

Seminar Report

Tam Pham

October 4th, 2004

1 Paper Overview

Understanding Data Lifetime via Whole System Simulation

By: J.Chow, B.Pfaff, T.Garfinkel, K.Christopher, M.Rosenblum

Institution: Department of Computer Science, Stanford University

Publication: Proceedings of the 13th USENIX Security Symposium

2 Paper Summary

Sensitive data such as passwords, credit card numbers, military data are handled everyday by general applications such as databases, web browsers, email systems. The longer this kind of data resides in the system, the more vulnerable the system is. The authors are particularly concerned with two issues: how long this data alive (in memory or persitent storage) and where it is propagated (via buffer, log files, etc). Minimizing data lifetime is hard due to several reasons such as the operating system may not have secure memory locking or the complier favours optimization over zeroing out memory. Additionally, it is not easy to track data lifetime using the traditional static and dynamic program analysis approaches as many components such as operating systems, libraries, programming languages runtim can be employed in one single transaction.

The paper introduces a new tool called TaintBochs which records sensitive workload and analyzes the results using the whole system simulation approach. The system tracks these data at hardware level and propagates the tainting information across the operating systems, language, application boundaries at a whole system level. TaintBoshs is used to evaluate the data lifetime management of several large, real world applications such as Mozilla, Apache-Perl, Windows 2000. The results show that sensitive data are not properly handled in these systems. The authors then provide an analysis on reasons why these applications fails to address the data lifetime problem and suggest some changes in these applications to make them more secure.

3 General Comments

The research presented in the paper is well motivated. The authors present clearly why long data lifetime poses a big threat and why minimizing data lifetime is hard. They also provide a good explanation why a whole system simulation could help in tracking this kind of data. Additionally, the paper presents interesting experiments with familiar applications, identifies their faults in dealing with sensitive data; which raises our awareness when using them.

However, the paper does not contribute significantly to the literature as it is basically just a tool for tracking sensitive data. The tool is even not complete and easy to use as the user has to define certain operations, for example, s/he needs to tell TaintBochs how to recognize a one-way function. Together with its weaknesses in slow performance and large log files, the tool seems to be more appropriate for developers who are interested in writing secure codes rather than for public use. The same reasons also imply that it is not suitable for real time systems. Additionally, at the beginning of the paper, many problems are introduced but none seem to have a reasonable solution at the end of the paper.

Another weakness of this paper is the way the authors conducted their experiments. They claim that they choose general applications and test their security in dealing with data lifetime. However, no justification is made to counter the fact that they do not choose a security application such as SSL to analyze. General applications tend to favour usability over security, and hence are potentially very insecure. Security applications will give a more interesting insight about the problem.

4 Acceptance Level

- Strongly Accept: 1 vote
- Accept: 10 votes
- Reject: 3 votes
- Strongly Reject: 0 vote