Journal of King Saud University – Computer and Information Sciences (2016) xxx, xxx–xxx

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

A formal basis for the design and analysis of firewall security policies

Ahmed Khoumsi [a,*], Mohammed Erradi [b], Wadie Krombi [b]

[a] Department of Electrical & Computer Engineering, University of Sherbrooke, Canada
[b] ENSIAS, Mohammed V University, Rabat, Morocco

Received 29 May 2016; revised 16 November 2016; accepted 16 November 2016

KEYWORDS

Firewall security policy;
Automata-based policy;
Completeness verification;
Anomaly detection;
Discrepancy detection;
Mixable policy;
Space and time complexities

Abstract A firewall is the core of a well defined network security policy. This paper presents an automata-based method to study firewall security policies. We first propose a procedure that synthesizes an automaton that describes a security policy given as a table of rules. The synthesis procedure is then used to develop procedures to detect: incompleteness, anomalies and discrepancies in security policies. A method is developed to represent the automaton by a policy qualified as mixable and that has practical utilities, such as ease to determine the whitelist and the blacklist of the policy. The developed procedures have been deeply evaluated in terms of time and space complexities. Then, a real case study has been investigated. The obtained results confirm that the developed procedures have reasonable complexities and that their actual execution times are of the order of seconds. Finally, proofs of all results are provided.

1. Introduction

The world today is fully connected through the internet, while its security remains a big challenge for the research and industrial communities. Network attacks gain a tremendous attention and constitute daily threats and preoccupations of network managers. Firewalls as crucial network security elements have been widely used as the frontier defense against these attacks.

A firewall today is considered as the core element of any well defined network security policy. A firewall security policy consists of filtering rules that are used to filter incoming and outgoing traffic (packets) from the secured network. A badly designed firewall security policy may lead to the acceptance of malicious packets or the rejection of acceptable packets. Therefore, the correct design and analysis of firewall security policies is an important issue that has been addressed by many researchers, such as Acharya and Gouda (2010, 2011), Al-Shaer and Hamed (2004), Al-Shaer et al. (2009), Bryant (1986), Cuppens et al. (2012), Garcia-Alfaro et al. (2008, 2013), Hoffman and Yoo (2005), Kalam et al. (2003), Kamara et al. (2003), Karoui et al. (2013), Liu et al. (2007, 2008, 2010), Lee and Yannakakis (1996), Lu et al. (2007), Mallouli et al. (2007), Madhuri and Rajesh (2013), Mansmann et al. (2012), Pozo et al. (2012), Wool (2004), and Yuan et al. (2006). Henceforth, the terms *policy* and *rule*

* Corresponding author.
E-mail address: Ahmed.Khoumsi@USherbrooke.ca (A. Khoumsi).
Peer review under responsibility of King Saud University.

ELSEVIER Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2016.11.008

Please cite this article in press as: Khoumsi, A. et al., A formal basis for the design and analysis of firewall security policies. Journal of King Saud University – Computer and Information Sciences (2016), http://dx.doi.org/10.1016/j.jksuci.2016.11.008

denote "firewall security policy" and "filtering rule", respectively.

In existing works, each proposed formalism addresses a specific firewall design or analysis aspect to resolve a specific problem. This has motivated the present work where an automata-based methodology is developed to address different problems using a *single* formalism. This methodology is based on the construction of an automaton that describes a policy initially specified by a table of rules. This automaton construction is used to detect incompleteness, anomalies and discrepancies in policies. A method is also proposed to represent the automaton by a table of rules called *mixable policy* that has practical utilities. The proposed procedures have been deeply evaluated in terms of time and space complexities; note that our complexity evaluation is more precise than in the literature. The obtained results are presented as formally proved propositions. Also, we discuss the results of a real-life policy use case taken from Chen et al. (2012).

The rest of the paper is structured as follows. Section 2 presents related work. Preliminaries on policies are given in Section 3. In Section 4, we propose a procedure that constructs an automaton which describes a policy initially specified by a table of rules. Sections 5–10 show how the procedure of Section 4 is applied for a rigorous study of policies. In Section 5, we present a method that determines if a policy is complete. Section 6 defines two categories of anomalies, qualified as conflicting and nonconflicting. In Section 7, we propose methods to detect three types of conflicting anomalies: shadowing, generalization and correlation anomalies. Section 8 proposes methods to detect two types of nonconflicting anomalies: LP-redundancy and MP-redundancy. In Section 9, we show how an automaton describing a policy can be represented in a practical form called mixable policy. Section 10 presents a method that detects and resolves discrepancies between several designs of the same policy. In Section 11, we evaluate the performances of the procedures developed throughout Sections 4 to 10 in terms of space and time complexities. Section 12 discusses the application of our methodology in a real case study. We conclude in Section 13 by recalling our contributions and presenting ideas of future studies. Finally, formal proofs of our results are presented in Appendix A.

