Reliability Engineering and System Safety 176 (2018) 125–134

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

SIL2 assessment of an Active/Standby COTS-based Safety-Related system

Giovanni Mazzeo[a,*], Luigi Coppolino[a], Salvatore D'Antonio[a], Claudio Mazzariello[b], Luigi Romano[a]

[a] University of Naples 'Parthenope' - Centro Direzionale, Isola C4, Napoli 80133 Italy
[b] Hitachi Ansaldo STS - Via Argine, 425, Napoli 80147, Italy

ARTICLE INFO

Keywords:
Dependability
Reliability
Safety-Related
COTS
ICS
SIL

ABSTRACT

The need of reducing costs and shortening development time is resulting in a more and more pervasive use of *Commercial-Off-The-Shelf* components also for the development of *Safety-Related* systems, which traditionally relied on ad-hoc design. This technology trend exacerbates the inherent difficulty of satisfying – and certifying – the challenging safety requirements imposed by safety certification standards, since the complexity of individual components (and consequently of the overall system) has increased by orders of magnitude. To bridge this gap, this paper proposes an approach to safety certification that is rigorous while also practical. The approach is hybrid, meaning that it effectively combines analytical modeling and field measurements. The techniques are presented and the results validated with respect to an *Active/Standby COTS*-Based industrial system, namely the Train Management System of Hitachi-Ansaldo STS, which has to satisfy *Safety Integrity Level 2* requirements. A modeling phase is first used to identify *COTS* safety bottlenecks. For these components, a mitigation strategy is proposed, and then validated in an experimental phase that is conducted on the real system. The study demonstrates that with a relatively little effort we are able to configure the target system in such a way that it achieves *SIL*2.

1. Rationale and contribution

The key building blocks of IT infrastructures responsible for the management, control, and regulation of industrial operations are generally referred to as Industrial Control Systems (ICS). Among these, a wide variety of critical systems exists, notably *Safety-Critical Systems (SCS)* and *Safety-Related Systems (SRS)*. A SCS has full responsibility for controlling hazards and consequently its failure or malfunction may result in catastrophic outcomes, such as death or serious injury to people, loss or severe damage to equipment/property, or environmental harm. SRS support SCS, since they include the hardware and software that carries out one or more safety functions. Thus, failure of an SRS increases the risk for the safety of people and/or of the environment (EN50129 says: "SRS carries responsibility for safety"). The focus of this paper is on SRS. Due to their importance, SRS must be proven to be reliable, through rigorous and internationally accepted methodologies. Standards (e.g. EN50129) exist, classifying quantitatively the likelihood of a failure through the concept of *Safety Integrity Level* (SIL*)*. Specifications include four *SIL* levels, where *SIL*0 indicates that there are no safety requirements and *SIL*4 is typically reserved to SCS. Table 1 shows the classification of the different SIL levels. For each of them, we reported the associated *Tolerable Hazard Rate* (*THR*) bounds, the failure mode, the consequent hazard, and the related system typology (i.e., SCS or SRS). Making a system compliant to a specific *SIL* means providing evidence of the achievement of *THR* thresholds, which – for a complex system – is by no means a trivial task.

The assessment process is even more difficult when Commercial-Off-The-Shelf (*COTS*) components – whose internals are partially or totally unknown – come into play. COTS are being increasingly used by the industry to reduce costs and to shorten development (and possibly deployment) time. However, since *COTS* are general-purpose components that have not been designed and developed for robust operation, obtaining a predictable operation profile (for individual components and – even more – for the resulting system) is a challenging endeavour. The research community has proposed techniques – mostly guidelines – to help reliability engineers carry out the assessment of safety properties [1–5]. However, the aforementioned studies are mainly qualitative. It is also worth emphasizing that the existing literature – with a few exceptions (e.g. [1]) – does not refer to real industry applications, which limits the applicability of the proposed techniques to commercial setups. Conversely, we contribute a methodological framework that can be used in practice as a reference for certifying a wide class of emerging critical systems, virtually any system for which: (i) the general architecture has already been designed, (ii) business constraints impose

[*] Corresponding author.
E-mail addresses: giovanni.mazzeo@uniparthenope.it (G. Mazzeo), luigi.coppolino@uniparthenope.it (L. Coppolino), salvatore.dantonio@uniparthenope.it (S. D'Antonio), claudio.mazzariello@ansaldo-sts.com (C. Mazzariello), luigi.romano@uniparthenope.it (L. Romano).

https://doi.org/10.1016/j.ress.2018.04.009
Received 8 June 2017; Received in revised form 9 April 2018; Accepted 11 April 2018
Available online 13 April 2018

Table 1
Classification of SIL levels with associated *THR and system typology.*

THR	**Failure mode**	SIL	Hazard	**System typology**
$\geq 10^{-9} to < 10^{-8}$	The system cannot be recovered by the operator	4	Catastrophic and Fatal Outcome	SCS
$\geq 10^{-8} to < 10^{-7}$	Only an experienced operator can recover the system	3	Critical and Fatal Outcome	SCS
$\geq 10^{-7} to < 10^{-6}$	An operator can recover the system	2	Marginal, Injuries may occur	SRS
$\geq 10^{-6} to < 10^{-5}$	Minor availability issues. Always recoverable	1	Negligible, Minor injuries may occur	SRS
	There is no safety requirement on the system	0	Nuisance, Dissatisfying to the user	

that (radical) changes to the architecture be avoided, and (iii) the main *COTS* components that must be integrated have already been chosen. We demonstrate, with respect to a real industrial system, that it is possible to achieve *SIL*2 via proper configuration of system parameters and set-up of rejuvenation procedures. The paper in fact addresses *SIL*2 assessment of a real *COTS*-based SRS, specifically the two-nodes cluster server hosting the Train Management System (TMS) application of Hitachi Ansaldo STS (*ASTS*). The study enabled *ASTS* to identify the conditions under which the architecture of the *Active/Standby* cluster – incorporating *COTS* software and hardware – can be certified as *SIL*2 compliant, as specified in EN50129.

