УНИВЕРЗИТЕТ „ГОЦЕ ДЕЛЧЕВ” - ШТИП
ФАКУЛТЕТ ЗА ИНФОРМАТИКА

ISSN 1857- 8691

ГОДИШЕН ЗБОРНИК 2012

YEARBOOK 2012

ГОДИНА 1 VOLUME I

GOCE DELCEV UNIVERSITY - STIP
FACULTY OF COMPUTER SCIENCE

УНИВЕРЗИТЕТ „ГОЦЕ ДЕЛЧЕВ“ – ШТИП
ФАКУЛТЕТ ЗА ИНФОРМАТИКА

ГОДИШЕН ЗБОРНИК
2012
YEARBOOK
2012

ГОДИНА 1 МАРТ, 2013 VOLUME I

GOCE DELCEV UNIVERSITY – STIP
FACULTY OF COMPUTER SCIENCE

ГОДИШЕН ЗБОРНИК
ФАКУЛТЕТ ЗА ИНФОРМАТИКА
YEARBOOK
FACULTY OF COMPUTER SCIENCE

За издавачот:
Проф д-р Владо Гичев

Редакција и администрација
Универзитет „Гоце Делчев“-Штип
Факултет за информатика
ул. „Крсте Мисирков“ 10-А
п. фах 201, 2000 Штип
Р. Македонија

Address of the editorial office
Goce Delcev University – Stip
Faculty of Computer Science
Krste Misirkov 10-A
PO box 201, 2000 Štip,
R. of Macedonia

СОДРЖИНА
CONTENT

IMPROVING THE SECURITY OF CLOUD-BASED ERP SYSTEMS

Gjorgji Gicev[1], Ivana Atanasova[2] and Jovan Pehcevski[3]

[1]Artisoft, Skopje, Macedonia, george.gicev@artisoft.net
[2]Artisoft, Skopje, Macedonia, ivana.atanasova@artisoft.net
[3]Faculty of Informatics, EURM, Skopje, Macedonia, jovan.pehcevski@eurm.edu.mk

Abstract: Enterprise resource planning (ERP) systems integrate internal and external management information across an entire organization, embracing finance/accounting, manufacturing, sales and service, customer relationship management, etc. ERP systems automate this activity with an integrated software application.The architecture of the software facilitates transparent integration of modules, providing consistent flow of information between all functions within the enterprise. ERP popularity has rapidly increased in the last few years and they are starting to be used by all types of businesses. In this regard, the ERP is becoming a system with high vulnerability and confidentiality in which security is critical for the system to operate. Recent studies show that many ERP vendors have already integrated some kind of security solutions, which may work well internally. However, in an open environment, one needs more advanced and innovative technological approaches to secure an ERP system.

In this paper, we evaluate how and to what extent one can improve the security of an ERP system by implementing a set of security measures addressing the current top security threats. The paper evaluates the effects and provides conclusions on the applied security measures using ArtAIIS – Artisoft's cloud-based, web-enabled software-as-a-service ERP system.

Keywords: Security measures, cloud ERP, web security, security framework.

11 Introduction

Information technology has been making huge impact on the civilization especially in the last two decades. Tremendous achievements in the micro technology made computers extremely fast and cheap, which triggered rapid development in all segments of everyday life. World Wide Web Technology combined with affordable workstations and fast networks caused explosion of networked people, institutions and businesses. As never before, we are witnessing a true globalization of the world in all segments. We do not have information flow anymore. Information is everywhere instantly from the moment it originates. But as always, good things also come with drawbacks. Leaving everything on the net allows for someone to harm or take opportunity to hack, steal or destroy the information or the system. Accordingly, we continually build security blocks, measures, procedures, and protocols in order to defend our systems from external and also internal attacks.
Today, we can say that Internet addiction leads humanity to a level where a small break in any of these services is having serious impact on our daily work as would for example electricity brake down. Civilization is networked as never before. We became de facto global village. But this networking is bringing huge possibilities for everybody, including those who want to harm the positive trends. The bigger the network is, the bigger the threat is, and the bigger the security vulnerability is. This is why implementation of security measures on application software, in our case web application software is an interesting area and topic for discussion.

Recent studies show huge, even extremely, increased number of users of Internet services. According to Internet World Stats, on December 31 2011, the number of Internet users was over 2 billion
.

