Chapter 7
Security Requirements Engineering for Evolving Software Systems: A Survey

Armstrong Nhlabatsi
The Open University, UK

Bashar Nuseibeh
Lero, Ireland & The Open University, UK

Yijun Yu
The Open University, UK

ABSTRACT
Long-lived software systems often undergo evolution over an extended period. Evolution of these systems is inevitable as they need to continue to satisfy changing business needs, new regulations and standards, and introduction of novel technologies. Such evolution may involve changes that add, remove, or modify features; or that migrate the system from one operating platform to another. These changes may result in requirements that were satisfied in a previous release of a system not being satisfied in subsequent versions. When evolutionary changes violate security requirements, a system may be left vulnerable to attacks. In this paper we review current approaches to security requirements engineering and conclude that they lack explicit support for managing the effects of software evolution. We then suggest that a cross fertilisation of the areas of software evolution and security engineering would address the problem of maintaining compliance to security requirements of software systems as they evolve.

INTRODUCTION
Software evolution refers to the process of continually updating software systems in response to changes in their operating environment and their requirements (Lehman and Ramil, 2001; Lehman and Ramil, 2003). These changes are often driven by business needs, regulations, and standards which a software application is required to continue to satisfy (Lam and Loomes, 1998; Breaux and Anton, 2008). The changes may involve adding new features, removing or modifying existing features (Keck and Kuehn, 1998; Calder et al., 2003), redesigning the system for migration to a new platform, or integration with other applications. Such changes may result...
in requirements that were satisfied in a previous
release of an application being violated in its
updated version (Ghose, 1999; Ghose, 2000).

Security requirements engineering deals with
the protection of assets from potential threats
that may lead to harm (Haley et al., 2008). This
document observes that current approaches to security
requirements engineering have limited capabil-
ity for preserving security properties that may
be violated as a result of software evolution. In
supporting this argument we review the state-of-
the-art in both literatures of software evolution
and security engineering.

In illustrating the need for security require-
ments engineering approaches to support software
evolution, we consider how the introduction of a
government regulation that only employees with
valid work permits are allowed to work may
affect a standalone payroll system. One way to
enforce this regulation could be introducing a
feature that allows a central immigration control
system to access employee database records in
the payroll system. Such a change, however, may
require migrating the payroll system to a platform
that supports public network access (such as the
Internet) where it can communicate with remote
applications. Allowing the immigration control
application access to the payroll implies that
immigration officers now have access to private
employee data which were only available with
the consent from the individual employees previ-
ously. Such evolution of the payroll system has
violated confidentiality (a subclass of security)
requirements of employees.

We suggest that one way to address the prob-
lem of violating security requirements as a result
of evolution is a cross fertilisation of approaches
to managing software evolution with security
requirements engineering. As a first step towards
achieving this cross fertilisation we propose to
use Jackson and Zave’s entailment relation (Zave
and Jackson, 1997), which relates requirements,
machine specifications and the environment, as a
tool for reasoning about both software evolution
and security requirements engineering. We envis-
age two benefits of using the entailment relation.
Firstly, it is based on a framework of requirements
ingineering that allows one to analyse software
evolution at a holistic but finer level of granularity
than other approaches in the literature (Lehman and
Ramil, 2001; Lehman and Ramil, 2003). Secondly,
by making context explicit, it allows one to elic-
t systematically security vulnerabilities associated
with context, which are very often critical (Haley
et al., 2008).

We hope that the cross fertilisation leads to an
ideal approach to security requirements engineer-
ing for evolving systems. However, we anticipate
that such cross fertilisation is non-trivial as it has
to strike a balance between security and evolution.
The theme of these challenges is how to design
software systems so that they are both secure and
evolvable. Current research in software evolution
does not explicitly address security issues and
approaches to security requirements engineering
do not provide systematic means to addressing
software evolution concerns. Meeting these chal-
enges is made harder by the fact that achieving
software systems that are both evolvable and
secure can be conflicting goals (Nhlabatsi et al.,
2008). One of the key characteristics of software
evolution is that in response to new requirements,
new features may be added to existing systems.
This mandates composition of the existing feature
set with new features. However, feature com-
position is non-monotonic (Velthuijsen, 1995);
that is, properties that were true of an existing
system before combination with a new feature,
are not guaranteed to hold after the addition of
new functionality.

This paper is structured as follows. In Section
2 we summarise the state of the art on approaches
to understanding and managing requirements evo-
lution. Section 3 reviews approaches to eliciting
and analysing security requirements and presents
a comparative evaluation of the extent to which
security requirements engineering approaches
support software evolution. The main objective
19 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the product's webpage: www.igi-global.com/chapter/security-requirements-engineering-evolving-software/65845?camid=4v1


Related Content

Hybrid Regression Testing Based on Path Pruning
www.igi-global.com/article/hybrid-regression-testing-based-on-path-pruning/125843?camid=4v1a

Teaching Agile Software Development Quality Assurance
www.igi-global.com/chapter/teaching-agile-software-development-quality/5074?camid=4v1a

The Effect of Online Service Retailers' Quality Gaps on Customer Satisfaction
Asem Majed Othman, Vincent Omachonu and Emad Hashiem Abualsauod (2017). International Journal of Systems and Service-Oriented Engineering (pp. 21-44).
www.igi-global.com/article/the-effect-of-online-service-retailers-quality-gaps-on-customer-satisfaction/188593?camid=4v1a

An Empirical Bandwidth Analysis of Interrupt-Related Covert Channels
www.igi-global.com/article/an-empirical-bandwidth-analysis-of-interrupt-related-covert-channels/136464?camid=4v1a