



## KEEPING UP WITH PHP

Are you getting the best performance  
and security PHP offers?

PHP has proven to be the platform of choice for business-critical applications. With roots as a community-driven web-scripting language with little enterprise backing, PHP evolved rapidly into a driver of web and mobile growth. PHP grew fast because it embodied two core tenets: Stay simple and be easy to adopt.

Usage statistics indicate that PHP accounts for over 80 percent of all websites<sup>1</sup>, topping 240 million sites according to a Netcraft web server survey<sup>2</sup>. PHP has closed the gap since its early incarnations and has far surpassed Java and .NET in the diverse world of web and mobile application platforms.

Now with the release of PHP 7, new language constructs and significant performance and resource utilization optimizations make it a natural choice for fast-pace business-critical applications. PHP is the runtime environment backing many of the world's highest-traffic websites and largest open source application platforms such as Drupal, Magento, and WordPress, which by itself has a market share of over 50 percent of all content management systems<sup>3</sup>. PHP is still fundamentally a community-maintained open source dynamic platform with commercial support provided by Zend, a Rogue Wave company, one of the primary contributors to the evolution of the language.

---

<sup>1</sup> [http://w3techs.com/technologies/overview/programming\\_language/all](http://w3techs.com/technologies/overview/programming_language/all)

<sup>2</sup> <http://news.netcraft.com/archives/2013/11/01/november-2013-web-server-survey.html>

<sup>3</sup> [http://w3techs.com/technologies/overview/content\\_management/all](http://w3techs.com/technologies/overview/content_management/all)

# INNOVATION IS ACCELERATING

The pace of innovation in PHP, which delivers new and phases out older releases yearly, allows it to adapt quickly and continuously to best practices and industry requirements. As a result, the performance, quality, and maturity of the platform are continuously improved with every new version. The expansion of PHP is driven by organic custom application growth and accelerated by the large adoption of PHP-based application platforms. PHP transformed from primarily community-driven projects to the standard in enterprises, often used for development of mission-critical applications. This drives PHP developers to quickly consume new methodologies and industry best practices. The enhancements in each new PHP version vary from fundamental language constructs such as the introduction of namespaces in PHP 5.3 to better support for larger application and modern frameworks, to driving coding practices to achieve better code quality such as register globals behavior in PHP 5.4. Competing with other enterprise languages drives optimization in performance and resource utilization, which drive innovations such as including the Zend byte code acceleration extension in PHP 5.5 to increase execution efficiency and extensive codebase refactoring to dramatically improve performance in PHP 7.

These constant changes in the language and the runtime engine have driven developers to write better, more secure, highly maintainable, and efficient code. PHP provides developers with language elements that make it easier to adopt design patterns and modern methodologies but also drop some bad habits that may lead to security or quality issues. These advantages also find their way into the frameworks and open source application platforms that often serve as the foundation for enterprise applications. Companies that keep up with new PHP versions are better positioned to deliver higher quality, efficiency, and more secure code, thereby allowing their developers to spend more time on adding new business-critical functionality.

## MAXIMIZING PHP PERFORMANCE

Performance and resource consumptions are a key consideration of any platform suited for business-critical application development. Practically every new PHP version has demonstrated performance improvements and resource utilization optimizations. According to many benchmarks, PHP 5.4 is nearly twice as fast as PHP 5.1 and 5.2<sup>4</sup>. Introducing the Zend OpCache in PHP 5.5 gives most web requests a substantial performance acceleration by caching the byte code to reduce runtime code interpretation. In addition to performance, a major effort was launched in PHP 5.5 to significantly optimize the memory consumption of the runtime. The performance focus continued to reach its peak in one of the most significant milestones in the evolution of PHP with the adoption of the PHPNG project as the new runtime for PHP 7. PHPNG was developed as a rewrite of the PHP runtime engine by Zend with the help of its industry partners in an effort to accelerate the performance of real-world PHP applications. The runtime was later adopted by the community, who continued the optimization efforts.

For most real-world applications such as eCommerce and CMS platforms, PHP 7 offers a significant performance boost. With execution time more than twice as fast compared to PHP 5.6 and 30 percent lower memory consumption, servers running PHP 7 are able to serve up to three times as many Magento requests as those running PHP 5.6<sup>5</sup>.

