CS 261: Computer Security Fall 2007

Lecture 1 — August 28, 2007

Prof. David Wagner *Scribe: Igor Ganichev*

1 Overview

In this lecture we discuss the notion of computer security from ten thousand feet. We talk about what is meant by computer security and how to approach the analysis of security of a system.

Ron Rivest describes the problem of computer security as a problem of communicating in the presence of adversaries. The key point in this description is the "presence of adversaries." Indeed, the different between computer reliability and security lies precisely in the presence of adversaries. Reliable computer systems guarantee proper operation (and some other properties) under normal conditions, while secure systems give certain guarantees about system's behavior in hostile environment.

2 Security Analysis of a System

One of the hardest aspects of system security is that there are usually no empirical evaluation methods. Unless the problem is to break into a system, and the attempt is successful, it is often impossible to give 100 percent guarantees of success. Therefore, instead of empirical evaluation, we often see analytical approaches that try to assure some security property through a careful analysis of system components. We next look at the three conceptual steps that provide high-level guidelines to carry out an analytical evaluation.

2.1 Goals

The first step is to decide on the goals of our system security problem. Depending on the situation, environment, economics, etc, we might be interested in ensuring different properties. Usually, however, the goal can be stated by answering the question "what resource are we trying to protect?" or more precisely "what property of what resource are we trying to ensure?"

The "goals" aspect of analytical analysis is more a non-technical than technical. Of course, certain level of technical expertise is required to specify the goals, however, domain and situation specific knowledge provides the determining factors. Therefore, this class will not concentrate much on this step.

2.2 Threat Model

The second step of analytical evaluation is to define the threat model we are considering by specifying the capabilities and limitations of attackers. The main purpose of defining a threat model is

to scope the problem, thus making it tractable and clarifying any assumptions that might be left unstated otherwise.

Like the "goals" step, the threat model step is also primarily a question of domain expertise. It is the question of choosing the right balance in the trade-off between the system complexity and cost on the one hand and the value of a certain level of security. Unless the system is turned off and buried under a 30 feet concrete, it is probably not absolutely secure. Thus, the inevitability of a trade-off.

One way to define a threat model is to categorize all possible attacks (or attackers) according to some criteria (motivation, capabilities) that are most appropriate in the situation. A perfectly secure system would be protected against all of the categories, but such a system is usually to expensive and complex. Thus, we declare some of the attack categories out of scope for reason's like an attack type is highly unlikely or protecting the system against an attack type is too expensive. At the end, we should be left with a categorization of attacks under our consideration that we want to secure the system against.

2.3 Analysis

The last step is to answer the question "Give the goals, and the threat model, does our system achieve the goals in the proposed threat environment?" Unlike the previous two steps, this step is almost completely technical. Moreover, it is usually the hardest and there are no clear guidelines on how to approach it (that is why we are taking this class). However, we present an example security analysis below, and some ideas from that can be generalized to other contexts.

3 Example Security Analysis of a Bike

In this section, we perform example security analysis of a system which consists of a regular bike, bike lock, and a bike stand that the bike is attached to using the lock. For a detailed analysis, complete diagrams of the bike, lock, and the stand would be required, as well as information about lock's key (are there master keys, how easy it is to make a duplicate, does the owner forget keys in the lock), area where the bike is locked, etc. However, because the purpose of our presentation is a high-level illustration, rather than a complete and faithful analysis, we ignore many (important) details.

3.1 Bike Security Goals

We set regular goals for the security of out regular bike. We can roughly state two goals.

1. *Full use of the bike whenever we want.* This goal is the high-level end-user goal that can be violated through very different ways. The bike, or some part of the bike, might be stolen. Some part of the bike might be damaged for pure vandalistic purposes. Even if all bike parts are in place and in proper working condition, the bike might still be unusable for many reasons like the following. Angry girl/boyfriend might put another lock on the bike so that we cannot unlock it. Construction site might get set up around the bike so that we cannot

get to it. The bike might be taken for an hour by a friend who has keys. This example, shows an interesting point. We might write low-level goals such as "the bike is not stolen" or "all parts of the bike work" while thinking all the time that we want full availability. The dangerous part is that low-level goals might allow situations that we overlooked but actually want to secure the system against.

2. *High resale value.* This goal is pretty clear. A dirty and scribbled bike can be fully available, but it won't sell for much (and is not pretty). Thus, we want to keep the bike's resale value high.

3.2 Bike Threat Model

In the physical world, it is often easy to categorize a profile of an attacker rather than of an attack. However, in the cyber world, the attacker is not usually directly involved in the attack. Therefore, his identity is of less importance than his capabilities, tools, and expertise. In our example, we concentrate of attack, but it is not as natural because a bicycle is a real life object.

