THE USV ANNALS OF ECONOMICS AND PUBLIC ADMINISTRATION | VOLUME 21, ISSUE 1(33), 2021

A COMPARATIVE STUDY ON THE SECURITY OF WEB CONTENT MANAGEMENT SYSTEMS

Vlad TOMA
"Ștefan cel Mare" University of Suceava, Romania
vlad.toma@usm.ro

Received 29 March 2021; Accepted 2 June 2021

Abstract:
The present paper aims to compare and analyze the security of the most common three open-source Web Content Management Systems: WordPress, Joomla! and Drupal. The paper is focused on describing the main vulnerabilities regarding the security of CMS platforms and reflects relevant data that can be used to avoid the most common attacks. In the digital era, where all the information is stored on web servers to be accessed from everyone, security and confidential data protection is an important part when it comes to choosing the right CMS for our needs.

Key words: CMS, Security, WordPress, Joomla, Drupal

JEL classification: L86

1. INTRODUCTION

Without doubt, we live in a world where the web, the internet, is more alive than ever. The total number of websites has already passed 1 billion [1] (1,197,982,359 in January 2021), even if the number is continuously changing, with a number of users reaching almost 5 billion and a penetration rate of 61,2% [2]. New internet satellite technologies are aiming to increase this percentage by enabling wireless signal in remote locations where typical services are not available. But, by observing the growing trend of this numbers in the last couple of years, we can say that the evolution of this data has reached a linear trajectory instead of an exponential one, with social media websites taking over the presentation and the photo gallery part of a typical website. However, the quality and functionality of this services are not comparable with dedicated and professional solutions for companies or businesses, with 60% of the small and medium businesses not having a corporate website [3]. In this troubled time of the new Coronavirus, this lack of web presence has affected many merchants, forcing them to move their products online. This means that is enough space for developing and research in this area, and this is the point where Content Management Systems (CMS) platforms come in handy, by facilitating online presence.

CMS platforms are software tools used whenever are needed to develop a website, especially when multiple users must work together on the same site, by assigning them different roles and permissions and is often combined with the lack of specialists in the field. These platforms are very useful especially for beginners in web development or for different businesses, taking in consideration the relative low costs for the early stage of development. A good example would be an online shop where the initial costs are related to hosting services and domain purchasing rather than the platform itself. The users can easily set up a store and manage it from a user friendly interface, depending on the privileges assigned to the account. In this scenario, CMS platforms have a great potential. The majority of them are based on PHP (Hypertext Preprocessor) server-side scripting language.

2. CONTENT MANAGEMENT SYSTEMS - OPEN-SOURCE SOLUTIONS

With the possibility to choose between open-source (free) or proprietary (paid) software, it can be difficult for beginners to pick up the right one. The paid software focuses on providing

premium functions regarding flexibility, SEO optimization, CRM (Customer Relationship Management), payment methods, marketing services, security and technical support. No doubt, this kind of software comes with obvious advantages, being more suitable for large companies and organizations in need of specific solutions. The present paper is focused on open-source software, providing powerful solutions suitable for a large audience with small and medium businesses at zero costs, excepting those related to hosting and domain acquisition.

2.1 WORDPRESS

With approximately 18 million installs and a market share of 64,2% [4] from all websites that use a content management system, WordPress is by far the most common open-source CMS software in the world. Being initially designed as a tool to create and manage blogs, it met a popularity which allowed the development of a very powerful and functional system. The large number of users which supports the online community brings up many benefits in terms on plugins (third party software) development, themes and functions, but makes WordPress very vulnerable to hackers due to the lack of maintenance and security expertise of the users. In this matter, updates are necessary which can lead to incompatibility with the installed extensions and plugins. Security problems are detailed in a dedicated chapter.

2.2 JOOMLA!

With a market share around 3,4% [4] from all sites than run a content management system, Joomla! seems to have lost the battle regarding the number of users. In fact, it occupies the second place in top open-source CMS platforms, targeting both beginners and specialists. It looks harder to use compared to it's number one competitor, but comprehensive functions are included without the need of extensions. For non-experts, the documentation is pretty solid, consisting in an online community. Joomla! Is in particular very popular in USA, being defined as a completely object oriented software, based on a **MVC** (Model-View-Controller) stand-alone. This enables users to design their own extensions, which can be installed from the back-end. Extensions are divided into plugins, component and modules, which can be used for the back-end or the front-end side.

