IBM MSS

THE RISKS OF CONTENT MANAGEMENT SYSTEMS

RESEARCH AND INTELLIGENCE REPORT

RELEASE DATE: FEBRUARY 26, 2015

BY: DAVID MCMILLEN, SENIOR THREAT RESEACHER

TABLE OF CONTENTS

EXECUTIVE OVERVIEW/KEY FINDINGS

In today's fast-paced business world, the need for quick changes to web content is greater than ever. In the past, this task was relegated to web masters and coders who created HTML code, JavaScript modules, and plugins on the fly. There was one fundamental problem with that arrangement; it left a most important task up to a handful of people. Web content needs to be dynamic, especially for retailers as their products change daily and, in some cases, hourly. In order to decentralize the tasks of web content management, a new method was introduced and dubbed Content Management Systems, or CMS.

Now, let's take a look at the security risks of CSM systems. Due to the ever growing need for quick changes to web content, more businesses are leveraging CMS systems. Some of the more common CMS platforms in wide use today are WordPress, Joomla and Drupal. In fact, these three together represent over 75% of all CMS platforms in use today. These CMS platforms come standard with many cheap web hosting companies via their CPANEL functions. The fact that they come as a standard offering proves their effectiveness and popularity. What is quite concerning, however, is that a WP White Security Study found 73% of all WordPress installations had known unpatched vulnerabilities that could be easily detected with a freeware vulnerability scanner. An argument can be made that 73% is overinflated as the WP study focused on a limited subset of WordPress installations. Regardless of the debate on numbers, cybercriminals know that there are large amounts of unpatched installations and, as a result, focus heavily on CMS.

REFLECTING ON CMS

The history of CMS begins in the late 1990's with the first of three stages of CMS development. Some of those early CMS platforms were Roxen, Blitzen, Ingeniux, and Vignette. They all offered a very structured development environment utilizing templates, but lacked a true WYSIWYG (what you see is what you get) component. Most of these early CMS platforms were developed by web design agencies rather than software developers. Following the dot-com crash, the majority of these systems were put out to pasture as most of the design agencies moved out of the coding business and focused more on design.

The next phase of CMS development came in the 2000's and was primarily led by software companies who thought out new ideas and began to build the foundation for the future of CMS systems. They built in features such as WYSIWYG, search capabilities, podcasts and survey tools. They even improved the HTML language. The leading CMS companies during this phase were DotNetNuke, Mambo, and RedDot who later joined together and created the Joomla CMS system. The Open Source movement got its start during this phase due in large part to the high costs of enterprise level software. This created a split

between paid and free software which today still exists. With the open source market booming, CMS became feature rich and the demand for CMS increased. Web agencies now could use both the coders and the designers to build and sell templates to consumers, allowing them to massage the look and feel of their web sites without having to know how to code.

The world is now living currently in what is the third phase of CMS development. This new phase concentrates on the recently well-publicized hacks on open source platforms. The argument of using open source software for commercial usage in an enterprise environment has its roots here. The omnipresence of open source software has not gone unnoticed by the hacker community at large. Hackers are always hard at work attempting to earn more badges of skill by defacing web sites and embedding malware into ecommerce sites in order to harvest credit card information.

Some of the key features of third generation CMSs are modular add-ons which require minimal coding for integration, the ability to be run as a hosted application, being able to be sold by design agencies and affiliates (templatemonster.com e.g.) and integration into databases, ecommerce and email as modules instead of plugins. Where back-end and server-side code used to be a requirement, CMS platformscan now run front-end client-side code.

WHY ARE CMS DEPLOYMENTS VULNERABLE

CMS platforms are highly prized targets by hackers. It would be easy to assume that the big three, WordPress, Joomla and Drupal must be security hardened to a great degree out of the box and that the platform developers would have ensured a high degree of security controls into their products. The reverse is true, however. These products are built on open source frameworks within shared developer environments just like Linux, Apache and Open Office. Since these three CSM platforms are so popular, and with widely publicized vulnerabilities built within them (mainly third-party themes and plugins designed by thousands of different authors), they are a prized target of both security researchers as well as hackers.

