Proceedings of the Tenth International Conference on Informatics and Information Technology

CIIT 2013
April 18-21, 2013, Molika, Bitola, Macedonia

Editors:
Igor Mishkovski and Sasko Ristov

Skopje, 2014
Preface

Continuing the tradition, the decennial 10th Conference of Informatics and Information Technology – CIIT2014 was held for the tenth time, traditionally in Bitola, Macedonia during April 18-21, 2013. The first 8 years this conference was being organized by the Institute of Informatics at Faculty of Natural Sciences and Mathematics, while these last two years it was organized by the Faculty of Computer Science and Engineering (FCSE), Ss. Cyril and Methodius University in Skopje, which is the successor of the two largest institutions in the area of informatics and computer technologies in Macedonia, i.e., the Institute of Informatics at Faculty of Natural Science and Mathematics and the Institute of Computer Techniques and Informatics at Faculty of Electrical Engineering and Information Technologies. FCSE continues the tradition of giving the opportunity to researchers to present their latest results in the field of Informatics and Information Technologies.

During the three days of the conference, 66 accepted papers were presented in 12 regular sessions. Additionally, a special PhD student workshop was held with 10 presenters, as well as a student session with 9 accepted presentations of FCSE’s student projects.

Professor Marjan Gusev, the founder/founding father of the CIIT conference, gave the invited talk “Ten Years of CIIT”.

Apart from the variety of topics and areas that were covered by the presentations, the participants broadened their scientific knowledge and social networking during the several social events.

We believe that the growing trend of papers and presenters will continue to promote the CIIT conference as an influential international conference with great impact on ICT research and development.

The Editors,

Igor Mishkovski and Saso Ristov
Editors
Igor Mishkovski and Sasko Ristov

Organizing Committee
Nevena Ackovska, PhD
Mile Jovanov, MSc
Magdalena Kostoska, MSc
Aleksandra Bogojeska, MSc
Petre Lameski, MSc
Program committee

Smile Markovski, PhD
Dragan Mihajlov, PhD
Margita Kon-Popovska, PhD
Vancho Kusakatov, PhD
Lupcho Kocarev, PhD
Marjan Gushev, PhD
Katerna Zdravkova, PhD
Suzana Loshkovska, PhD
Zhaneta Popeska, PhD
Kosta Mitreski, PhD
Verica Bakeva, PhD
Vladimir Trajkovikj, PhD
Ana Madevska Bogdanova, PhD
Dejan Gjorgjevikj, PhD
Dimitar Trajanov, PhD
Sonja Gievska, PhD
Andrea Kulakov, PhD
Ljupcho Antovski, PhD
Slobodan Kalajdzhiski, PhD
Nevena Ackovska, PhD
Marija Mihova, PhD
Sonja Filiposka, PhD
Anastas Mishev, PhD
Ivan Chorbev, PhD
Goce Armenski, PhD
Dejan Spasov, PhD
Goran Velinov, PhD
Vesna Dimitrova, PhD
Boro Jakimovski, PhD
Lasko Basnarkov, PhD
Igor Trajkovski, PhD
Ivica Dimitrovski, PhD
Igor Mishkovski, PhD
Gjorgji Madzarov, PhD
Smilka Janeska-Sarkanjac, PhD
Sasko Ristov, PhD
Aleksandra Mileva, PhD
Pece Mitrevski, PhD
Sime Arsenovski, PhD
Adrijan Bozinovski, PhD
List of Participants

