Vulnerability Discovery Modeling for Open and Closed Source Software

Article *in* International Journal of Secure Software Engineering · October 2016

DOI: 10.4018/IJSSE.2016100102

3 authors:

Ruchi Sharma
International Management Institute Kolkata
22 PUBLICATIONS 99 CITATIONS
SEE PROFILE

Ritu Sibal
Netaji Subhas Institute of Technology
27 PUBLICATIONS 313 CITATIONS
SEE PROFILE

Avinash K. Shrivastava
International Management Institute Kolkata
104 PUBLICATIONS 504 CITATIONS
SEE PROFILE

International Journal of Secure Software Engineering
Volume 7 • Issue 4 • October-December 2016

Vulnerability Discovery Modeling for Open and Closed Source Software

Ruchi Sharma, Department of Computer Engineering, Netaji Subhas Institute of Technology, New Delhi, India

Ritu Sibal, Department of Computer Engineering, Netaji Subhas Institute of Technology, New Delhi, India

A.K. Shrivastava, Amity Center for Interdisciplinary Research, Amity University, Noida, India

ABSTRACT

With growing concern for security, the researchers began with the quantitative modeling of vulnerabilities termed as vulnerability discovery models (VDM). These models aim at finding the trend of vulnerability discovery with time and facilitate the developers in patch management, optimal resource allocation and assessing associated security risks. Among the existing models for vulnerability discovery, Alhazmi-Malaiya Logistic Model (AML) is considered the best fitted model on all kinds of datasets. But, each of the existing models has a predefined basic shape and can only fit datasets following their basic shapes. Thus, shape of the dataset forms the decisive parameter for model selection. In this paper, the authors have proposed a new model to capture a wide variety of datasets irrespective of their shape accounting for better goodness of fit. The proposed model has been evaluated on three real life datasets each for open and closed source software and the models are ranked based on their suitability to discover vulnerabilities using normalized criteria distance (NCD) technique.

KEYWORDS

Closed Source, Modeling, Normalized Criteria Distance (NCD), Open Source, Prediction, Ranking, Vulnerability

1. INTRODUCTION

By virtue of ongoing advances in networking, internet has evolved as a necessity but, the connectivity of computing systems has further raised the software security concerns. Substantial amount of efforts are made every year to detect, publish, and fix security vulnerabilities in software products. With increasing number of vulnerabilities

DOI: 10.4018/IJSSE.2016100102

in the system, the numbers of possible security breaches also shows an upward trend. These concerns marked the outset for quantitative modeling of the process of vulnerability discovery. Vulnerability discovery models assist the developers in patch management, optimal resource allocation and assessment of associated security risks.

Software community frequently faces the challenge to update, protect, maintain, and improve the software product. The testing and maintenance of software depends on its development strategy which could be closed source software development (CSSD) or open source software development (OSSD). Owing to the structural differences between open and closed source software, their maintenance and support also varies. (Potdar and Chang, 2004). In open source software the source code is widely available due to which the potential attackers can easily analyze the source code for possible vulnerabilities. However same is not true in case of closed source software. The availability of resources in case of open source software is typically higher than closed source software. Due to these differences, the trend of vulnerability discovery in both these software communities also shows significant differences. All software- be it open or closed-source are inherently insecure and the growing demand of open source software (OSS) in recent years has motivated researchers to identify vulnerability trends in patching strategies as well as on up gradations. Since these attributes show differences when analyzed for open and closed source communities, as a result, the vulnerability trends in both these software communities show deviation in behavior when compared to each other (Raymond, 1999),(Robles, 2004),(Llanos & Castillo, 2012), (Schreyn, 2009), (Schreyn & Kadura, 2009). These trends are captured using quantitative modeling techniques termed as Vulnerability Discovery Models.

Work has been done in the past to perform quantitative characterization of security vulnerabilities and to find potential number of vulnerabilities in a software. Vulnerability Discovery models (VDMs) facilitate the task of resource allocation for security testing, development of security patches and scheduling. VDM assess the probability of risk and help the developer to allocate resources required to handle potential breaches. Estimating the number of vulnerabilities in a system will also support the critical decision of when to stop testing for vulnerabilities in order to release a stable version.

In this paper, we have proposed a new VDM to find the number of vulnerabilities and their distribution with time in a software system by using analytical modeling techniques while enumerating the difference in vulnerability detection patterns for open and closed source software. We have evaluated the applicability and precision levels of existing and proposed VDM on datasets from open source and proprietary or closed source software. The vulnerability detection rate in open and closed source software show some significant differences owing to the differences in strategies followed during their development and testing. The lifecycle of an open and closed source software have differences which in turn affect their vulnerability discovery patterns (Groot, Kugler, Adams & Gousios, 2006), (Llanos & Castillo, 2012).

Paper organization is as follows: Related work is presented in Section 2. Section 3 proposes a new methodology for software vulnerability discovery based on Gamma