Security in Resource-Constrained Embedded Systems

Embedded systems constitute the bulk of computer systems we interact with in our daily lives. Complex systems, such as cars and industrial processes, often contain interconnected embedded systems that implement all kinds of functionality. Both previous and current research initiatives have been working to change the network environment for those systems; few of them have considered the side effects of this. Visions of an Internet-of-Things and similar systems-of-systems aim to connect previously stand-alone systems with each other and other infrastructure. Beside possibilities for information sharing and increased cooperation between systems, more interconnections also mean more threats.

From a security perspective, stand-alone systems have not been particularly well protected from adversaries. Mostly because the risks involved have not been incentive enough to enhance the security architecture. With more information — for safety, entertainment, and other services — being communicated among systems the incentives for security has increased. A major incentive is the fact that manipulation of system behaviour becomes possible through, for example, cooperative applications and remote diagnostics.

The development of larger interconnected systems poses several challenges on how to secure embedded systems. The most important category of these systems to secure is the safety-critical systems, such as vehicle computer systems where consequences of system failures can include fatalities.

First of these challenges is to know what to secure. Analysing systems for vulnerabilities and threats should always be done before deciding on the implementation of security mechanisms. However, analysing complex embedded systems is not an easy feat. Typical security analysis methods focus on either vulnerabilities or attacks, but not on the link between the two categories. This means that the analysis will either result in knowledge about vulnerabilities, but not their exploitability, or attack scenarios, but not their plausibility. To close the gap and utilise as much information as possible, a method for analysing systems with regard to both vulnerabilities and attacks is necessary. Developers and designers can get the most out of the analysis results when they are able to see exactly why an attack is realisable, compared to only knowing the concept of it. We have been working on a method to integrate these two different categories into a single method for analysing security of complex systems.

The second major challenge is to actually implement security mechanisms in a resource-constrained environment. The challenge is not as much about finding security mechanisms to thwart the threats, as it is about finding the optimal configuration and balance with regard to prevention, detection, resource consumption, fault-tolerance, etc. Implementing trust mechanisms, voting or the likes typically require probabilistic models or resource overheads that might not always be practical in embedded, safety-critical systems. Thus, there are a lot of factors to consider when developing security mechanisms for embedded network systems.

One of the more specific challenges when designing security mechanisms is how to distinguish between errors and attacks when something goes wrong and how to respond accordingly. The wrong response at the wrong time might have as severe consequences for the user of the system as an attack. Determining how to respond to attacks is the third major challenge. Appropriateness of a response is dependent on the environment the embedded system resides, which is not easily derived.