Aleksandar Kotevski
Adrijan Bozhinovski
Aleksandar Bahtovski
Aleksander Donevski
Aleksander Stojmenski
Aleksandra Bogojeska
Aleksandra Kanevce
Aleksandra Mileva
Aleksandra Popovska-Mitrovikj
Ana Madevska Bogdanova
Anastas Mishev
Andrea Kulakov
Andrea Naumovski
Andrej Gajduk
Biljana Risteska Stojkoska
Biljana Stojchevska
Biser Dugalic
Bojan Ilioski
Bojan Kostadinov
Bojan Najdenov
Bojana Koteska
Boro Jakimovski
Christian Huemer
Dalibor Serafimovski
Daniel Kareski
Daniela Kjurchevska
Danijel Novakovic
Darko Martinovikj
Dejan Gjorgjevikj
Dejan Koneski
Dejan Kovachev
Dejan Spasov
Dimitar Trajanov
Dobre Blazhevski
Dragan Jovanovski
Dragan Mihajlov
Dragi Zlatkovski
Eftim Zdravevski
Elena Janevska
Elena Stojanova
Elena Vlahu-Gjorgieviska
Elvis Imeroski
Emil Stankov
Eva Blazhevskaja
George Tanev
Georgina Mirceva
Gjorgi Kakasevski
Gjorgji Madjarov
Goce Gjorgijiski
Goran Stanoiev
Goran Velinov
Goran Velkoski
Hristina Mihajloska
Igor Kulev
Igor Mishkovski
Ilija Jolevski
Ilinka Ivanoska
Ivan Chorbev
Ivan Kitanovski
Ivana Gjorgievska
Ivica Dimitrovski
Jasna Veljanovska
Josip Kolic
Jovan Janevski
Jugoslav Achkoski
Katarina Trojancanec
Kire Jakimoski
Kliment Mahoski
Kosta Mitreski
Kristina Spirovskaja
Leonid Djinevski
Ljupco Antovski
Magdalena Kostoska
Maja Siljanoska
Marija Mihova
Marija Petkowska
Marina Vasileva
Marjan Gusev
Martina Toceva
Mile Jovanov
Milos Jovanovik
Monika Simjanoska
Natasa Suteva
Nevena Ackovska
Nevena Serafimova
Nevenka Pop-Angelova
Nikola Koteli
Peca Mitrevski
Petre Lameski
Predrag Tasevski
Radmila Koteska
Radoslav Bozhinovski
Riste Stojanov
Roman Brandt
Sanja Angelovska
Sasha Ivanovska
Sasho Gramatikov
Sasko Ristov
Saso Koceski
Sime Arsenovski
Slobodan Kalajdziski
Smile Markovski
Sonja Filiposka
Sonja Gievska
Svetlana Loshkovska
Tatjana Vasileva-Stojanovska
Tome Dimovski
Tony Janeski
Tony Malinovski
Trajche Kocev
Vangel Ajanovski
Vasilev Gangelovski
Vasil Pavlov
Vasko Popovski
Veno Pachovski
Verica Bakeva
Vesna Dimitrova
Vesna Kirandziska
Vladimir Popovski
Vladimir Trajkovik
Vladimir Zdravevski
Zaneta Popeska
Zlatko Trajcheska
Zoran Milevski
Zoran Zdravev
# Table of Contents


