Abstract. Software distribution to target devices like factory controllers, medical instruments, vehicles or airplanes is increasingly performed electronically over insecure networks. Such software often implements vital functionality, and so the software distribution process can be highly critical, both from the safety and the security perspective. In this paper, we introduce a novel software distribution system architecture with a generic core component, such that the overall software transport from the supplier to the target device is an interaction of several instances of this core component communicating over insecure networks. The main advantage of this architecture is reduction of development and certification costs. The second contribution of this paper describes the validation and verification of the proposed system. We use a mix of formal methods, more precisely the AVISPA tool, and the Common Criteria (CC) methodology, to achieve high confidence in the security of the software distribution system at moderate costs.

1 Introduction

1.1 Network Enabled Software Distribution

In recent years, computer systems that support industrial applications, energy management and distribution, transportation systems, medical and many other applications started to use network interconnections for a range of communication needs. One such need is the distribution of software to devices in the field, in particular to allow for software updates. If such software is used to implement critical functionality that can affect the safety of people or valuable property, the software distribution process itself becomes highly critical. In other words, networked software distribution makes the safety and/or security of a system dependent upon securing communication over potentially insecure channels, facing threats like corruption, injection, diversion, replay, and disclosure of the software payload.
Various methods can be used to ensure security properties of networked systems. However, methods typically used in software development, such as testing, do not work well for security properties due to the severe consequences of subtle errors or small oversights. After all, security properties have to hold in the presence of attackers who actively try to exploit any weaknesses. A better approach to assess security of systems is to work with a well-designed catalog of requirements that is based on a broad range of experience. Certification according to Common Criteria, as discussed in the next section, falls into this category. Another proven approach is to use exhaustive search as offered by formal methods, in our case by model checking.

### 1.2 Security Certification

For assessing the security of a system, i.e., assuring that the system implements countermeasures for all relevant security threats, the Common Criteria (CC) [5] is one of the most advanced and widely accepted methodologies. The aim of an evaluation according to the CC is to systematically and objectively demonstrate that the countermeasures are sufficient and correctly implemented. The first step is to produce a specification called Security Target (ST). It defines the Target of Evaluation (TOE) which is the software, firmware and/or hardware component(s) to be evaluated, identifies threats the TOE is exposed to, derives objectives to cover the threats, states functional requirements to implement the objectives, and demands assurance requirements. The Security Target can be an instance of a generic Protection Profile (PP) which specifies the evaluation of a class of systems. We have defined such PPs for an Airplane Asset Distribution System (AADS) and its core component [7].

The CC predefined Evaluation Assurance Levels (EALs) range from 1 to 7 and determine the rigor and depth of the analysis process. Evaluation at high assurance levels, i.e., EAL5-EAL7, requires high effort for the design and implementation and also for the CC evaluation. For example, EAL6 requires a semiformal verified design based on a formal security model, and EAL7 requires full formal verification.

In [8] we have determined the assurance levels that must be met by a distribution system for airplane software. Given the high criticality of some airplane software, according to the NSA, EAL6 is recommended for safety-relevant threats, whereas EAL4 is shown sufficient for threats on airline business. In general, the distribution of software controlling safety-critical processes will require a high assurance level.

Usually CC certifications are applied to single strongly confined IT components, not to whole distributed systems consisting of several interacting entities. This is done mainly in order to limit the evaluation effort. The component-wise certification of complex systems also gives flexibility for the assembly of the overall system: components may be developed and certified individually, even by different partners.

On the other hand, we face the composition problem: the threats and vulnerabilities at system level may be different from the ones at component level. Therefore, whether the security objectives of the overall system are met as a consequence of the security properties of the individually certified components is a question to be addressed separately. The latest version 3.1 of the CC provides a first step to address this problem by providing composed assurance package (CAP) evaluations. However, CAP evaluations cannot achieve a high evaluation assurance level.