2. Related work and contributions

Previous work on firewalls, such as Hoffman and Yoo (2005), Wool (2004), Kamara et al. (2003) provide practical analysis algorithms, for example to test, analyze configuration and detect vulnerability in policies. Acharya and Gouda (2010, 2011), Liu and Gouda (2008, 2010), Al-Shaer et al. (2009) are more fundamental and provide analysis algorithms with estimations of time complexities. Elmallah and Gouda (2014) show that the analyzes of several problems of firewalls are NP-hard.

Madhuri and Rajesh (2013) define an anomaly in a policy by the existence of at least one packet that matches several rules of the policy. Al-Shaer and Hamed (2004), Karoui et al. (2013) present techniques to detect anomalies in a policy, where a policy is specified by a *Policy tree* in Al-Shaer and Hamed (2004) and a *Decision tree* in Karoui et al. (2013).

Garcia-Alfaro et al. (2013), Cuppens et al. (2012) propose methods to study *stateful* anomalies.

Liu and Gouda (2008) show how to detect discrepancies between several designs of the same policy, where the policy is modeled by a *Firewall Decision Diagram* (FDD) defined in Liu and Gouda (2007).

Yuan et al. (2006) introduce a toolkit *Fireman* which detects several types of errors, for example a violation or inconsistency of a policy. Fireman is implemented by *Binary Decision Diagrams* (BDD) (Bryant, 1986).

Mallouli et al. (2007) propose a framework to generate test sequences to check the conformance of a policy to a specification. The system behavior is described by an extended automaton (Lee and Yannakakis (1996)) and the policy that we wish to apply to this system is described by organization-based access control (OrBAC) (Kalam et al., 2003).

Lu et al. (2007) propose a method to verify if two policies are equivalent.

Mansmann et al. (2012) present a tool to visualize and analyze firewall configurations, where the policy is modeled in a hierarchical way.

Pozo et al. (2012) propose CONFIDDENT, a model-driven design, development and maintenance framework for firewalls.

Garcia-Alfaro et al. (2008) propose mechanisms to detect anomalies in configuration rules of security policies.

In each of the above works, a specific problem is solved using a given formalism: anomalies, discrepancies and violation/inconsistencies are studied using a policy tree (Al-Shaer and Hamed, 2004), a FDD (Liu and Gouda 2007) and a BDD (Bryant, 1986), respectively. This observation motivated the works of Krombi et al. (2014), Khoumsi et al. (2014), where the same model of automata is used to solve various problems of policies. The present paper improves the latter two references and completes them with the following new contributions:

1. We show how our approach can be more interesting than FDD, especially in terms of efficiency for deleting, adding, modifying and switching rules in a policy (see Section 4.5).
2. We propose a method to resolve discrepancies between several implementations of a policy (see Section 10).
3. We show how to construct mixable policies and clarify their utility, in particular to determine the whitelist and the blacklist of a policy (see Section 9).
4. We evaluate space and time complexities to execute the automaton of a policy, and show that such automaton execution is more efficient than executing the table of rules of the policy (see Prop. 18).
5. We formally prove all the results given throughout Sections 4 to 11 (including those already in Krombi et al. (2014), Khoumsi et al. (2014)) (see Appendix A).
6. We illustrate the application of our approach to a real-life policy (see Section 12).
7. We explain much more clearly the reason why our complexity evaluation is more precise than in the literature (Sections 11.2, 11.3 and 11.4).

To have a self-contained paper, we present also the main contributions of Krombi et al. (2014), Khoumsi et al. (2014). Indeed, since this paper is the first one that formally proves the results of these two references, their contributions are presented.

Security enforcement is another relevant issue which can be briefly defined as preventing automatically an untrusted system

Please cite this article in press as: Khoumsi, A. et al., A formal basis for the design and analysis of firewall security policies. Journal of King Saud University – Computer and Information Sciences (2016), http://dx.doi.org/10.1016/j.jksuci.2016.11.008