| STUDENT CLASSIFICATION IN E-LEARNING SYSTEMS | Aleksandar Kotevski, Radmila Kotevska | 3 |
| ON-LINE LEARNING ENVIRONMENT FOR TEACHING THE LOGO PROGRAMMING LANGUAGE | Sanja Angelovska, Dejan Gjorgjevikj | 7 |
| TEACHERS’ ACCEPTANCE OF THE SMART BOARD IN PRIMARY EDUCATION SCHOOLS, QOE ANALYS | Toni Malinovski, Marina Vasileva, Tatjana Vasileva-Stojanovska, Vladimir Trajkovik | 11 |
| EFFICIENCY IN THE USAGE OF E-LEARNING MOODLE IN THE PROCESS OF EDUCATION | Zoran Milevski, Nevenka Pop-Angelova, Zoran Zdravev | 16 |
| ANALYSIS OF THE BENEFIT FROM COSTLESS AND COMMERCIAL SOFTWARE THROUGH COMPARISON BETWEEN OPEN SOURCE SOFTWARE AND CLOSED SOURCE SOFTWARE | Goran Stanoev, Adrijan Bozihnovski, Eva Blazhevska | 20 |
| MOODLE IMPLEMENTATION OF ACTIVITY FOR ON-LINE COLLABORATIVE BUILDING | Mile Jovanov, Marjan Gushev, Darko Martinovikj, Stefan Spasovski | 24 |
| BUSINESS CONTEXT SENSITIVE BUSINESS DOCUMENTS: AN ONTOLOGY BASED BUSINESS CONTEXT MODEL FOR CORE COMPONENTS | Danijel Novakovic, Christian Huemer | 29 |
| RECOMMENDATION ALGORITHM BASED ON COLLABORATIVE FILTERING AND ITS APPLICATION IN HEALTH CARE | Igor Kulev, Elena Vlahu-Gjorgievska, Vladimir Trajkovik, Saso Koceski | 34 |
| SOCIAL-BASED GUIDANCE AND SELF-ADAPTIVITY WITHIN ENROLMENT PROCESSES | Vangel Ajanovski | 39 |
| IT ASPECTS OF THE APPLICATION OF ONTOLOGIES IN ACADEMIC DIGITAL LIBRARIES | Daniela Kjurchevska | 44 |
| SCRAPECARDS: FLEXIBLE FLASH CARDS FOR UBIQUITOUS MICRO-LEARNING | Roman Brandt, Dejan Kovachev | 47 |
| WEB-BASED DISASTER AND CRISIS MANAGEMENT SYSTEM | Vasko Popovski, Bojan Kostadinov, Riste Stojanov, Igor Mishkovski, Dimitar Trajanov | 52 |
| LINKED OPEN DRUG DATA FROM THE HEALTH INSURANCE FUND OF MACEDONIA | Milos Jovanovik, Bojan Najdenov, Dimitar Trajanov | 56 |

**Intelligent Systems, Robotics, Bioinformatics**

| WORD-SPACE APPROACH TO CASE-BASED RETRIEVAL | Ivan Kitanovski, Katarina Trojanac, Ivica Dimitrovski, Suzana Loshkovska | 65 |
| AUTHONOMUS BEHAVIOUR MODELS FOR AR DRONE | Stefan Spasovski, Nevena Ackovska | 69 |
The 10th Conference for Informatics and Information Technology (CIIT 2013)

Computer Networks, Multimedia and Digitalization
ALGORITHM FOR COST-EFFECTIVE DISTRIBUTION OF VOD CONTENTS - Sasho Gramatikov, Gece Gjorgjijoski ..................................................................................................................155
ERP SYSTEM BASED ON A MODEL FOR GIS POSITIONING OF FIBRE OPTIC NETWORK - Dejan Konevski, Ilija Jolevski, Dragan Mihajlov ...........................................................................161
ANALYSIS OF THE PROBLEM OF MACEDONIAN FOLK DANCE RECOGNITION - Darko Martinovikj, Marija Mihova, Mile Jovanov ..........................................................165
AN ONLINE PLAYLIST MANAGEMENT SYSTEM - Aleksandar Stojmenski, Dragan Mihajlov, Ivan Chorbev ...........................................................................................................169

Theoretical Foundations of Informatics and Signal Processing
OPTIMIZATION OF THE POLYNOMIAL GREEDY SOLUTION FOR THE SET COVERING PROBLEM - Stefan Spasovski, Ana Madevska Bogdanova ..................................................................175
SOME NEW RESULTS FOR RANDOM CODES BASED ON QUASIGROUPS - Aleksandra Popovska-Mitrovikj, Smile Markovski, Verica Bakeva .......................................................................178
VIRTUAL STUDIO TECHNOLOGY AND ITS APPLICATION IN DIGITAL MUSIC PRODUCTION - George Tanev, Adrijan Božinovski ..........................................................182
GESTURE RECOGNITION SOLUTION FOR PRESENTATION CONTROL - Darko Martinovikj, Nevena Ackovska ...............................................................................................187
MATRIX PRESENTATION OF QUASIGROUPS OF ORDER 4 - Maja Siljanoska, Marija Mihova, Smile Markovski .................................................................................................192
PROGRESS REPORT OF GREEDY ALGORITHMS IN CODING THEORY - Dejan Spasov .................................................................................................................................197

