. Suggest and motivate reasons why such a file-server should apply on the one hand discretionary access control, or on the other hand mandatory access control.

With the example of files for a hand-in assignment, some of these might be group assignments. In order to facilitate working on the assignment members of the same group might need to share files for that assignment. With discretionary access control the file owners (in this example the students) would themselves be able to set file privileges that would allow the other students within the group to access and even to update the files. Giving the responsibility to the student is practical, especially where students form project groups themselves. As they form groups themselves they can also alter the privileges, and there is no need for excessive administrative routines such as involve any systems administrator in order to set the privileges needed.

The above example make the assumption that one can expect students to be both honest and competent. Otherwise file privileges might be either purposefully or accidentally set to allow other students than are in that assignment group to read the group's working files. If all student groups are expected to work on the same problem in their respective groups, there may be students who are willing to break the rules and copy others' work rather than do their own. If this is a major problem then a mandatory access control system might suit better, where examiners or system administrators define rules that set the privileges of files so that only authorised group members can access them, and no others.

Many student answers to this question discussed the relative security of the two schemes without discussing the relative administrative overhead. In such a discussion, surely mandatory access control would win out every time(?).

A number of students apparently have difficulties in distinguishing between the mechanisms for changing rights (e.g. MAC & DAC) and the actual rights on objects. Marks could not be given for answers that show such confusion.

I was a little surprised to find that many students seemed to have some difficulty in understanding the concept of a file server, instead assuming all sorts of other strange functionality in their answers. This occasionally had the effect of making the arguments all the more vague.

Problem 3

> Among the many fields contained in a certificate there is an ID. Explain why this ID field is necessary, and what kinds of requirements should be put on the ID value. Give examples of suitable ID values, and of how the associated certificate would be used.

In computer security the term *certificate* is primarily understood as referring to *public key certificates*. Private keys can also be kept in a certificate structure for the sake of convenience, but in terms of public key infrastructures the most important use of certificates is for public keys.

In order to use a public key we need to have trust that the corresponding private key is owned and exclusively available to the entity with which we wish to communicate. There is no way to ascertain who the originator of a public key is from simply examining it. We need some way to ascertain who the originator is. The first step is therefore to find a way to uniquely reference the originator, so that we can never become confused about who has the corresponding private key. The name Alan is by no means unique, nor is Alan Davidson, so these would be unsuitable values. We can be fairly sure that alan@dsv.su.se will only ever refer to a single individual, so that may be a more useful identifying name. X500 Distinguished Names are an alternative scheme in order to be able to uniquely reference entities on a global scale.

Binding unique originator IDs to public keys is the method by which we can know which key to use when communicating with that originator. It is clearly important that the name be bound to the public key in a trustworthy manner. This will typically be achieved by a trusted third party digitally signing the data structure that contains the public key, the ID attribute, and other useful data, to make a certificate.

Once the trusted digital signature has been verified, we could use the certificate's public key to encrypt a symmetric key that has in turn been used to encrypt a message. We can then be safe in the knowledge that only the holder of the private key, i.e. the party that the ID attribute identifies, can use that private key to decrypt the session key, and thereby decrypt the message.

We can also use the public key from that certificate to verify digital signatures that have been made with the corresponding private key. The digital signature contains the ID of the signer that is then used as an index to find the corresponding certificate. By decrypting the signature with the certificate's public key the signature is verified.

I had the impression that several students had the impression that certificates are things that sometimes are passed to you when surfing on specific sites on the Web, and their answers to this problem were influenced by this view. A minimum requirement to show that you understand the issues of this question is to relate it to asymmetric cryptosystems.

Problem 4

Describe each of the following IT security related terms. Also, for each of these terms further illustrate the concept by choosing a closely connected IT security concept and explaining the relationship between the concepts. Furthermore, give an example of an application of these tools/threats/concepts. Give concrete examples wherever possible. Structure each of your answers with headings *description*, *relationship to [your chosen related concept]*, and *example.*

- Military security policy
- Dictionary Attack
- Challenge response
- Computer worm

Military security policy

description

A military security policy is a policy that is primarily concerned with upholding the confidentiality of the system that it is defined for. It is not a pure confidentiality policy though, since some integrity aspects will normally be included.

relationship to commercial security policy

A commercial security policy in contrast to a military is primarily concerned with upholding the integrity aspects of the system it is designed for. Whereas the military policy is mostly confidentiality and some integrity, the commercial policy will be mostly integrity with some confidentiality.

example

In terms of information, a military organisation that is intending to send a guided missile to its enemies will presumably need to spend large amount of effort in ensuring that the location of the existence of the missile base and their plans are kept secret from the enemy. The assets of the organisation are most vulnerable when the enemy can discover such information and thereafter plan defence, counter or pre-emptive measures. Confidentiality would therefore seem to be of great importance. However, certain integrity issues are also of relevance, such as the integrity of the data that guides the missile. If the enemy could manipulate that data, plans and assets of the organisation could be endangered.