In order to achieve this goal, the system is assessed by means of a *hybrid* approach, i.e., a combination of analytical modeling and experimental evaluation. Analytical modeling is done using PRISM (the well-known and widely used tool for formal model checking), while experimental evaluation relies on direct measurements on the real system. The *hybrid* approach we present consists of several phases, which can be grouped in five main iterative stages, i.e.: (1) Identifying the *THR* and safety bottlenecks of the system in its current configuration through formal models; (2) Defining possible corrective actions; (3) Weighing their effectiveness, i.e. evaluating the potential impact on the *THR* cluster model; (4) Validating the results of previous phases through an experimental campaign on the real system. (5) Using experimental estimates within models to calculate the final *THR*.

In the specific case of the *ASTS TMS* system, the safety bottlenecks turned out to be *COTS* Operating System (OS) and Cluster Resource Manager (CRM). Extended versions of the cluster model were drawn to evaluate the effectiveness of mitigation actions. The less costly one (in terms of resources and effort), which also provided results of *THR* in *SIL*2, was *software rejuvenation,cotroneo*. A *Quantitative Accelerated Life Test (QALT)* and an *Accelerated Degradation Test (ADT)* were used to estimate the experimental *Mean Time To Failure (MTTF)* and *Time To Aging-Related Failure (TTARF)* of a single server node in the short-term and in stressed execution, respectively. Such measurements were used as input to the rejuvenation-extended cluster model, and the final value of *THR* was found.

This paper makes three important contributions:

- It presents a comprehensive methodology, providing a practical path to *SIL*2 certification of *COTS*-based systems.
- It provides quantitative measurements of failure rates in a real setup, that can be reused in other industrial contexts.
- It quantifies the impact of some best practices that can be implemented in a wide class of systems, to increase their dependability.

The remainder of this work is organized as follows. Section 2 gives an insight into previous work. Section 3 describes the *ASTS* use case. Section 4 presents the approach adopted. Sections 5 and 6 describe the cluster modeling and the mitigation strategies proposed, respectively. These are followed by Section 7, that describes the experimental validation phase. Finally, Section 8 concludes the document with some final remarks.

2. Related work

Goal of this section is to show related work focused on methods useful to assess the reliability of clustered *COTS*-based systems.

Connelly et al. [1] propose an approach to delineate safety assurance for the use of *COTS* OS in safety-related applications, which must fall in SIL2. Their proposed solution is focused on developing encapsulation mechanisms able to isolate the influence of a *COTS* failure. They analyzed OS failure modes, which may affect safe functionalities, and then proposed mitigation techniques whose impact is not reported. Unlike [1], our work provides an entire methodology with tangible and quantified results that can be actually useful for other use. Qualitative estimations are definitely needed but not enough. Our contribution is a rigorous analysis of models and experimental results of the clustered system that gives deep evidence on the system reliability.

Jones et al. [5] survey available practical methods useful to assess the safe integrity of *COTS*-based systems with standard IEC61508. Authors propose the adoption of testing techniques like stress, interface and statistical test (black-box methods) or tools available to check software, analyze data flows, and inject faults (white-box methods). Limitations of both methods are presented, i.e., the lack of failure data confidential to the supplier, the difficulty when computing results of tests without automated mechanisms, the problem of optimizing time-consuming tests, the difficulty of covering a wide range of software faults. Our paper does more than that: it defines an assessment methodology flow and shows an implementation on a real use case with also on-field evaluations.

Park et al. [4] evaluate the effect of rejuvenation actions on the availability of an *Active/Standby* cluster. Authors defined a generic Markov model through which estimate how the system availability varies by changing repair time and period of rejuvenation. Main weaknesses of this paper that we contribute to address are: (i) a purely theoretical estimation without evidence or proves from real world; (ii) an estimation of the best rejuvenation period without considering that the aging of the system depends on aging factors and so needs empirical evidence; (iii) the underestimation of the possibility that the cluster resource manager fails, which is something that can certainly occur.

Skramstad et al. [2] perform a study of the possible solutions proposed by the academic and industrial community to address the certification of critical systems composed of *COTS* to a specific *SIL* level. They got to three different considerations: one could be to supervise the memory through memory mapped storage or by calculating checksums regularly; another one could be to test the system even though this is an unfeasible solution when the *COTS* is represented by a whole operating system; and finally, to diversify the adoption of *COTS* components to avoid common failure modes. This study, unlike ours, is fairly superficial, authors report few approaches available in literature without providing a comprehensive analysis.

Finally, Pierce et al. [3] present a detailed report on how to assess Linux for SRS. This document provides guidelines that should set more guarantees on the OS reliability. Many OS features are identified as possible sources of failure and for this reason they should be disabled, i.e., the developer should create a monolithic kernel with a minimum number of functionalities. The conclusion of the study is that Linux, properly tuned, would be suitable for use in many safety related applications with *SIL*1 and *SIL*2 requirements. Such a work is indeed of value for the amount of information provided, but at the same time the effective impact of mitigation techniques on the reliability is not predicted.

What emerges from literature is the lack of quantitative estimations and practical examples of certification assessment in industrial applications. Several techniques or approaches are presented but none of them