Security and Cryptography
THE HARDWARE PERFORMANCE OF AUTHENTICATED ENCRYPTION MODES - Hristina Mihajloska .................................................................................................................201
FREQUENTLY-OCCURRING SECURITY INCIDENTS - Predrag Tasevski .................................................................................................................................205
SECURITY PENETRATION TEST ON FCSE’S IT SERVICES - Radoslav Bozhinovski, Vesna Dimitrova, Boro Jakimovski, Sasko Ristov ..........................................................208
MODES OF OPERATION OF THE AES ALGORITHM - Dobre Blazhevski, Adrijan Bozhinovski, Biljana Stojchevska, Veno Pachovski .................................................................212
ANALYZING THE CRYPTOGRAPHIC PROPERTIES OF THE BOOLEAN REPRESENTATION OF QUASIGROUPS OF ORDER 4 - Vesna Dimitrova, Zlatka Trajcheska, Marija Petkovska ..................................................................................217
EVALUATION AND TESTING OF SEVERAL FREE/OPEN SOURCE WEB VULNERABILITY SCANNERS - Natasa Suteva, Dragi Zlatkovski, Aleksandra Mileva ........................................................................221
SOCIAL NETWORKS – WHO IS ACCESSING AND USING YOUR (PRIVATE) INFORMATION? - Ivana Gjorgievka, Adrijan Bozhinovski, Eva Blazhevksa, Veno Pachovski ...........................................................................225
Computer Architecture and Parallel Processing
VARIOUS ALGORITHM BEHAVIOR IN THE CLOUD - Goran Velkoski, Sasko Ristov and Marjan Gusev ..........................................................231
COMPARISON OF RECTANGULAR MATRIX MULTIPLICATION WITH AND WITHOUT BORDER CONDITIONS - Petre Lameski, Igor Mishkovski, Sonja Filiposka, Dimitar Trajanov, Leonid Djinevski ..........................................................235
VHDL IP CORES ONTOLOGY - Vladimir Zdraveski, Dimitar Trajanov ..........................................................240

Distributed Systems, GRID and Cloud Computing
A SOFTWARE ENGINEERING PERSPECTIVE FOR HIGHER QUALITY GRID DISTRIBUTED DEVELOPMENT - Bojana Koteska, Anastas Mishev ..........................................................247
TRANSFORMING COMPUTER LAB INTO A MINI HPC CLUSTER - Jovan Janevski, Dragan Jovanovski, Josip Kolic, Jasna Veljanovska, Elvis Imeroski, Leonid Djinevski, Sime Arsenovski..........................................................252
A CONCEPT FOR A SMART WEB PORTAL DEVELOPMENT IN INTELLIGENCE INFORMATION SYSTEM BASED ON SOA - Jugoslav Achkoski, Vladimir Trajkovik, Nevena Serafimova ..........................................................256
EVALUATION OF POSSIBLE CLOUD STANDARDS - Magdalena Kostoska, Marjan Gusev ..........................................................................................................................261
IMPLEMENTING COMPUTATIONAL RESOURCES IN DISTRIBUTED DATABASES - Gjorgi Kakasevski, Anastas Mishev..........................................................264
NESSUS OR METASPLOIT: SECURITY ASSESSMENT OF OPENSTACK CLOUD - Aleksandar Donevski, Sasko Ristov, Marjan Gusev ..........................................................................................................................269
WEB SYSTEM FOR INTERLOCUTOR’S AVAILABILITY - Ljupco Antovski, Elena Janevska..........................................................................................................................274
ANALYSIS OF CLOUD PORTABILITY - Aleksandar Bahtovski, Marjan Gusev ..........................................................................................................................280