2.3 DRUPAL

Drupal it's the third most used CMS platform, being the first one of its kind to be released in 2001. It covers 2,4% [4] of all website that are based on a content management system, focused on delivering services with high, reliable performance, high security and personalization. The online community passes 1 million, with dedicated developers, designers, editors and sponsors from all over the world. Moreover, they organize work meetings in different places for interested users to rise their level of expertise. Drupal offers extensive opportunities for personalization, based on its modular structure, being suitable for a variety of projects. The large specter of extensions makes possible the implementation of a complex multi-domain structure for company portals. If the base functions are not enough, extensions can be installed via FTP (File Transfer Protocol).

3. COMPARATIVE STUDY

Before comparing the three CMS platforms in our study and highlighting the differences, advantages and disadvantages, we should mention that there are similarities as well. From the incubation period point of view, all three were founded between 2000 and 2005 [5]. This means that these platforms have been around for about 16-21 years, with plenty of time to reach a software maturity which allowed them to survive and prosper on a continuously developing market, with constant threats from new technologies and solutions.

First of all, all of them are open-source, meaning that can be used under common license, being developed in a collaborative way. This means that are efficient from the costs point of view, and due to the large online communities which work and develop the software, the fallowing benefits are assured:

- new capabilities and concept are quick introduced;
- common software problems are dealt in due time;
- the matter of security and stability problems are verified on a large scale;

Secondly, all three CMS platforms are powered by PHP based engines combined with MySQL (Structured Query Languages) databases.

3.1 VULNERABILITIES

Regarding the security, with great popularity comes great responsibility. Due to the large number of active installations and the lack of expertise from the users, these CMS platforms are often targeted by hackers. Security is very important thing, especially for the kind of websites that stores user data, such as account details and passwords, and for e-commerce solutions as well. Among the possible threats we can mention [6,7]:

- **Data manipulation**: affecting the integrity of data. For example, SQL injection and parameter manipulation;
- **Confidential data breach**: unauthorized person gains access at sorted data. For example, SQL injection and Cross-Site Scripting-XSS (a form of injection).
- **Phishing**: a way to collect confidential data using forms and spam e-mails on a clone site that looks the same as the original.
- **Spam**: a malicious way to use the e-mail address published on the site;
- **Execution of code**, running scripts or programs stored on a server exploiting vulnerabilities.

By analyzing the data from figure 1 [8], JavaScript and PHP are the programming languages with the most vulnerabilities, with a high amount of threats. Java environments and technologies are more secure, the common breaches pointing poor coding from the developer. Being the most common web server, Apache [9] is not missing the attentions of hackers, exploiting different loopholes in code design.

At the application level, 61% of the attacks are made through a browser and 86% of this type are attributed to a **XSS attack**, forming a majority. In this kind of security breach, malicious commands are injected in a website executed on the clients-side browser, without them knowing, granting access to important data, such as tokens and cookies which are stored in the browser [10].

Another common type of attack is **SQL injection** [11]. In this case, an unauthorized user is accessing the website's database, with de possibility to alter it. Databases are fundamental in creating and using a CMS platform. All the data and the user info are stored in databases. To allow this kind of attack, the website must have a method to insert a SQL interrogation, being executed against the server. If the attack is successful, confidential data are exposed and can be accessed, read, modified or even deleted. Moreover, the hacker can execute operations that need administrator rights on the database, obtaining information about its structure. **While SQL injection** attacks are focused on obtaining data, XSS based ones are exploiting the output data provided by the browser. In either case, serious attacks can compromise the entire system.

