Vulnerability Detection and Correction with WAP

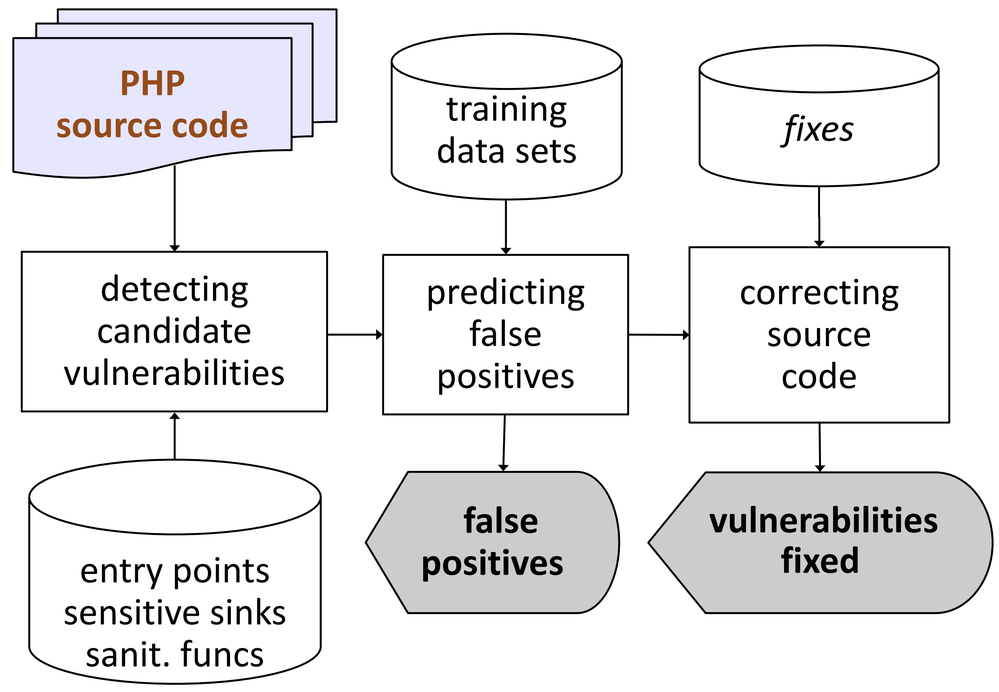
Contact: Ibéria Medeiros, Nuno Neves
{imedeiros, nuno}@di.fc.ul.pt
FCiências.ID / LASIGE, Faculdade de Ciências, Universidade de Lisboa
www.navigators.di.fc.ul.pt

Overview

Web applications are used in several contexts of the smart grid, allowing for instance the creation of flexible user interfaces. The security of these applications continues to be a challenging problem, as programmers make mistakes and leave vulnerabilities in the source code .

The WAP tool aims to assist on the protection of web applications. It helps programmers find vulnerabilities in their source code and correct them, by automatically fixing the flaws that were found. For more accurate detection, the tool integrates machine learning to predict if the vulnerabilities that were identified are real or false alarms. Experts in secure coding can configure WAP for new vulnerability classes without programming.

Web Application Protection



Approach

Web Application Protection (WAP) aims to protect automatically smart grid applications by analyzing and fixing their source code.

Our approach is based on:

TAINT ANALYSIS to find vulnerabilities

WAP searches for ***input validation vulnerabilities*** in PHP source code, such as XSS and SQL injection. It does ***taint analysis***, checking if inputs can reach sensitive functions (sensitive sinks) without proper sanitization or validation. The lists of input entry points, sensitive sinks, and sanitization/validation functions are produced by humans. However, our research has shown that taint analysis tends to produce false positives, i.e., to identify bugs that do not exist in practice.

DATA MINING to predict false positives

WAP uses a second form of analysis - ***data mining*** - to refine the results of taint analysis, classifying them as being false positives or real vulnerabilities. This analysis is based on ***learning about attacks automatically*** using a labeled data set of true and false positives.

CODE FIXES to remove vulnerabilities

WAP also removes identified bugs automatically using ***code fixes***. These fixes do essentially proper validation or sanitization of user input before it is used in a sensitive sink. Fixes are PHP functions developed by us, as security problems were identified in several of sanitization functions available in PHP. They are inserted in sensitive sinks or close to them.

Experiments

Research questions:

1. Is WAP able to process a large set of PHP applications? *(see Table 1)*
2. Does the tool detect the vulnerabilities that it was programmed to detect? *(see Table 1)*
3. Is it more accurate and precise than other tools that do not combine taint analysis and data mining? *(see Table 2)*

We compare WAP with Pixy and PhpMinerII. Pixy does taint analysis such as WAP to discover vulnerabilities, but only detects SQL injection and XSS vulnerabilities. PhpMinerII does data mining of program slices that end at a sensitive sink, regardless of data being propagated through them starting at an entry point or not. PhpMinerII does this analysis to predict vulnerabilities, whereas WAP uses data mining to predict false positives in vulnerabilities detected by the taint analyzer.

Table 1. Two sets of experiments with WAP.

Metric	Exper 1	Exper 2	Total
Web applications	45	54	**99**
WP plugins	--	115	**115**
Files	6,700	8,300	**15,000**
MLoC	1,3	2,0	**3,3**
Vulnerabilities	388	582	**970**
Zero-day	1	519	**520**

Table 2. Evaluation between WAP, Pixy and PhpMinerII.

Metric	WAP	Pixy	PhpMinerII
accuracy	**92,1%**	44,0%	87,2%
precision	**92,5%**	50,0%	85,2%

Vulnerabilities

The WAP tool can detect eight classes of input validation vulnerabilities:

- SQL injection
- XSS (reflected and stored)
- Remote file inclusion (RFI)
- Local file inclusion (LFI)
- Directory/Path traversal (DT/PT)
- Source code disclosure (SCD)
- OS command injection (OSCI)
- PHP command injection (PHPCI)

References

[1] Ibéria Medeiros, Nuno Neves, Miguel Correia, *Automatic Detection and Correction of Web Application Vulnerabilities using Data Mining to Predict False Positives,* In Proc. of the International World Wide Web Conference (WWW), Seoul, Korea, April 2014

[2] ___. *Detecting and Removing Web Application Vulnerabilities with Static Analysis and Data Mining*, IEEE Transactions on Reliability, Vol. 65, No. 1, pp 54-69, March 2016

[3] ___. *Equipping WAP with WEAPONS to Detect Vulnerabilities*, In Proc. of the 46th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Toulouse, France, June 2016

SEGRID Security for smart Electricity GRIDs

www.segrid.eu info@segrid.eu

This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 607109