---

<sup>4</sup> <http://www.lornajane.net/posts/2012/php-5-4-benchmarks>

<sup>5</sup> [https://www.zend.com/en/resources/php7\\_infographic](https://www.zend.com/en/resources/php7_infographic)

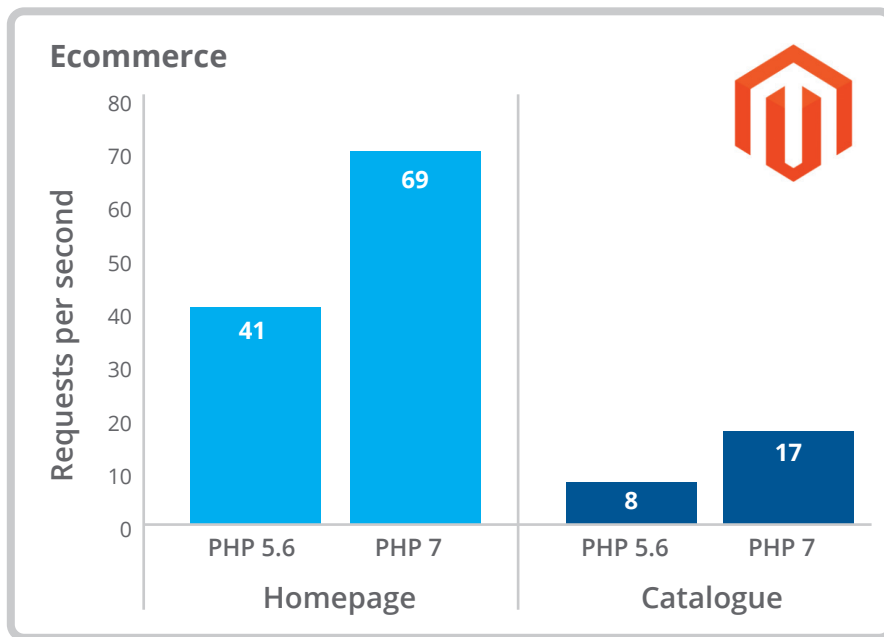


Figure 1: Magento 1.9, PHP 5.6 vs. PHP 7

Drupal 8 runs 72 percent faster and WordPress only executes 25M CPU instructions on a PHP 7 runtime compared to just under 100M to do the same job on older PHP versions<sup>6</sup>.

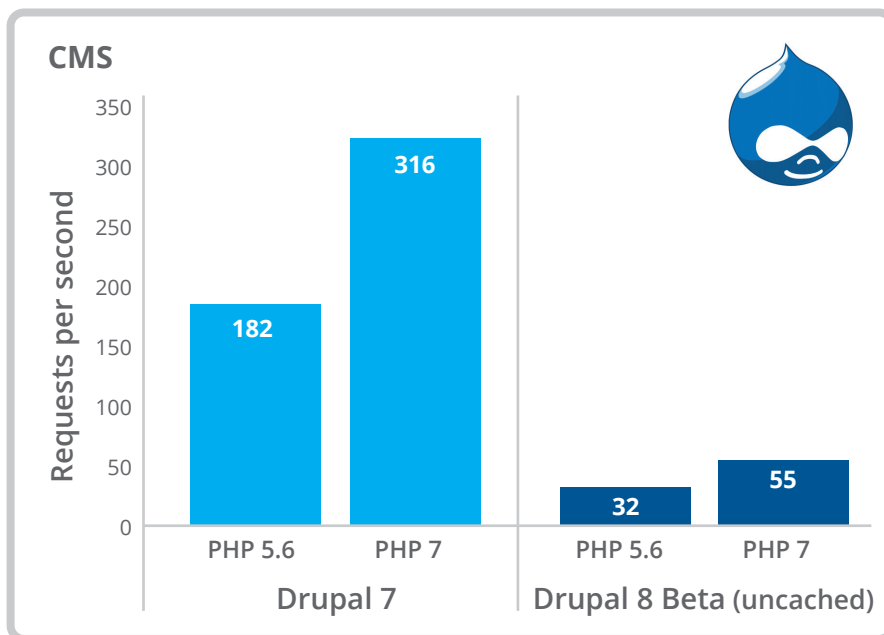


Figure 2: Drupal, PHP 5.6 vs. PHP 7

<sup>6</sup> [https://www.zend.com/en/resources/php7\\_infographic](https://www.zend.com/en/resources/php7_infographic)

# THE SECURITY CHALLENGE

Application security is a major concern and risk factor for business. In industries such as eCommerce, financial, and healthcare, data breaches carry regulatory impacts in addition to financial and reputation losses. PHP is no different than the other widely utilized application runtime platforms. It's a well maintained open source project with a strong contributor support, however, multiple vulnerabilities are still identified each year. According to the Common Vulnerabilities and Exposures (CVE) database, PHP averages between 20-25 vulnerabilities per year<sup>7</sup>, many of those impacting multiple PHP versions.

