Research on Security Analysis of PHP based Websites

Shijia LING[1, a]

[1]Zhongshan Polytechnic, Zhongshan 528400,China

[a]lingshijia@126.com

Keywords: PHP Programming; Security Analysis; Encoder Performance; Case Study

Abstract. The PHP as an open-source coding language is becoming more and more popular for companies, web developers and researchers. It is frequently adopted when building up a server. Protecting PHP code from being plagiarized is also a hot research issue especially with the rapid development of dynamic web industry and people's copyright protection consciousness. In general cases, encrypting the PHP based code blocks using encoder before making them available to the world is a way to protect the unsafe circumstances. Starting from different theoretical backgrounds, there are different kinds of PHP encoders. In this paper, we analyze and compare the security level of some well-known encoders. From a new point of view, we try to use the random statistical test analysis of the output of the encoder, which is never done. In addition, we show that our method is robust. We find out the well performed encoder and analyze it with sufficient experiment. Finally, we undertake an overview conclusion.

Introduction

With the progresses of Web, more and more dynamic web pages are developed to be used in every aspect of our lives, such as social network sites, electronic commercial shopping, management system and e-banking. Many of these sites are used to realize the PHP server[1]. As a server-side scripting language, PHP is designed for the production of dynamic pages with open source code. Instead of calling external functions, PHP is embedded into the HTML source code to handle the data. This is the first time in 1994 by Rasmus with C language programming method. Since the first release, we touch the rapid development of PHP. After a large number of optimization of the original release, it has now reached version 5. PHP in the Web PHP on the server processor get interpretation. It also can be used as shell other graphics applications independent of platform and operating system. PHP can be used to manage dynamic content, support database, processing session tracking, and even build entire station in electronic commerce. It supports many popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, Microsoft SQL server. Unlike other similar server-side scripting such as Microsoft ASP development, PHP is free, get more and more attention. People realize gradually the practicality of it. Now, PHP is widely used in the world [2].

PHP code protection has been the core concerns of many companies, such as some of the well-known forums such as vBulletin, Discuz, PHPWind and ShopEx. However, PHP technology is rapidly produce panic in Zend companies, or even the entire PHP user group. All related products, including almost all of the large PHP lysate. Their source code even published. In this case, Zend has admitted that all of the encryption technology has been solved. In this paper, we first analyze the security level of some well-known encoders. We compare these encoders from different aspects. We believe that the security from a new point of view by random statistical test. Based on the comparison of different testing random encoder output, gives several widely used more in-depth understanding of PHP encoder. Finally, we take the experiments on some PHP encoders and explain the results. Based on the results, we draw a more detailed and clearer conclusion about existing PHP encoders.

Our Proposed Methodology and Related Work

Related Work. There are several PHP encoders now. Some of them are free of charge, while others are not. Other lightweight encoders include PHP screw and PHPCodeLock. Micro-shield PHP encryption expert (PHPCodeLock) is good PHP script encryption software, without having to install any third-party components in the server-side. The encrypted files can be run on any ordinary PHP environment. This software requires no additional costs. Another PHP code called Web tiger can protect any type of file, not only PHP. For example, an additional extension of the file, the configuration file contains a user name and password to connect to the database. Web tiger is reliable and easy to use. It provides a silent installation, so that users can Web the tiger for the installation of their products. It has a simple user operation and convenient maintenance. According to different requirements, the Web three version of the tiger. The first version is not restricted, the machine is installed PHP application software.

Random Statistical Tests. In the field of internet and cryptography, the randomness of output sequence is an important measurement for the security. The defects of the output sequences are tested by the statistical tests. The focus of each random testing is a statistic and its distribution. Random judgment is based on the hypothesis testing theory. The following formula 1 is the definition of normal distribution. As a special case of the gamma distribution, the chi-squared distribution is a continuous distribution of other important statistics and probability theory field, which also marked as the χ^2 distribution. A chi-square distribution with k freedom degrees is the distribution of the sum of squared k independent variables with the standard normal distribution. Usually, the chi-square distribution is exploited in basic chi-square hypothesis tests for measuring the distance between an observed or empirical distribution and a theoretical distribution, and this is also called goodness of fit tests. Except applied in hypothesis tests. This distribution is also exploited by many other statistical tests, such as Friedman's analysis of variance by ranks [3].Also, it is used for classifying the independence of two criteria of qualitative data, estimating a population standard deviation of a normal distribution from a sampled standard deviation, determining the confidence interval field. The detailed definition of chi-square distribution is as follows in the formula 2.