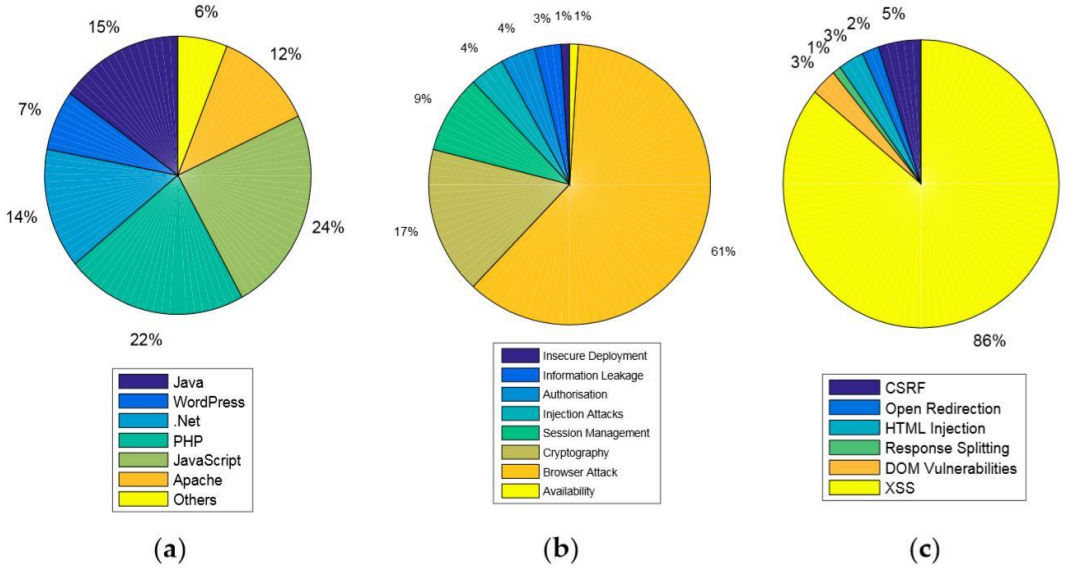

Figure 1. Vulnerabilities
Source: Vulnerability Statistics Report; EdgescanTM Portal: Dublin, Ireland, 2016.

(a) – probability to discover vulnerabilities in the language framework
(b) – probability to find vulnerabilities at the application level
(c) – Distribution of browser-based attacks

3.2 INFECTION RATE DISTRIBUTION

Regarding the infection rate among the three CMS platforms in our analysis, it seems at first sight that WordPress doesn't exceed at security. According to a company specialized in web security (Sucuri.net), 94% of all infected platforms are running WordPress, as shown in figure 2.

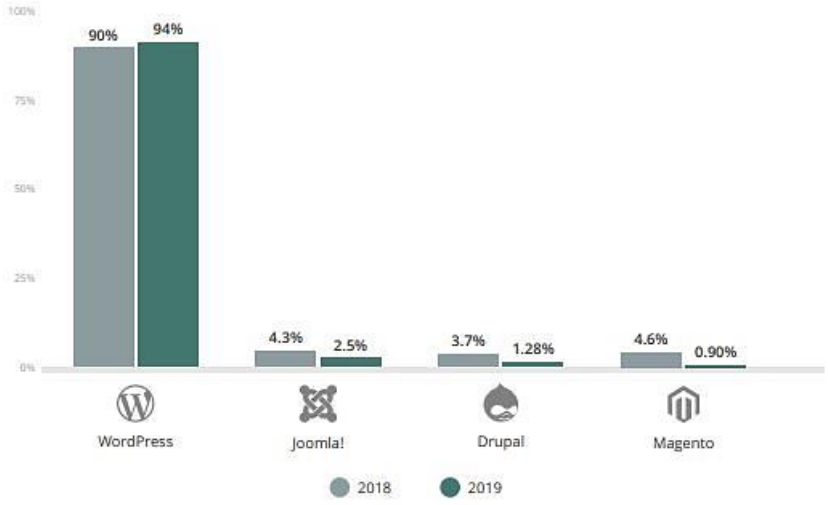

Figure 2. CMS infection rate (2018/2019)
Source: https://sucuri.net/

The high infection rate of WordPress sites is affected by multiple factors. One could be its popularity and high number of installs, taking 64,2%**Error! Bookmark not defined.Error! Bo okmark not defined.** of the market share. The exploiting of the weak points is facilitated by this large numbers when scripts are used to scan specific known vulnerabilities in the network. It's nearly impossible to maintain a completely secure platform, taking in considerations it is made for regular users that are often committing mistakes and neglecting updates of the core functions and

plugins. For example, a known vulnerability was the inappropriate use of the **WordPress update_option() function** and other design flaws.

3.3 VULNERABILITIES BASED ON OUTDATED SOFTWARE

In 2019, around 56% of all CMS applications were running outdated core and components, as shown in figure 3.