SECURITY CONCERNS WITH CMS

Vulnerabilities within CMS platforms are a literal gold mine for hackers allowing them an efficient way of executing mass-scale attacks in an automated fashion. Let’s take a look at some security problems that negatively affect CMS.

BRUTE FORCE

Web site operators who use weak passwords leave their Administrator accounts vulnerable to brute force attacks. Obtaining access to an admin account can lead to injection of malware into the web site which could allow them to be turned into Distributed Denial of Service (DDoS) bots, as an example. Obtaining admin level access also allows a hacker to deface or disable a company's web site and distribute malware which could lead to blacklisting on Google and other search engines.

THEMES AND PLUGINS

There are thousands of developers who design CMS themes and plugins for custom use. Because of the diversity within the development community, no guarantee can be made that any or all of these components are not vulnerable. Once again, this makes them a popular target of hackers. It was found that 20% of the 50 most popular WordPress plugins were vulnerable. Of those 20%, eight million were downloaded from WordPress last year. The average CMS deployment uses four plugins at a minimum.

SQL INJECTION AND CROSS SITE SCRIPTING

There are hundreds of known SQL Injection and cross-site scripting(XSS) attack parameters available in a simple Google search that affect CMS platforms, specifically within the PHP environment - too many to include in this report. New and improved attack strings are widely reported on a daily basis on many underground hacking sites. The SQL Injection attack vector remains on the OWASP list of top ten web vulnerabilities for years and that position at number one is not expected to change.