Wireless and Mobile Computing
MOBILE AD HOC COMMUNICATION – THE ABILITY OF NETWORKING WITHOUT NETWORK - Trajche Kocev, Pece Mitrevski, Tome Dimovski ..........................................................................................................................287
PERFORMANCE ANALYSIS OF THE REAL TIME SERVICE CLASSES IN MOBILE WIMAX - Kire Jakimoski, Toni Janevski ..........................................................................................................................292
CHALLENGE OF CREATING CROSS - PLATFORM MOBILE APPLICATIONS - Dalibor Serafimovski ..........................................................................................................................296
NODES LOCALIZATION IN 3D WIRELESS SENSOR NETWORKS - Biljana Risteska Stojkoska ..........................................................................................................................301
COMPARISON OF DIFFERENT DATA PREDICTION METHODS FOR WIRELESS SENSOR NETWORKS - Biljana Risteska Stojkoska, Kliment Mahoski ..........................................................................................................................307
EVALUATION AND TESTING OF SEVERAL FREE/OPEN SOURCE WEB VULNERABILITY SCANNERS

Nataša Šuteva
Faculty of Computer Science, UGD
Štip, Macedonia

Dragi Zlatkovski, Aleksandra Mileva
Faculty of Computer Science, UGD
Štip, Macedonia

ABSTRACT
Web Vulnerability Scanners (WVSs) are software tools for identifying vulnerabilities in web applications. There are commercial WVSs, free/open source WVSs, and some companies offer them as a Software-as-a-Service. In this paper, we test and evaluate six free/open source WVSs using the web application WackoPicko with many known vulnerabilities, primary for false negative rates.

I. INTRODUCTION
Our everyday live heavily depends on using different web applications, as web e-mail clients, web instant messaging clients, Voice over IP services, e-learning portals, social networks, electronic banking, e-commerce platforms, etc. Because of this, the web applications became the most interest target for attackers to gain an authorized account access, steal sensitive data and identity, etc.

The OWASP (Open Web Application Security Project) Top Ten 2013 [16] offers a list of the most critical Web application vulnerabilities, including different types of injection, broken authentication and session management, cross-site scripting, cross-site request forgery, etc. This list is often used also as a minimum standard for website vulnerability assessment and PCI compliance according to Payment Card Industry Data Security Standard (PCI DSS). Classification of web application vulnerabilities can be found also in Common Vulnerabilities and Exposures database [3] and Web Application Security Consortium (WASC) Threat Classification v2.0 [20].

Web Application Security Scanners (WASSs) or Web Vulnerability Scanners (WVSs) are software tools for identifying potential vulnerabilities in the web applications, independently of the particular technology used for their implementation. They access the web applications in the same manner as user do, through the web front-end. Usually they are black-box testers, because they do not have access to the source code, so they detect vulnerabilities by actually performing attacks or by looking for known vulnerabilities and report potential exposures.

The beauty of WVSs hides in automatically and cost-effective conduction of security checks and production of the final report. Almost every report includes a remedy for found vulnerability, which is necessary for PCI compliance. Today there are more than 130 scanning vendors approved for PCI compliance [11]. Vulnerability scanning is essential part of maintaining security in a given organization and should be used continuously, especially when new version of web application or new equipment or technology is planning to use. But WVSs are not all-in-one oracles, they are not capable of detecting all of the possible vulnerabilities and attack vectors that exist. There are several reports showing that today WVSs fail to detect a significant number of vulnerabilities in test applications [1, 4, 12, 14, 15, 22]. Bau et al [1], testing eight WVSs, showed that WVSs need to be improved in detection of the “stored” and second-order forms of XSS and SQLI, and in understanding of active content and scripting languages. Khoury [7, 8] analyzed three state-of-art black box WVSs against stored SQLI, and their results showed that stored (persistent) SQLI are not detected even when these automated scanners are taught to exploit the vulnerability. They propose also a set of recommendations for increasing a detection rate in WVSs for this type of vulnerability. Doupé et al [4] tested eleven WVSs, and found that eight out of sixteen vulnerabilities were not detected by any of the used scanners. They discuss also a critical limitations of current WVSs, lack of better support for well known, pervasive technologies as JavaScript and Flash, and the need for more sophisticated algorithms to perform “deep” crawling and track the state of the application under test.