Dictionary Attack

description

A dictionary attack is a method primarily used for discovering passwords whereby lists of the most likely combinations of characters in that password are first tested by trial and error. More likely combinations come about in password creation methods that are less than random, most notably due to human tendencies to choose easily remembered passwords such as those found in a dictionary, or simple transformations thereof, such as by adding numbers to the end of names. Note that the “dictionary” in a dictionary attack (i.e. the database of likely words and transformations that are tested) should not be assumed to be an actual digital dictionary of real words. The dictionary in the attack is instead compiled specifically for the purpose of testing the most likely passwords first. We would therefore expect passwords that are not included in normal dictionaries to occur in this attack dictionary (e.g. “querty” or variations on the username). Dictionaries should not be assumed to be static either, but could even be specifically constructed for specific attacks.

Dictionary attack tools are useful not only for crackers, but also for systems administrators who might need to systematically check the strength of the system's user's passwords.

relationship to brute force attack

A brute force attack also tries to discover passwords through testing systematic combinations of possible characters. As opposed to a dictionary attack, a brute force attack will not make assumptions about one guess being more likely than another. It may well instead first test the empty password, followed by all single characters, followed by all combinations of two characters, and so on. Tools for dictionary attacks will often revert to brute force attack strategies once all the likely possibilities that it knows about have been exhausted.

example

If a cracker were to manage to gain access to a system she might well attempt to retrieve a copy of that system's password hash list (the one used in the authentication process). Knowing what algorithm has been used in creating those hashes, the cracker could use a dictionary attack to guess likely passwords, hash each guess, and check whether each generated hash matches any in the password hash file. Whenever a match is found, the cracker will have discovered the password behind the original hash, and will have access to yet another account, or in a worst case scenario, gain access to administrator accounts. On systems with many users, the chances that any of the users will have chosen a relatively easily guessed password will increase.

Examples of passwords that a dictionary attack could be expected to easily discover: guest, password, secret, Stockholm, qwerty, alan69, alanalan, nala, m0th3r...

Challenge-response

description

Challenge-response is a type of protocol that can be used in an authentication process. The secure system asks a question, or requests some action of a subject seeking authentication, where it is assumed that only the subject would know the correct answer or be able to complete that action. In a strong challenge-response authentication the challenge would be one that an imposter would not be able to guess beforehand. If a challenge-response method can provide a unique challenge for each authentication attempt then the method will be immune to replay attacks.

relationship to password based authentication

Password authentication could be seen as challenge-response in its simplest form (I note that Wikipedia includes it as an example [Wikipedia09]) since the authenticating server first prompts for a password, whereupon a password is the response. Note however that the password challenge is entirely predictable, and an attacker who manages to discover the password will know that it can be used at any time for the same authentication process. It is therefore questionable whether password based authentication is a meaningful example of challenge-response.

example

When conducting transactions with my Internet bank the bank will present me with a series of digits which I am then expected to enter into a so called *security token*, i.e. a small calculator-like device that they have provided for my use. When the digits are entered into the security token it applies a mathematical function to those digits and shows a resulting different sequence of digits. The function applied is unique for the security token, so when I respond to the bank by providing the resulting digits, that is taken as proof that I am the holder of that particular security token.

Computer worm

description

According to Bishop's definition:

“A computer worm is a program that copies itself from one computer to another” [Bishop05, p373]

We may assume that this definition (in common with other definitions from the same chapter on malware) assumes an element of ill intent. A worm is commonly programmed with the ability to exploit security vulnerabilities in order to spread itself from one computer to another.

relationship to computer virus

Some definitions of the term computer virus will cover many kinds of malware, However, the more precise definitions will usually establish that a computer virus attaches itself to program hosts. In that case we can say that while viruses have programs as hosts, worms will not attach themselves to a program but to the system itself. Since worms have the ability to spread themselves from system to system, without relying on a a host program to first be activated, they will commonly spread much faster than the classic computer virus.

example

The so called Morris Worm infected the Internet while it was still young in 1988. It exploited known vulnerabilities of Unix systems, including a very rudimentary dictionary attack. Though it was not written to be harmful, accidental side effects caused it to become an effective denial of service attack for a significant portion of the Internet of the day, and an estimated 10% of Internet connected machines were infected.

Problem 5

> Suggest and describe schemes by which a software manufacturer might establish potential customers' trust in the security of their products.

The relevant scheme that we have primarily studied during the course is the Common Criteria. Please see other sources such as wikipedia or the course book for descriptions.

There are other schemes that could be discussed, such as SSE-CMM, but other schemes were not included in the course or in the reading notes, so they were not expected as answers.

There might be techniques, methods, tricks, or vague pieces of advice that can be applied to instil trust but unless they have in some way been standardised they can hardly be regarded as *schemes*. Since answers were in this sense dependent on the student's understanding of the term *scheme*, I did stretch the concept somewhat, given that not all students will have sufficient command of the English language. Nevertheless, answers would be expected to show an understanding of the Common Criteria to gain a pass.

References

Bishop05	Matt Bishop, *Introduction to Computer Security*, Addison Wesley, 2005.
Wikipedia09	Wikipedia article *Challenge-Response*, http://en.wikipedia.org/wiki/Challenge_response, visited 2010-01-07.