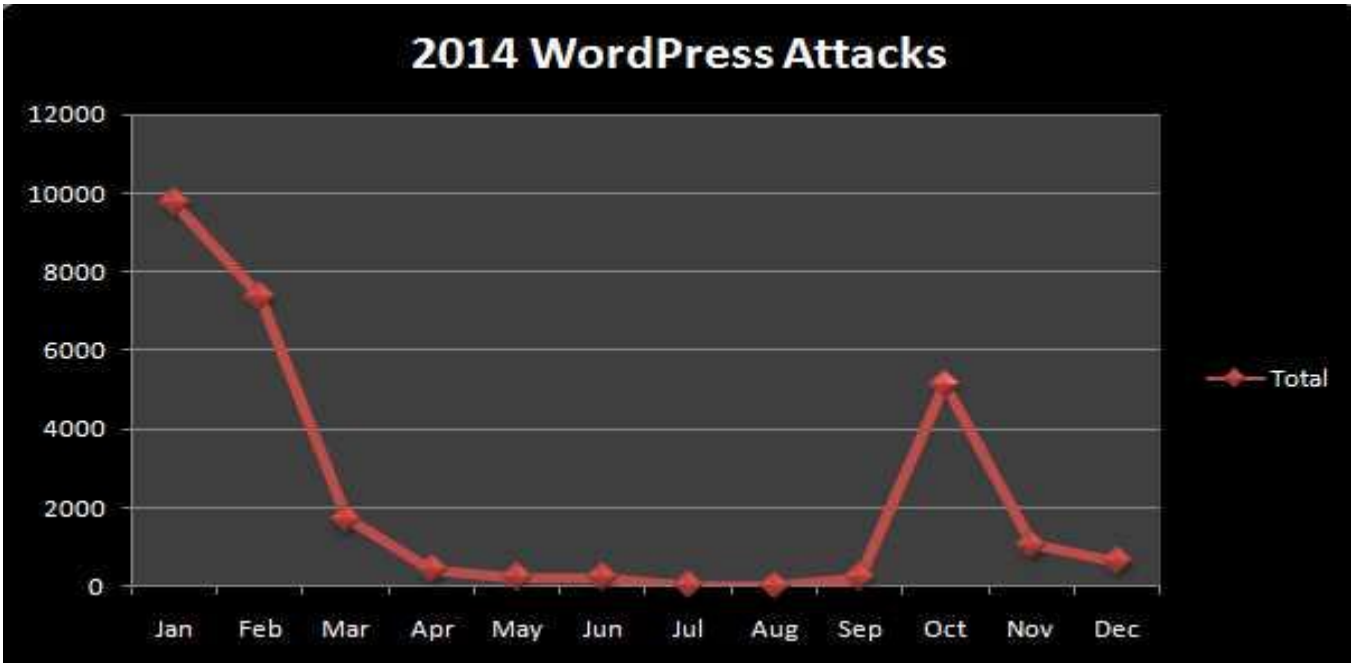
DDOS

Security researchers from Securi uncovered a simple trick where attackers simply sent a pingback request to the XML-RPC file within WordPress. A large DDoS attack in 2014 utilized this technique. Over 162,000 WordPress sites were leveraged creating a super DDoS net that focused on one website and took it down. Using this tactic, hackers are able to greatly amplify the bandwidth at its disposal. XML-RPC is a protocol used by WordPress and other CMS platforms and applications in order to provide services such as pingbacks, trackbacks and remote access to users. In this scenario, one single attacker can use thousands of WordPress sites to perform their DDoS attack while remaining hidden. What makes this DDoS attack type even more effective is that XML-RPC is directed at layer 7 (application layer) which handles many

different protocols including HTTP, DNS, and FTP. Most mainstream DDoS attacks focus on sending streams of data to layer 3 (network layer). Layer 7 DDoS attacks require much less data to be effective.