For evaluating and testing WVSs, vulnerable test applications are needed. These applications need to have exactly listed known vulnerabilities, so one can obtain the false positive and false negative rates also. Unfortunately, no standard test suite is currently available. There are several well-known, publicly-available, vulnerable web applications like DVWA (Dam Vulnerable Web Application) [13] and WebGoat [17], but their design is focused more on teaching web application security rather than testing WVSs. The exception is the realistic and fully functional web application WackoPicko [21] with 16 known vulnerabilities, created by A. Doupé, and used in [4] for their testing. We use this web application for our experiments. Additionally, WASC [19] has published evaluation criteria for web application scanners.

Because most of the research papers are concentrated on commercial WVSs, we decided to test and evaluate only free/open source WVSs. After Introduction Section, Section II is devoted to basic architecture of the black box WVSs. In Section III we give brief explanation of used testbed application, used six WVSs with their general characteristics and input vector support, followed by used methodology and obtained results on the false negative rates at the first place. At the end, we give short concluding remarks.

©2013 Faculty of Computer Science and Engineering

221
II. BLACK BOX WEB VULNERABILITY SCANNERS

Conceptually, almost all WVSs consist of three main components: a crawling component, an attacker component, and an analysis component.

At the beginning of the scanning process, the user enters at least one URL, with or without user credentials for the given web application. Using these data, the crawling component identifies all the reachable pages in the application, and all the input points to the application, such as the parameters of GET requests, the input fields of HTML forms, etc. After user sets the scanning profile, scanners can proceed automatically or with user interaction. We used only automated mode for our experiments.

The attacker component analyzes discovered data and for each web form, for each input and for each vulnerability type for which the WVS has test vectors, the attacker module generates values that are likely to trigger a vulnerability. Then, the form content is send to the web server using either a GET or POST request, and appropriate response is obtained from the server via HTTP.

Next, the analysis module has to parse and interpret the server response. Decision if a given attack was successful is made by calculation of confidence value, by using attack-specific response criteria and keywords.

III. EXPERIMENTS AND RESULTS

A. Vulnerable web application

Vulnerable WackoPicko application is a photo sharing and purchasing site. Users of WackoPicko can upload photos, browse other user’s photos, comment on photos, and purchase the rights to a high-quality version of a photo. It has 10 vulnerabilities accessible without authentication (reflected and stored XSS, reflected XSS behind JavaScript, predictable Session ID for admin, weak admin password, reflected SQLi, command line injection, file inclusion, unauthorized file exposure, and parameter manipulation), and 6 vulnerabilities accessible after logging into the web site (multi-step stored XSS, stored SQLi, directory traversal, forceful browsing, logic flaw and reflected XSS behind Flash).

The web server hosting WackoPicko and used in our experiments was run in the OWASP Broken Web Applications Project virtual machine [18], which has numerous intentionally vulnerable applications (we ignore other applications). The following technologies are used: Apache 2.2.14 (Ubuntu), PHP/5.3.2-1ubuntu4.5 with Suhosin-Patch, and MySQL 5.0.67.

B. Tested Web Vulnerabilities Scanners

The scanners were run on a machine with a Pentium (R) Dual Core 2 x 2.00GHz CPU, 4 GB of RAM, and Windows 7 Home Premium.