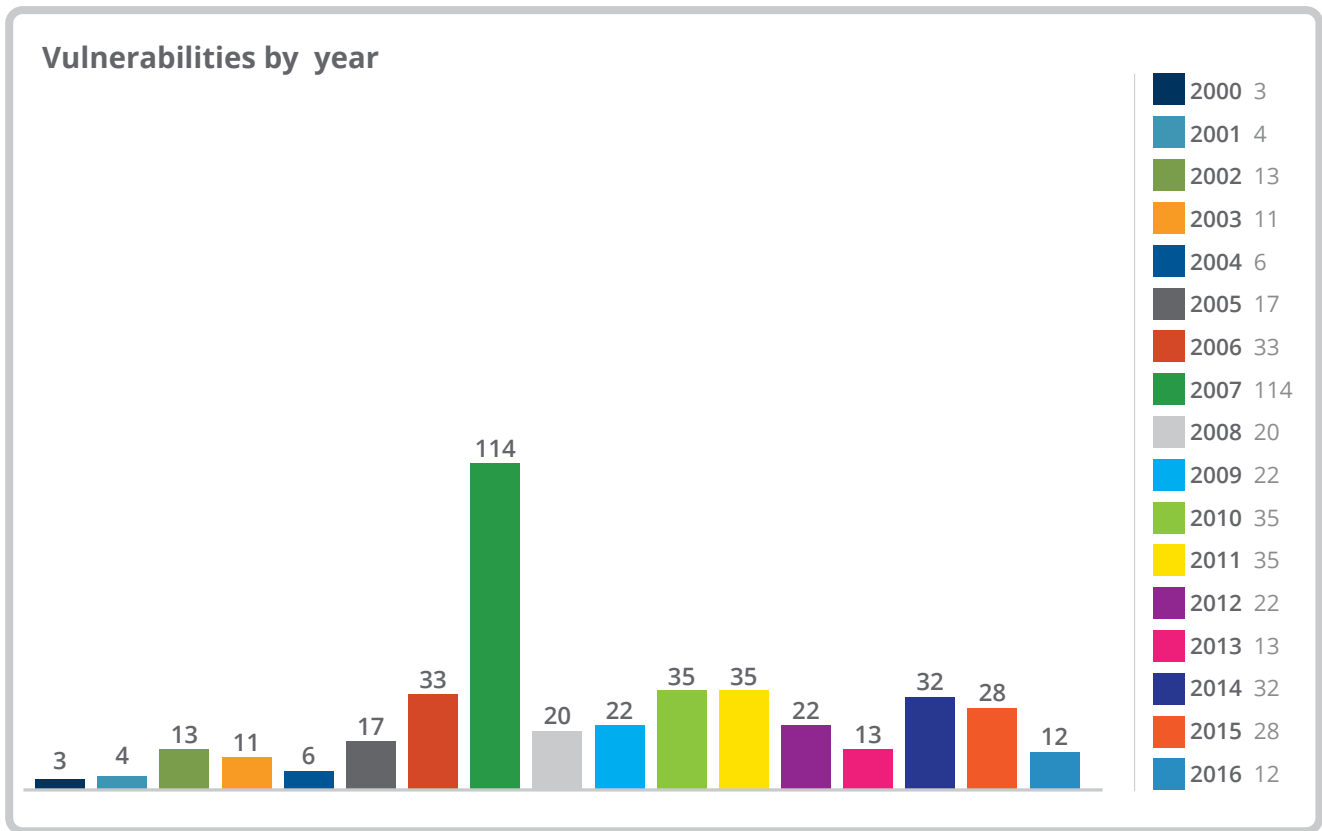


Figure 3: CVE database vulnerabilities by year

The average total cost of a data breach is \$3.79 million or \$154 cost per lost or stolen record. Forty seven percent of all breaches are caused by malicious or criminal attacks<sup>8</sup>. Though one cannot completely secure an application runtime environment, the security perception of a platform is often attributed to the speed in which vulnerabilities are fixed and released to the market. The PHP community does in fact respond quickly to identified threats and regularly releases maintenance versions for security issues deemed critical. The PHP community provides defect support for a period of two years, and security fixes from three years after a version is release. Zend provides long term commercial support with guaranteed SLA for both defects and security issues for no less than five years. It's still very common, however, to encounter production applications running on old unsupported and insecure runtime versions.

<sup>7</sup> [http://www.cvedetails.com/product/128/PHP-PHP.html?vendor\\_id=74](http://www.cvedetails.com/product/128/PHP-PHP.html?vendor_id=74)

<sup>8</sup> 2015 Cost of Data Breach Study: Global Analysis, Ponemon Institute, 2015

# UNSUPPORTED, VULNERABLE ENVIRONMENTS

PHP 5 serves as the foundation for nearly 100 percent of all the websites whose server-side programming language is known<sup>9</sup>, including many of the world's most-visited websites. Surprisingly, the vast majority of PHP applications (71 percent according to W3Techs data) are running on older PHP runtime versions. This means that millions of applications and websites are relying on PHP 5.4 or older, versions that are no longer maintained by the open source community. These versions don't provide the performance gains introduced by PHP 5.5 and PHP 7. Unless supported by a commercial vendor, running applications on PHP 5.4 and older expose the business to a multitude of well-documented security vulnerabilities.