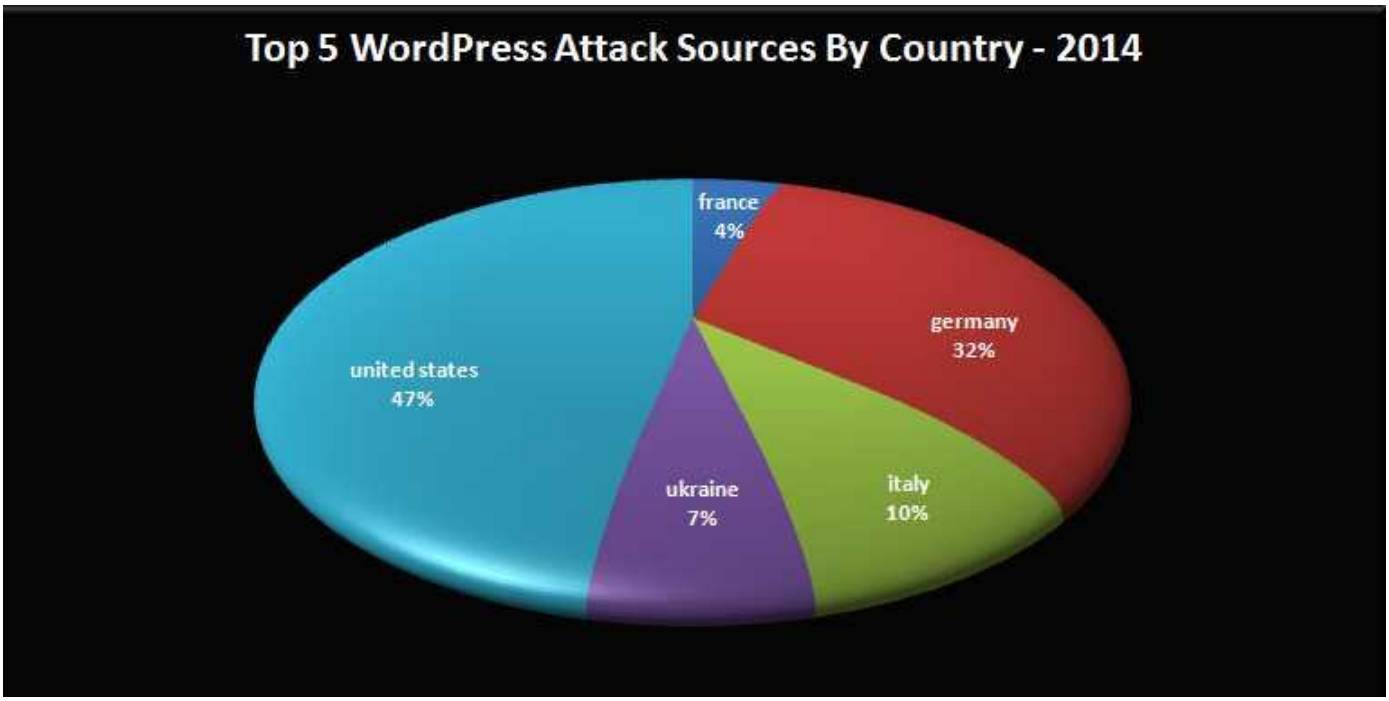
WORDPRESS ATTACK METRICS

IBM MSS data indicatesmany SQL Injection and Command Injection attackswere specifically targeting WordPress instances. In the chart below, we see WordPress installations being attacked heavily during the first three months of 2014. The pattern then diminishes from April through September where it then briefly resurges. Shellshock attacks against WordPress were noted in the November through December time frame, but were not numerous enough to include in the data. The data represents actual Security Incidents where customers were notified of these attacks. The data query was focused primarily on when the path to WordPress was found within a SQL Injection or Command Injection Security Incident.



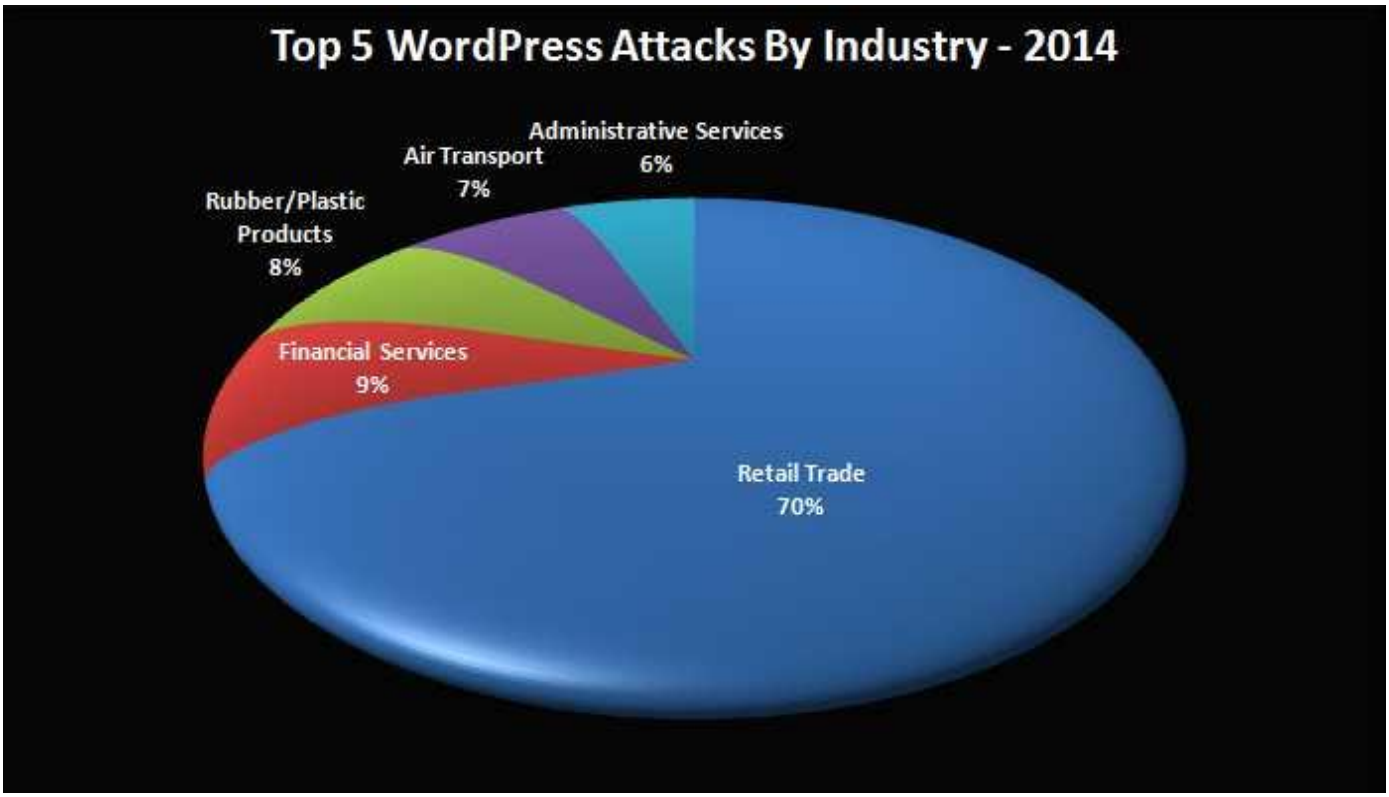
GEOGRAPHICAL DISTRIBUTION OF ATTACKSOURCES

IBM MSS WordPress attack metrics indicate that the United States was recognized as the largest source of attacks in 2014. A report issued in October 2014 by Imperva entitled the Web Application Attack Report (Waar), blames the United States as the most frequent source of WordPress attacks and cites that attackers from other countries are using hosts within the U.S. to launch attacks due in large part because they are closer to their targets. IBM MSS attack metrics indicate the same findings.