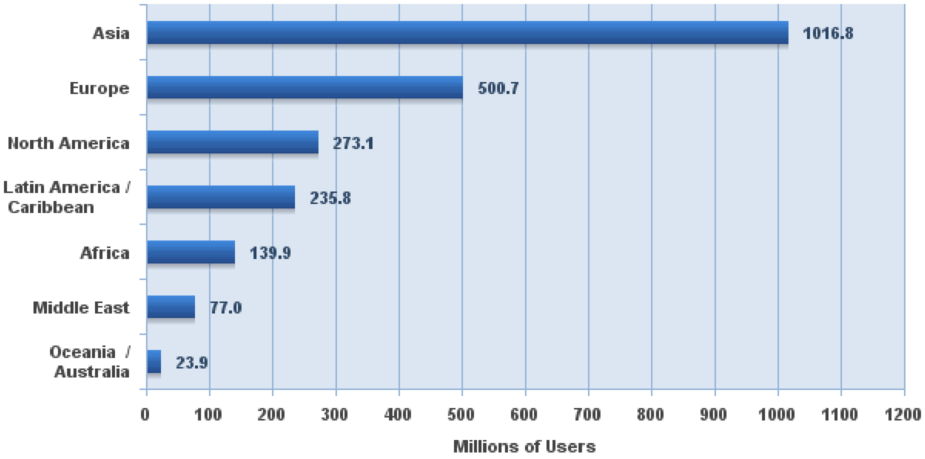

Fig.1.1 Analysis of the number of Internet users (web sites and web applications) according to Internet World Stats, 2011

Because of the technology globalization, ERP systems have started to appear as a necessity not solely for large companies but also for small and medium businesses [1]. This sent ERP system utilization to the sky. However, according to Dhilloni [2], information security in the ERP solutions has traditionally been an afterthought. Because of businesses' increased dependence on information, security is being considered proactively. On the other hand, the process of security implementation in ERP system is a long and everlasting process. The great popularity of integrated ERP system solutions, today more than ever, lead to the need of a solid security framework in order to set the base minimum when it comes to ERP security. This framework should only be used as a starting point in the process of securing and fortifying an existing ERP system.

In this paper we evaluate how and to what extent one can improve the security of an ERP system by implementing a set of security measures addressing the top current web application threats in the world.

12 Existing security frameworks to ERP systems

Security in ERP solutions requires grasping a wider approach and concept where security measures will involve people, network, host and application security [6]. In this regard, we have analyzed existing security frameworks which include all aspects of security in an ERP system.

We note, however, that both, the Security framework for an ERP system [4] and the Security framework for ERP in cloud [3] have not gone into much detail as far as the technology component is concerned.

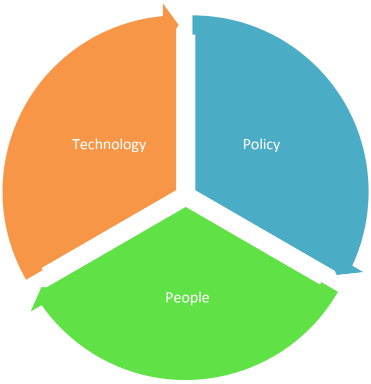

Fig. 2.1 Components of the Security framework for ERP system[1]

According to Marnewick [4], the weakest link in an ERP system is still its users, as open gateways to the information stored and obtained through the system. User behavior is still unpredictable and it represents an issue that must be addressed in the stages before starting to use the ERP system. The most important stages are:

- Policies and procedures – for explicit control and management of user behavior;
- Risk analysis – to increase the level of security to the assets of greatest value to the business; and
- Awareness – users need to become aware of the security threats and risks of their behavior.

The technology component also involves the following elements [4]:

- Identification & Authentication – ensuring that the system is accessed by legitimate, authorized users;
- Authorization – restriction of the access rights and actions of the user in the system;
- Confidentiality – only authorized users can see and use specific data;
- Integrity – only authorized users can modify specific data;
- Non – repudiation - a transaction which took place must have a continuity information in the system in order to undoubtedly prove its existence and the user who initiated it;
- Availability – the system must be available 24/7 for business continuity; and
- Auditing –requires a risk-based systems review supported by detailed checklists and practical experience in designing controls.

13 Improving the cloud-based ERP security

The main goal of this paper is to contribute to the technology component of the existing security frameworks to ERP systems. In this section, we propose, implement and evaluate a set of security measures for an ERP system in the cloud [3]. We use the ArtAIIS ERP system – a product of the company Artisoft - which is web-enabled and offered as a service (SaaS) on a cloud platform.

3.1 Security threats

When considering the technology component of the system, it is necessary to include and analyze different types of threats and attacks which are current and recurrent.

According to the list of Top 10 threats of web application software OWASP [5], the number one threat is the *SQL Injection attack*, immediately followed by the *XSS attack*. In this paper, we focus only on the top 5 threats of web application software, as from our experience these are identified as the most occurring threats in the ERP industry. They are described as follows.

