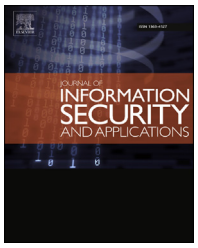
Available online at www.sciencedirect.com

ScienceDirect

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



ASArP: Automated Security Assessment & Audit of Remote Platforms using TCG-SCAP synergies

Mudassar Aslam [a,b,c,*], *Christian Gehrmann* [a], *Mats Björkman* [b]

[a] *SICS Swedish ICT, Isafjordsgatan 22, SE-16440 Kista, Sweden*
[b] *Mälardalen University, Högskoleplan 1, SE-72123 Västerås, Sweden*
[c] *COMSATS Institute of Information Technology, Abbottabad, Pakistan*

ARTICLE INFO

Article history:
Available online 12 January 2015

Keywords:
Platform security
Security automation
SCAP
TCG
Audit
Compliance

ABSTRACT

Many enterprise solutions today are built upon complex distributed systems which are accessible to the users globally. Due to this global access, the security of the host platforms becomes critical. The platform administrators use security automation techniques such as those provided by Security Content Automation Protocol (SCAP) standards to protect the systems from the vulnerabilities that are reported daily; furthermore, they are responsible for keeping their systems compliant to the relevant security recommendations (governmental or industrial). Additionally, third party audit and certification processes are used to increase user trust in enterprise solutions. However, traditional audit and certification mechanisms are not *continuous*, that is, not frequent enough to deal with the daily reported vulnerabilities, and for that matter even auditors expect platform administrators to keep the systems updated. As a result, the end user is also forced to trust the platform administrators about the latest state of the platform. In this paper we develop an automated security audit and certification system ($\mathcal{ASArP}$) which can be used by platform users or by third party auditors. We use security automation techniques for continuous monitoring of the platform security posture and make the results trustworthy by using trusted computing (TCG) techniques. The prototype development of $\mathcal{ASArP}$ validates the implementation feasibility; it also provides performance benchmarks which show that the $\mathcal{ASArP}$ based audit and certification can be done much more frequently (e.g. daily or weekly). The feasibility of $\mathcal{ASArP}$ based continuous audits is significantly better than traditional platform audits which are dependent on the physical presence of the auditors, thus making frequent audits much more expensive and operationally infeasible.

1. Introduction

The availability of high speed internet (Aslam et al., 2013) and its extended reach has been instrumental in increasing the magnitude of the user base to over 2.4 billion (Internet World Stats, 2014). Considering the current (Mell and Grance, 2011) and future trends (The 10 Hot Consumer Trends, 2013), this number is expected to grow even more rapidly. This increase in the connected user base opens up many business opportunities useful for both service providers and their users. However, the flip side of this global connectivity is that it also

* *Corresponding author*. COMSATS Institute of Information Technology, Abbottabad, Pakistan.
E-mail addresses: mudassar@sics.se (M. Aslam), chrisg@sics.se (C. Gehrmann), mats.bjorkman@mdh.se (M. Björkman).
http://dx.doi.org/10.1016/j.jisa.2014.09.001
2214-2126/

increases the attack surface for attackers whereby they can target any system, (from) anywhere in the world. If this is coupled with the current rate of vulnerability reporting (which is more than 15 vulnerabilities per day (CVE and CCE Statistics, 2014)), one can understand how quickly a system can become vulnerable and available to attack by the global hacker community. The situation gets even more alarming when a target platform is providing a critical service to its end users. Currently, security automation techniques are used by service providers for the monitoring and redressal of the problems. However, there are various shortcomings in the existing mechanisms of platform security management (*monitoring*, *auditing* and *certification*) which we discuss in this paper. One main requirement that arises to deal with the challenges of today's internet connected world, is to replace manual methods of security management with automated techniques which are trustworthy, robust, rapid, consistent and continuous.