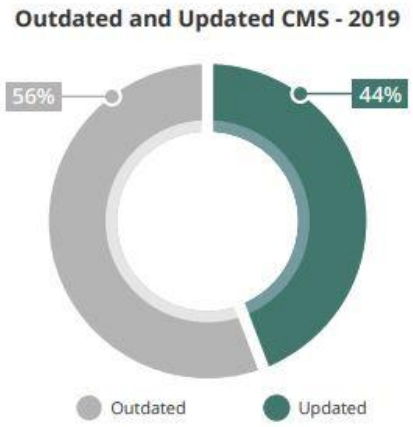

Figure 3. CMS Infection rate due to outdated components.
Source: https://sucuri.net/

A more detailed look on the data from figure 4 brings up the fact that once the automatic backround updates were introduced starting with version 3.7, WordPress holds an advantage over platforms without this function. As a result, "only" 49% of all WP installations were running **outdated core versions** and functions at the moment of infectation, a considerable smaller percentage in comparison with Drupal (77%) and Joomla! (90%).

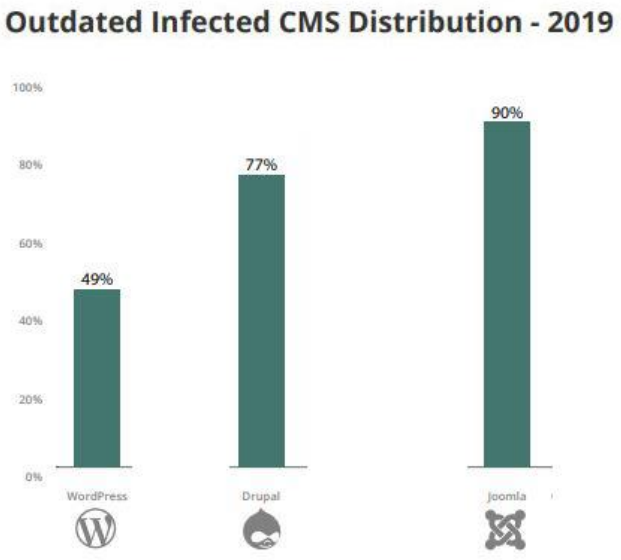

Figure 4. Outdated infected CMS distribution
Source: own elaboration using data from https://sucuri.net

4. CONCLUSIONS

Both Joomla! and Drupal are relying on major releases with big importance, which can result in a more complex update process. This translates into the difficuly of users to maintain the

software up to date, with the tendancy to neglect or postpone the available updates, as shown in figure 3. Even so, when reported to the number of active installations, **Drupal** offers the best security oriented software, with many members working at this department, dedicatede to this scope. In fact, security is the main strong point which determined certain users to choose Drupal, such as big companies and guvernamental institutions. **Joomla!** seems to encounter difficulties when it comes to security, maybe due to the relatively few members working at this department. Considering the large number of users and their different level of expertise, it is nearly impossible to create a completly secured environment when it comes to WordPress. However, it can be kept safe by updating it at the right time and maintaing a secure workspace by installing plugins with caution and updating them as well. I would say that the security of any website depends on the human resource, not only Content Management Systems.

Most of the time, the security of a CMS platform rely on its maintainance, by applying updates both on the server-side software – Apache, PHP – and the client-side – CMS version, plugins.

REFERENCES

[1] https://www.netcraft.com/ - accessed on January 2021
[2] https://www.internetworldstats.com/ - accessed on January 2021
[3] How Very Small Businesses Are Utilizing the Internet Today and Future Future Expectations; GoDaddy LLC & Redshift: Scottsdale, AZ, USA, 2015.
[4] https://w3techs.com/technologies/history_overview/content_management - accessed on March 2021
[5] https://www.wikipedia.org/
[6] Newman, R.C. Cybercrime, Identity Theft, and Fraud. In Proceedings of the 3rd Annual Conference on Information Security Curriculum Development, Kennesaw, Georgia, 22–23 September 2006; ACM Press: New York, NY, USA, 2006; p. 68.
[7] Tanenbaum, A.S.; van Steen, M. Distributed Systems: Principles and Paradigms; Prentice-Hall: Upper Saddle River, NJ, USA, 2002.
[8] 42. 2016 Vulnerability Statistics Report; EdgescanTM Portal: Dublin, Ireland, 2016.
[9] Apache. online: https://www.apache.org/
[10] Yusof, I.; Pathan, A.-S.K. Mitigating Cross-Site Scripting Attacks with a Content Security Policy. Computer 2016, 49, 56–63.
[11] SQL Injection—OWASP. Available online: https://www.owasp.org/index.php/SQL_Injection