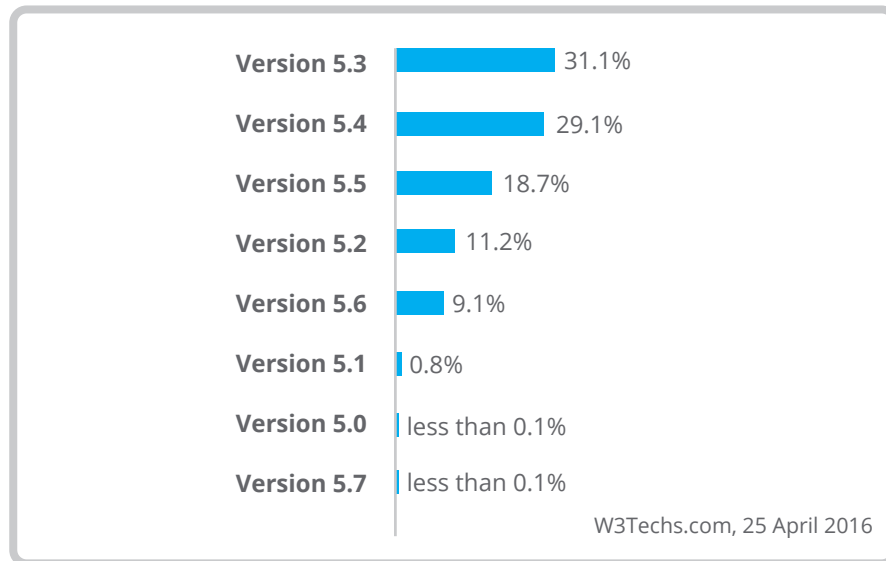


Figure 4: Percentages of websites using various subversions of PHP 5<sup>9</sup>

So why are companies running the majority of their code on older, unmaintained, and mostly unsupported runtime environments? There's certainly an awareness issue. PHP is considered very stable, which often causes IT organizations to delay proper maintenance and updating of their runtime environments. Most developers and IT managers realize they need to migrate their applications to improve their performance and security. However, they find it challenging to move forward with these projects usually due to time considerations, lack of expertise, and fear of introducing risks.

- **Time** – Migration and modernization projects are often perceived as a distraction to the business as they don't directly create new business functionality. Most development organizations operate under strict timeframes and don't have the bandwidth allocated to modernize their applications. Migrating between PHP versions requires an effort which includes code changes and testing.
- **Expertise** – Lack of experience is another common reason for delaying migration projects. Developers and architects focus on their domain of expertise and knowledge of their specific business. They aren't typically utilizing tools, nor do they possess the experience required for large migrations and code refactoring projects.

<sup>9</sup> <http://w3techs.com/technologies/details/pl-php/5/all>

- **Stability** – Constant desire for stability and fear of introducing risks deter organizations from stepping into a code refactoring and modernization process. The application works in production delivering its business value, and any attempt to modify the code or environment is an opening for problems, thus introducing unnecessary risks to the business.

Time, expertise, and risk mitigation translate to added costs which bring modernization and optimization planning to a halt. As the gap between the latest available release and the version being utilized widens, migration becomes more challenging.

## HOW CAN WE HELP?

Our consulting team helps plan, manage, and deliver PHP migration and modernization projects, and the Architecture Migration service is designed to help organizations successfully update PHP applications to take advantage of PHP 7 security and performance benefits. The methodology we apply utilizes both automated tools developed specifically for identifying code patterns that require refactoring, and manual review of the code. Automated tools reduce the migration effort by identifying deprecated functions and code patterns that require re-architecting. Understanding potential issues ahead of time accelerates the migration effort and streamlines the transition, saving time and frustration. Our consultants work with your developers to guide them through code modifications required for PHP 7. This hands-on mentoring approach provides developers with the skills and confidence to move forward with migrations and ensure optimal performance, security, and efficiency of applications. To learn more, visit [zend.com](https://zend.com).



Rogue Wave Software 1315 West Century Drive, Suite 150 Louisville, CO 80027  
Tel 800-487-3217 · Fax 303-473-9137

© 2016 Rogue Wave Software, Inc. Zend and Zend Server are registered trademarks of Rogue Wave Software, Inc.  
All other trademarks are the property of their respective owners.