The focus of this paper is trustworthy and automatic security evaluation of computing platform software components. We consider two factors in defining and evaluating the security state of a platform. First, that the correct list of software is running on the platform which we refer to as *software stack integrity*. The integrity of the software stack is important in determining the overall security state of the platform because the *presence* of a "good" software (e.g. antivirus, firewall) can improve the security state, whereas the presence of a "bad" software (e.g. malware) can degrade the security posture; similarly, the *absence* of a required patch/update can expose the platform to known vulnerabilities which can be exploited locally (by insiders) or remotely (by external attackers). The software stack integrity is not enough to guarantee the security state of the platform; for example, a legitimate software running with bad configuration can be used to launch various remote attacks (e.g. an SSH server running with wrong permissions, such as 'remote login allowed using password and as root', can be exploited to brute force the password and get administrator access). Therefore, the second factor, i.e. the configuration of the running software especially the ones which are security critical, is also very important for determining the platform security state.

There is no hard-line definition of a secure/insecure software stack, or a right/wrong software configuration; because the security requirements vary in different use cases and across organizations. However, various security standards do exist which are developed and mandated by the industry (PCI, 2010) and/or governments (Recommended Security Controls for Federal Information Systems and Organizations, 2010) for a particular use case. Therefore, a platform is considered secure or trustworthy if it *complies* with the mandated security requirements for that particular use case. In order to ensure these compliance requirements, there are many solutions which are either currently used in industry such as Security Content Automation Protocol (SCAP) (Waltermire et al., 2011), or they are still in research phase (Sailer et al., 2004; Huang and Peng, 2009) (presented in Section 3).

This paper (Aslam et al., 2013) 1 describes an approach for Automated Security Assessment and Auditing of Remote Platforms ($\mathcal{ASArP}$). This work is an extension of our previous work (Aslam et al., 2013) wherein the techniques of combining TCG-SCAP synergy were presented. In this paper, apart from describing the suggested approach, we provide a proof of concept implementation that is also evaluated from performance and security points of views. The solution we present shows how to build the tools needed to remotely evaluate the software stack integrity and software configurations in a trustworthy way. This automated evaluation can be used to audit and certify the platforms more frequently (i.e. continuous) by the users themselves or by the trusted third parties (TTP). Our proof-of-concept prototype validates the implementation feasibility of the $\mathcal{ASArP}$; however, there are some deployment/management challenges which are presented in Section 6. In calculating the performance overhead of using the $\mathcal{ASArP}$, the complete evaluation and certification process takes around 45 min which is 3.1% of the day time if platforms are to be evaluated on a daily basis. However, the impact can be further reduced if the evaluation is done during off-peak hours and on a weekly (or bi-weekly) basis. It is important to mention here that this time can be improved further when real platforms with much better computing power are used. Moreover, the vulnerability assessments and configuration tests which take around 99% of the total time are not newly introduced tests, that is, they are already part of the existing platform security management solutions, but without any automated auditing and certification framework as provided by $\mathcal{ASArP}$. Therefore, we think that $\mathcal{ASArP}$ provides a feasible solution with respect to its implementation and performance. The main contributions of this paper are highlighted below:

- We analyze the shortcomings and limitation of existing platform security management solutions.
- We present a standards based, automated remote platform security evaluation framework which can be used for frequent platform level audits and certifications.
- Our solution can be useful in shifting the source of user trust from the platform owner to a trusted third party.
- We validate the implementation and performance feasibility by developing a proof of concept prototype.

The rest of the paper is organized as follows. We use a threat model presented in Section 2 which is also discussed with respect to existing solutions in Section 3. We present the details of our proposed $\mathcal{ASArP}$ framework in Section 4 which aims at addressing the identified threats and solve the limitations of existing solutions. We validate the feasibility of our solution by implementing a prototype which is described in Section 5. In Section 6, we analyze the security of the proposed solution and discuss its implementation and deployment challenges. Finally, we present related work in Section 7 before we conclude in Section 8.

2. Threat model

The security of a platform can be compromised due to incorrect platform setup, deliberately or indeliberately, which can expose the platform to many attacks. We consider a threat