Table 1 lists the six free/open source WVSs used in our study and their general characteristics. All have graphical user interface and support for proxy mode (manual crawling). Only NetSparker Community Edition and N-Stalker Free 2012 run only on Windows, and other four can be installed on Linux and OS X also. W3Af additionally is available on FreeBSD and Open BCD. Their input vector support are given on Table 2. Many different characteristic comparisons on older versions of these WVSs can be found on Chen’s web site SecToolMarket [2].

Table 1: General characteristics of the evaluated scanners

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>2.5</td>
<td>7.1.1.126</td>
<td>2.0.0</td>
<td>12.4664</td>
<td>0.9.5.0</td>
<td>1.0 (beta)</td>
</tr>
<tr>
<td>License/ Technology</td>
<td>Freeware .Net 3.5</td>
<td>Freeware Unknown (Win32)</td>
<td>ASF2</td>
<td>Java</td>
<td>1.6.x</td>
<td>GPL3</td>
</tr>
<tr>
<td>Operating System</td>
<td>Windows</td>
<td>Windows</td>
<td>Linux</td>
<td>OS X</td>
<td>Windows</td>
<td>Linux</td>
</tr>
<tr>
<td>Authent.</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Report</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Scan Log</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

NetSparker Community Edition have many features disabled, compared to its commercial version, but still you can scan and exploit SQL injection vulnerabilities without any false-positives.

N-Stalker Free 2012 provides a restricted set of features, compared to its commercial version, and will inspect up to 500 pages within target application.

OWASP Zed Attack Proxy (ZAP) is an easy to use integrated scanning and penetration testing tool, and it is designed to be used by people with a wide range of security experience.

Table 2: Supporting input vectors by the evaluated scanners

| HTTP Query String Parameters | Yes | Yes | Yes | Yes | Yes | Yes |
| HTTP Body Parameters | Yes | Yes | Yes | Yes | Yes | Yes |
| HTTP Cookie Parameters | Yes | Yes | Yes | Yes | Yes | Yes |
| HTTP Headers | Yes | Yes | Yes | Yes |
| HTTP/Parameter Name | Yes | Yes | Yes | Yes |
| XML Element | Yes | Yes |
| XML Attributes | Yes | Yes | Yes | Yes |
| XML Tags | Yes | Yes | Yes | Yes |
| HTML Parameters | Yes | Yes | Yes | Yes |
| Flash Action Message Format | Yes | Yes | Yes | Yes |
| Platform Input Vector | Yes | Yes | Yes | Yes | Yes | Yes |

SUMMARY

W3Af stands for Web Application Attack and Audit Framework, it is written in Python, and it was started by Andres Riancho in March 2007. In July 2010, W3Af
announced its sponsorship and partnership with Rapid7. It uses more than 130 plug-ins. Users have available a command-line interface also.

IronWASP stands for Iron Web application Advanced Security testing Platform, created by Lavakumar Kuppan. It uses various external libraries, as IronPython, IronRuby, Json.NET, Jint, etc, making it more powerful. It has a scripting shell for both Python and Ruby giving full access to the IronWASP framework, and this can be used by the pentesters to write their own fuzzers, create custom crafted request, analysis of logs, etc.

Vega includes an automated scanner for quick tests and an intercepting proxy for tactical inspection.

C Methodology

In our experiments, scanners that support authentication, were run without logging and with logging, and only the default values for configuration parameters were used. In the NO_LOG mode, the scanner was directed to the initial page of WackoPicko and told to scan for all vulnerabilities. In the LOG mode, the scanner was given first a valid username and password. We did not use proxy mode for scanners that have support for it. For N-Stalker Free 2012 we start automated mode with OWASP Policy. W3Af is run with activated plugins: audit, auth, bruteforce, grep and mangle.

D. Results

Figure 1 plots the time needed for each scanner to scan used web application. One can see, that running time ranges from 3 minutes to 9 hours and 52 minutes.