Another point to be aware of is that when specifying the threat model, one should specify the capabilities rather than specific methods that employ these capabilities. The danger in describing the threat model in terms of methods is similar to the danger of choosing low-level goals that we discussed earlier. Given a certain set of capabilities, there is usually a vast number of methods to use them. Therefore, if we specify methods, either our list will grow too long or we will miss some methods, thus excluding them from consideration and possibly leaving an open door for an attacker.

First, for illustration purposes, we explore the breadth of possible threat models by looking at the following dimensions.

- *Expertise:* none, some technical knowledge/skills, advanced technical knowledge/skills, some bike theft expertise, advanced bike theft expertise.
- *Tools:* none, some basic tools from a car trunk, basic bike theft tools, advanced bike theft tools.
- *Motivation:* economic, revenge, satisfying ego, necessity.
- *Access:* access to the place bike is locked at (during the day or at night), access to key making machine, access to bike owner's friend who has extra keys, access to bike or lock manufacturer's insider information and/or tools.

Considering some non-technical values, preferences, and trade-offs let us decide that we will only try to protect our bike from an attack carried out with basic tools, for economic purposes, and with no special expertise or access privileges.

3.3 Bike Security Analysis

One way to make the analytical evaluation more tractable is to break the evaluation process into two parts considered in the following subsections.

3.3.1 Reliance Analysis

The goal of reliance analysis is to identify the reliance set of components and what property(ies) of each component we are reliant upon. The reliance set is the set of all components on whose proper operation the security of the system is dependent. In the bike case, the obvious element of the reliance set is the lock. Less obvious ones are the wheel releases, bike stand, frame, etc. Note that we base our decision of whether to include or not a certain component into the reliance set on the goals and the threat model we specified earlier. For example, if our goals did not include full availability and the threat model considered only attackers with no tools, then (assuming the bike frame is solid) bike frame would not be part of the reliance set.

After selecting the components into the reliance set, for each one, we specify which of its properties are we dependent upon. While for some components the property might be obvious (e.g. proper functioning), it is good to be aware of the fact that one can rely on a specific property of a component without relying on another one. There is a good anecdote that explains this point. One can trust her bank with her money but not her children, and trust her relatives with her children, but not her money.

3.3.2 Reliance Evaluation

Once the reliance set has been identified, we have to evaluate if our reliance on each component is well-placed. This is the most difficult part and there are no general guidelines.

This framework provides a good starting point to deepen our understanding of the system from security's perspective. However, it should also be noted that this framework has deficiencies and should only be used as an aid rather than a panacea in general.

4 Miscellaneous Points

Finally, we make a number of random remarks about the field of computer security.

- To understand how to defend, we need to understand how to attack. Adversaries are first class citizens in any security problem. Thus, talking about them in third person, might not be adequate. Instead, putting ourselves in the place of an attacker and making full use of our imagination might reveal ways of attacking that we did not perceive before.
- We talked about different possible goals, however, generally computer systems have similar security-oriented goals.
  - **Confidentiality** Confidentiality is also called "privacy." If we are to make an analogy to file systems, confidentiality can be compared to "read" permissions. We don't want third parties to be able to read the data, or even know that there is data.
  - **Integrity** The goal of integrity is to prevent unauthorized modifications. Extending the file system analogy, we can compare integrity with "write" permissions.

  - **Availability** Availability is self-explanatory and we already had it as a goal in the bike security example. Other examples include preventing attackers from using up all CPU, or ensuring that attackers cannot prevent the system from being able contact amazon.

- Computer security literature has a somewhat misleading jargon, most widely used of which is the term "trust". When security researchers say "lock is a trusted component," the don't mean that lock is trustworthy and can be relied upon. Instead, they mean that the lock is in the reliance set. In other words, that the system's security can be compromised if the lock is compromised. Articulating the difference further, trust for security researchers is inherently a "bad" thing - it is bad if you need to trust a component and you should strive to make the number of trusted components small. On the other hand, in general usage "trust" has a positive connotation - it is good when you have many people you can trust.

- *Trusted Computing Base* is another name for reliance set, i.e. the set of all components that the system's security is dependent upon.

- *Transitive Trust* is a phenomenon when one entity relies on another entity, and the latter, in turn, relies on a third party. Thus, security of the former, is dependent upon the third party that he/she might not actually trust. Another way to articulate the issue is to note that the technical relation of trust is transitive, while the personal trust relation is often not. Transitive trust is often overlooked and can be a problem.

- *Trusted Path (problem)* occurs when two parties trust each other but don't trust the communication channel between them.

- Our final remark is a pragmatic one. Security is often about economics.