Available online at www.sciencedirect.com

SciVerse ScienceDirect

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

Computers & Security

Securing distributed systems using patterns: A survey

Anton V. Uzunov [a,*], *Eduardo B. Fernandez* [b], *Katrina Falkner* [a]

[a] *School of Computer Science, The University of Adelaide, Adelaide, South Australia 5005, Australia*
[b] *Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University, 777 Glades Rd., Boca Raton, FL 33431, USA*

ARTICLE INFO

Article history:
Received 23 February 2012
Received in revised form
31 March 2012
Accepted 23 April 2012

Keywords:
Computer security
Patterns
Distributed systems
Software engineering
Methodologies

ABSTRACT

Driven by expanding scientific computing and business enterprise needs, the last decade has seen a shift toward software paradigms in which distribution plays a central role. The increasing size, complexity and heterogeneity of the corresponding systems is accompanied by an increase of security vulnerabilities that require mitigation via combined security and software engineering strategies. In this respect security patterns, which build on the success of design patterns and software patterns more generally, are a tool of great value. In this paper we comprehensively survey the state-of-the-art in securing distributed systems using (security) patterns, considering both relevant patterns and methodologies for applying them. In the first part of the survey, we provide detailed reviews of our selected security patterns, classify the patterns using a multi-dimensional scheme and evaluate them according to a set of quality categories. This highlights deficiencies in the reviewed patterns and provides a basis for identifying new or "missing" patterns and pattern classes. The newly identified and surveyed patterns are a step forward in defining a pattern language for distributed computing. In the second part of the survey, we briefly review a number of pattern-based security methodologies and evaluate their maturity and appropriateness for securing distributed systems.

1. Introduction

Over the last decade the already large variety of distributed systems has grown significantly, spurred by applications such as peer-to-peer file sharing (Milojicic et al., 2002) and scientific computing (Celesti et al., 2010; Foster, 2005), as well as by enterprises moving toward software technologies in which distribution plays a central role. This trend has been accompanied by a general increase of security vulnerabilities (McGraw, 2004), both potential and real, as corresponding systems increase in size, complexity and heterogeneity, requiring a broad range of strategies for the incorporation and enforcement of security attributes.

Researchers generally agree that for any security strategy to be successful, a software system of any type must be designed for security from the earliest stages (Anderson, 2008; Devanbu and Stubblebine, 2000; Fernandez, 1999; McGraw, 2006), with a consistent maintenance of the "security-push" throughout the whole development life-cycle afterward. The tools of software engineering thus play an integral part in supporting security development activities (Mouratidis and Giorgini, 2006), becoming powerful aids in reducing the number and severity of potential vulnerabilities.

In this respect, security patterns (Fernandez, 2009; Schumacher et al., 2006), which build on the success of design patterns (Gamma et al., 1995) and software patterns

* *Corresponding author*. Tel.: +61 8 8313 7561; fax: +61 8 8313 4366.
E-mail addresses: anton.uzunov@adelaide.edu.au (A.V. Uzunov), ed@cse.fau.edu (E.B. Fernandez), katrina.falkner@adelaide.edu.au (K. Falkner).
0167-4048/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cose.2012.04.005

more generally (Buschmann et al., 1996, 2007b) can be seen as a software engineering tool of great value in the quest for building secure distributed systems. As software patterns, security patterns capture the experience and expertise of many professionals in a form accessible to the security non-expert (Fernandez and Larrondo-Petrie, 2006; Schumacher, 2003). Security techniques and countermeasures captured as patterns can be customized and applied in a number of contexts while guiding developers from a problem in a given context to a proven solution with predictable consequences. A large number of security patterns have appeared in pattern catalogs (e.g. Blakley and Heath, 2004; Dougherty et al., 2009) and books (e.g. Fernandez, in press; Microsoft, 2006; Schumacher, 2003; Schumacher et al., 2006; Steel et al., 2005) to cover a wide variety of concerns and contexts – ranging from operating systems and VOIP to cryptography and network security. Despite their usefulness and, one could argue, "proven track record" in the field, there are few comprehensive surveys of security patterns, and currently no surveys concentrating on security patterns – or the means for applying them – applicable to distributed systems in general.