<table>
<thead>
<tr>
<th>Reflected SQLI</th>
<th>Stored SQLI</th>
<th>Reflected XSS</th>
<th>Stored XSS</th>
<th>Reflected XSS behind JavaScript</th>
<th>Predictable Session ID</th>
<th>Command line injections</th>
<th>File inclusion</th>
<th>File</th>
<th>Exposure</th>
<th>Parameter Manipulation</th>
<th>Directory</th>
<th>Traversal</th>
<th>Logic Flow</th>
<th>Forceful browsing</th>
<th>Weak passwords</th>
</tr>
</thead>
<tbody>
<tr>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
<td>No_LOG</td>
</tr>
</tbody>
</table>

Table 3: Numbers of vulnerabilities of the evaluated scanners according to their severity without logging

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>High Vulnerabilities</td>
<td>7</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>45</td>
<td>3</td>
</tr>
<tr>
<td>Medium Vulnerabilities</td>
<td>15</td>
<td>4</td>
<td>18</td>
<td>1</td>
<td>78</td>
<td>1</td>
</tr>
<tr>
<td>Low Vulnerabilities</td>
<td>8</td>
<td>16</td>
<td>414</td>
<td>8</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Informational Vulnerabilities</td>
<td>12</td>
<td>21</td>
<td>177</td>
<td>9</td>
<td>2</td>
<td>17</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>42</td>
<td>44</td>
<td>613</td>
<td>11</td>
<td>133</td>
<td>46</td>
</tr>
</tbody>
</table>

Table 5 summarized the numbers of found known vulnerabilities and the number of false negatives. All examined free/open source WVSs have very high rates of false negatives, running from 68.8% for IronWasp to 100% for W3Af. NetSparker can scan only SQLI and XSS vulnerabilities without authentication, so it performed very well, with finding all possible vulnerabilities of these kinds. N-Stalker Free 2012 offer only reduced analysis of XSS vulnerabilities.

The number of found vulnerabilities classified according to their severity is given on Table 3. The total number ranges from 11 to 613 vulnerabilities. High values for founded vulnerabilities do not mean better scanners.

From the evaluated scanners, we find that the report from OWASP ZAP is very confusing, because it mixes vulnerabilities with different severity.

At the start, we know that three scanners NetSparker Community Edition, IronWasp and Vega, do not support authentication, so they could not find any of the vulnerabilities accessible after authentication.

Table 4 summarized obtained results. An empty cell indicates that the given scanner did not discover the vulnerability. NO_LOG means that the given vulnerability was found without authentication. One can see from the obtained results, that for WVSs that support authentication with scanning, the scanners did not find additional vulnerabilities. Also, W3Af for example, did not find any of the known vulnerabilities.
vulnerabilities, with or without authentication, so it found only two out of five XSS vulnerabilities. Other modules are disabled in this version.

Table 5: Number of false negative

<table>
<thead>
<tr>
<th></th>
<th>Number found vulnerabilities</th>
<th>Number of false negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetSparker Community Edition</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>N-Stalker Free 2012</td>
<td>2</td>
<td>14</td>
</tr>
<tr>
<td>OWASP ZAP</td>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>W3M</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>Iron WASP</td>
<td>5</td>
<td>11</td>
</tr>
<tr>
<td>Vega</td>
<td>3</td>
<td>13</td>
</tr>
</tbody>
</table>

IV. CONCLUSIONS

Because the web application WackoPicko is almost three years old, and has only 16 known vulnerabilities, and because it is only one of its type, there is a need of a new application with more recent vulnerabilities, with versions other than Apache/PHP/MySQL also. Also, OWASP Broken Web Applications Project need to be updated with the latest versions of used technologies, because Apache 2.2.14 (Ubuntu) and PHP/5.3.2-lubuntu4.5 with Suhosin-Patch have known vulnerabilities and exploits, which have been detected by WVSs, and have made our tasks harder. Because of this, we did not gave the false positive rates.

REFERENCES