1. *XSS scripting* - represents a type of injection problem, where harmful script is inserted into seemingly reliable website. XSS attack occurs when an attacker uses a web application to send malicious code, usually in the form of a browser - script to another user of the application.
2. *SQL injection* - SQL injection attack exploits weaknesses in the validation of input parameters in the web application to execute commands and activities in the database.
3. *Cookie replay* - takes user authentication cookie through software to monitor traffic and performs its replay for obtaining access to the application under a false identity.
4. *Session replay* - With special software to scan and monitor network traffic, the attacker can intercept the user’s session token and it can be used to pass authentication.
5. *Cookie, query and form field manipulation* - Attacker can easily perform manipulation and modification of the query string parameters that are passed via HTTP GET from the client to the server as they are visible in the URL. There are numerous tools that allow the attacker to modify a cookie that is stored in memory. This type of attack is performed in order to gain access to a particular application or web site. Values of HTML form fields are sent in clear format via HTTP POST Protocol. This applies to

visible and hidden form fields. These fields can be modified very easily and they can skip the validation procedures.

3.2 Security measures

The proposed security measures we consider [7, 8, 9] are shown in Table 1.

Table1. Proposed security measures for extending the technology component of an existing security framework for ERP system [7, 8, 9].

Threat	**Measures**
XSS Scripting	• Full validation of user input parameters.The application must verify that the input query strings, form fields, and cookies are valid for the application. The approach that works here is to treat all input parameters as harmful. • Usage of HTMLEncode() or URLEncode() functions. In this way, the code that performs harmful scripts is transformed into harmless HTML.
SQL Injection	• Full validation of user input parameters.The application must verify that the input query strings, form fields, and cookies are valid for the application. The approach that works here is to treat all input parameters as harmful. • Procedure and query parameterization. Provides the value of parameters that are not treated as executive code. In this way, the incoming user parameters cannot contain other types of data than those defined for the type parameter. • Least privilege rule. To access the database to be used user orders with minimal privileges to perform the required actions on the data.
Cookie replay	• SSL. Use SSL encrypted communication channel each time when an authentication cookie is transmitted. • Cookie timeout. Use cookie timeout value to force re-authentication after expiration of the specified time interval. This measure does not

	prevent replay attacks but shortens the time frame in which such an attack can be carried out.
Session replay	• Re-authentication. Re-authentication before performing a critical operation is sufficient protection from this attack. • Session timeout. Session expiration process should be fully respected, including all cookies and tokens. • Remember me. Creating the option "Remember me" in order to allow the user not to store any information for the client session on his computer.
Cookie manipulation	• Cookie encryption or protection using HMAC
Query manipulation	• Session identifier. Use a session identifier to identify the client and keep the session data in the server state warehouse. • HTTP POST. Select HTTP POST instead of GET for submission of forms • Encryption. Encryption of query string parameters
Form field manipulation	• Session identifier. Instead of using hidden fields, those values should be stored in session identifiers that are safely protected in the server's state warehouse.

3.3 Experimental methodology

The validity of the proposed security measures, described in subsection 3.2, was evaluated in two phases:

- Phase I - the ERP solution is tested in the current state to detect existing weaknesses and vulnerabilities; and
- Phase II - the ERP solution is tested with the same security tools for vulnerabilities after the proposed security measures have been applied.

The analysis is made using the following security tools:

- NetSparker – powerful scanner of vulnerabilities of web applications, which performs crawling, attacks and detection of vulnerabilities, regardless of platform or operating system. NetSparker is the only tool that allows false-positive testing with a built-in exploitation engine which confirms the detected weaknesses.

- Acunetix- web application auto scanner more like a black-box tool. Acunetix provides full scanner, crawler and search reports, as well as huge database security checks for all server platforms

- WebCruiser –effective and powerful tool that allows detailed analysis of websites and web applications. The tool offers a weakness scanner as well as numerous safety tests. WebCruiser is an automated SQL Injection, XPath Injection and XSS tool. In our paper Web Cruiser is used only for testing high level threats

According to the level of threat, the application tools detected weaknesses into three categories: High, Medium and Low level threat.
The ERP solution was evaluated in its current state, as offered on the SaaS platform of Artisoft. No additional protection measures existed aside from the initial security measures implemented when the solution was built.

3.4 Results and discussion

In Phase I the following top 3 results were obtained (shown in Table 2).

Table2. Top 3 detected weaknesses and threats by the security tools used to scan ArtAIIS ERP system in its current state

Application	Weaknesses
NetSparker	High: password is transmitted in its original form via HTTP Medium: autocomplete is enabled for sensitive data Medium: cookie not marked as HttpOnly
Acunetix	Medium: server admin page publicly available

WebCruiser	Medium: cookie not marked as HttpOnly Medium: unprotected username and password transmission High: six POST SQL INJECTION attacks and one XPath INJECTION.