INDUSTRIES MOST ATTACKED

Retail Trade was by far the most WordPress attacked industry in 2014, followed by finance which was a distant 2nd. The Web Application Attack Report (Waar), also indicated that the Retail Trade sector was the most frequently attacked WordPress target followed by the Finance Industry. The metrics collected by IBM MSS reflect their findings exactly.



WHO IS USING THIS ATTACK?

Due primarily to the weaknesses outlined in this report, WordPress is a very appealing target to both hackers and security researchers. Tools are very easy to obtain online that enable hackers to perform a wide variety of attack types on many CMS brands. Security Researchers, however, perform a white hat service to the open source industry by trying to identify weaknesses in order to help authors design patches and mitigation procedures as well as providing information sharing concerning the weaknesses they find.

RECOMMENDATIONS/MITIGATION TECHNIQUES

Always run the latest version of any CMS.

Update CMS systems regularly. Look specifically for vulnerability patches and bug fixes.

Always use trusted sources for themes and plugins. Never use free themes and plugins.

Never use default settings. Change the default "ADMIN" name. Rename default database prefixes to prevent SQL Injection.

Reduce credentials. The administrator account should only be needed for performing updates or adding/changing themes and plugins. Those that are editing posts or writing articles should never need to be at an administrator level.

Always utilize strong passwords.

Protect the .htaccess file. The following code, added within the .htaccess file will prevent anyone from reading or writing any files that begin with "hta". (see "Securing .htaccess" in the References section)

Use a Cloud-Based Security Service. Solutions such as Cloudflare and Akamai act as a shield in front of your website. These services block bad user agents and offer some protection against SQL Injection and DDoS attacks.

Backup your CMS installations at regular intervals and design a robust disaster recovery plan.

IDPS SIGNATURES AND/OR SIEM RULES

See Attachment 1 for this information.

REFERENCES

CMS history

http://www.contegro.com/info-center/designers-blog/blog-article/_thread_/a-brief-history-of-cms-development

Content Management Systems Security and Associated Risks

https://www.us-cert.gov/ncas/alerts/TA13-024A

Securing .htaccess

http://thematosoup.com/wordpress-security-htaccess/

Is your WordPress site being used as an DDoS attack source?

http://labs.sucuri.net/?is-my-wordpress-ddosing

The perils of freeware

http://premium.wpmudev.org/blog/free-wordpress-themes-ultimate-guide/

Securing and hardening Content Management Systems

http://www.luminweb.com/clients/knowledgebase.php?action=displayarticle&id=9

162,000 WordPress Sites used in DDoS attack

http://arstechnica.com/security/2014/03/more-than-162000-legit-wordpress-sites-abused-in-powerful-ddos-attack/

OWASP list of top 10 web vulnerabilities

http://owasptop10.googlecode.com/files/OWASP%20Top%2010%20-%202013.pdf

WordPress Most Attacked Apllication

http://www.computerweekly.com/news/2240232352/WordPress-most-attacked-application

Web Application Attack Report (WAAR)

http://www.imperva.com/docs/HII_Web_Application_Attack_Report_Ed5.pdf

CONTRIBUTORS

Lyndon Sutherland - Security Specialist XFTAS

Michelle Alvarez –Researcher/Editor, Threat Research Group

Nick Bradley – Practice Lead, Threat Research Group

DISCLAIMER

This document is intended to inform clients of IBM Security Services of a threat or discovery by IBM Managed Security Services and measures undertaken or suggested by IBM Security Service Teams to remediate the threat. The data contained herein describing tactics, techniques and procedures is classified Confidential for the consumption of IBM MSS clients only.