Security patterns can be applied to two ways: individually or in groups to help (partially) secure a system; or as a part of an overall, coherent approach, i.e. a pattern-based security methodology. In either case, any factors of software distribution should be taken into account both by the patterns and by any methodology applying those patterns.

In this paper we comprehensively survey the state-of-the-art in securing distributed systems using (security) patterns, addressing both individual patterns and methodologies for applying them. The survey is divided into two parts, reflecting its two-fold purpose.

In the first part, we comprehensively survey individual and groups of security patterns specific for or with strong applicability to general distributed settings with the aim of identifying patterns that require development. The surveyed patterns and our suggestions and concrete ideas for new patterns or pattern languages can be seen as a first step toward a more complete security pattern language for distributed computing. As features and subsidiary contributions leading to the latter aim, we:

1. provide detailed reviews of each pattern or group of patterns according to a hierarchical classification scheme, attempting to cover as many relevant security patterns in the published literature as possible;
2. group and organize the patterns by adapting the classification schemes of Washizaki et al. (2009) and VanHilst et al. (2009), which promotes easier navigation of the existing pattern catalog;
3. evaluate the quality of the patterns according to an adaptation of the scheme proposed by Laverdiere et al. (2006), attaching quality indicators to each pattern as appropriate, which allows software professionals to quickly identify patterns with less reliability and/or missing features.

The subsidiary contributions above are made in light of the fact that, beyond their value in the software engineering process, security patterns are also used by software professionals for security educational and training purposes. Indeed, according to a recent study by Elahi et al. (2011), security patterns account for over 20% of educational resources used by professionals working in international and Chinese national IT companies, being favored over books and government standards due to their reliability and ease of use. The ultimate aim of this part of the survey (to identify patterns requiring development) can also be seen in this light as a contribution toward identifying additional domain-knowledge resources.

Limiting the scope of patterns to those relevant for distributed systems enables us to review and analyze a number of patterns not present in existing surveys (e.g. Hafiz et al., 2011; Yoshioka et al., 2008; Yskout et al., 2006 and others) or research dealing with pattern quality analysis (e.g. Heyman et al., 2007; Laverdiere et al., 2006 and others), providing a considerably more in-depth and comprehensive overview than has been available until now.

In the second part of the survey, we attempt to briefly review all existing pattern-based security methodologies with the aim of evaluating their appropriateness for securing distributed systems. Quality indicators are proposed and attached to methodologies as was done for patterns, providing a simple and effective evaluation of whether a given methodology can be used in practical situations. As a whole, the evaluations support decisions as to which methodology is best suited for a given project, and also help to highlight tendencies and possible research directions. Taken by itself, this part of our survey also helps to increase awareness of a range of pattern-based methodologies, which so far have not received due attention in the literature.

Using the individual patterns or their composites from the first part of the survey can aid in (partially) securing a distributed system, but their greatest merits become manifest when they are used as part of an overall methodology. Using a pattern-based methodology for securing a distributed system implies that the relevant patterns must necessarily originate from a collection applicable to such systems, and hence the quantity and quality of patterns effects the methodology's capacities. Therefore, the two parts of the survey are mutually interdependent, with the first part being enhanced by the second, and the second relying on the first.

The rest of this paper is structured as follows. As a prelude to our survey of patterns, Section 2 briefly covers background material and related work on security patterns: in Section 2.1 we discuss the history and necessary concepts in software and security patterns to give a better foundation for the ensuing discussions; in Section 2.2 we provide a brief overview of other surveys and catalogs of general security patterns, as well as pattern classifications and quality evaluations to place our own survey of patterns in context; in Section 2.3 we briefly outline a set of distributed system security concerns; and in Section 2.4 we outline the organization of the patterns in the survey. The survey of patterns itself is contained in Section 3.

In Section 4 we organize the surveyed patterns and evaluate their quality, attaching indicators to each pattern as appropriate. Based on an analysis of the existing patterns and distribution concerns, we subsequently propose ideas and concrete suggestions for a number of new patterns or pattern languages for distributed settings.

In Section 5 we survey pattern-based security methodologies, dividing the approaches according to their maturity,