The security measures from Table 1 were applied to the ERP solution ArtAIIS. Phase II consisted of reassessment of the ERP solution again after applying the security measures from Table 1. The summarized results are shown in Fig. 3.1

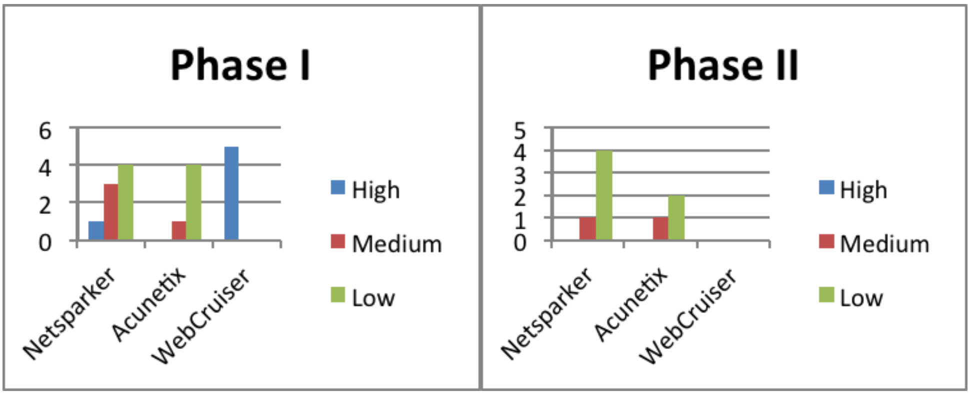

Fig. 3.1 Results from phase I and phase II analyses

5 Conclusions and future work

Results of our analysis show that, when even a small set of standard security measures are applied to a sample ERP software solution, one can greatly contribute to the security of the system. The results demonstrate that after the implementation of the set of security measures (shown in Table 1) the security scanner applications did not detect any high level threats and decreased the number of medium level threats. This substantially improved the overall security of the ERP system.

However, we must note that there is no recipe to complete security when it comes to ERP systems. The weakest element is still the people component where no security measures can guarantee the confidentiality level of the people of the company, regardless of the awareness and procedures that are put in place. As mentioned previously, security is an ongoing process where the security measures have to be tuned every time the system is updated. What matters even more, of course, is the price/performance ratio, meaning that security measures will always try to first comply with a higher performance ratio and then to advantage the security of the system.

In the future, this small set of security measures could be extended and reevaluated with different security scanner applications to confirm the effectiveness of the same in the ERP system architecture. There always should be balance between the price/performance ratio and the security level applied to the ERP system, since usability is directly connected to the performance of the system. In this way one would define a baseline set of security measures which will successfully increase the security of the ERP system while maintaining at the same time a satisfying level of performance. Nevertheless, critical and sensitive modules should always get a security base line measures applied regardless of the effect they have on the system performance, as data security and protection is the foundation of today's ERP web-based systems.

References

Book chapter:

[1] M. A. Rashid, L. Hossain and J. D. Patrick (2002): *Evolution of ERP Systems: A Historical Perspective, Chapter 01,* DOI: 10.4018/978-1-931777-06-3.ch001, IGI Global

Journal papers:

[2] G. Dhillion (2004): Guest Editorial: the challenge of managing information security. International Journal of Information Management. Volume 24. pp 3 – 4

[3] G. Fathima Haseen Raihana (2012): Cloud- ERP: A Solution model. IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 2, No. 1

Proceedings:

[4] C. Marnewick and L. Labuschagne, (2006): *A Security framework for ERP Systems*, Academy for Information Technology, University of Johannesburg.

Web pages:

[5] OWASP, (2010): *The ten most critical Web application security risks (Top 10)*. The Open Web Application Security Project. Accessed on 18.04.2012

[6] C. Herberger – SCMagazine (2010): *Defense in depth: building a holistic security infrastructure,* Accessed on: Februray 2012, (http://www.scmagazine.com/defense-in-depth-building-a-holistic-security-infrastructure/article/190025/

[7] Microsoft (2005): *Prevent Cross-site scripting in ASP.NET,* Accessed on: February 2012. http://msdn.microsoft.com/en-us/library/ff649310.aspx

[8] Microsoft (2005): *Protect from SQL Injection in ASP.NET*, Accessed on: February 2012. http://msdn.microsoft.com/en-us/library/ms998271.aspx

[9] C-SharpCorner (2004): *How to secure your Web Applications*, Accessed on: February 2012. http://www.c-sharpcorner.com/UploadFile/krishvr/securewebapp11262005011914AM/securewebapp.aspx