$$f(x)=\frac{1}{\sqrt{2\pi\sigma}}e^{\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < \infty \tag{1}$$

$$f(x)=\begin{cases}\frac{1}{\Gamma(v/2)2^{v/2}}x^{v/2}e^{x/2}, & 0 \le x < \infty \\ 0 & x < 0\end{cases} \tag{2}$$

Proposed Work. As mentioned before, there are many statistical tests used for testing the randomness of the outputs. And the random levels of these output can reflect the security of the algorithm. Because of the performance of PHP encoder, we can also use this idea to test their safety. PHP PHP code encoder to output file can predict the none ideal case under some meaningless. Once the none random behavior is found in the output file, PHP code can be recycled and PHP encoder is not considered safe. We choose the appropriate statistical tests, the formation of the test suite is the most suitable for testing these none random of the PHP output. We focus on several well-known tests, select some they form our suite and explain them for PHP encoder. (1) The Frequency Test: This test calculates the hamming weight of a sequence, i.e. the number of 1's. Since random binary sequence should take 0 and 1 with equal probability, the number of 0 and 1 should be approximately equal if the output is random. So the test calculates the number of 0 and 1 of the sequence, and eliminates the sequence which has overmuch 0 or 1. This test is the basis for other tests and should be taken firstly each time. The sequences eliminated by this test usually fail in other tests. (2) The Run Test: The test focuses on the number of runs in a sequence. The sections with consecutive 0 or consecutive 1 are called run. For the sequences with the same hamming weight, too large run length or too small run length both indicate non-randomness. The number of runs also reflects the change frequency of 0, 1 in the sequence. (3) The Frequency Test in Sub-sequences: This is a test with secondary statistics. That is, it is the second-level version for frequency test. This test is to examine whether the number of 0 and 1 in each subsequence satisfies the requirements. When N = 1, this test is the same as the basic frequency test. (4) The Longest Run Test in Sub-sequences: Similarly, it is also a statistical test. This is the second edition of the running test. This test is to check whether the number of 0 and 1 of each sub sequence satisfies the requirements of the. When n = 1, this test is the same as the basic running test. As a sub sequence with known the Hamming weight of 1, the longest running should meet certain distribution.

Experiment Analysis and Simulation

The Poker Test. Note that in this test, N should satisfy that $N \geq 5 \cdot 2^M$ which also divide the Sequence into Non-overlapping subsequences each of length M. Let n_i be the number of occurrences of the i^{th} type of sequence of length M, The poker test decides if each mode of subsequences of length M appears approximately the same number of times in the original sequence, as would be expected for a sequence with random behaviors. (1) Compute the $t = \frac{2^M}{N}\left(\sum_{i=1}^{2^M} n_i^2\right) - N$. This approximately follows a chi-square distribution. (2) Compute the p-value based on t to judge whether the sequence is random. Note that the poker test is a generalization of the frequency test: setting m = 1 in the poker tests yields the frequency test.

The Autocorrelation Test. The purpose of this test is to check the correlations between the tested sequences and its non-cyclic shifted version. Let d be a fixed positive integer. The number of bits in the sequence not equal to their d-shifts is $A(d) = \sum_{i=0}^{n-d-1} s_{i+d} \Theta s_i$. Θ is the XOR operation.

The General Result. We carry out the experiments on the Zend Guard encoder and the Ioncube encoder with our test suite. We use these two PHP encoders to generate the outputs with different lengths and test them. We compare and analyze their results in Table 1and Table 2. Note that we show the average values of the results instead of every p-value. These average p-values are more reasonable and correct. They can reflect more about the results. We also divide the outputs into two classes: one is for the outputs with long lengths; the other is for the outputs with short lengths.

The tests	P-value for Long length	P-value for Short length
The frequency test	0.837	0.799
The run test	0.673	0.645
The frequency test in sub-sequences	0.465	0.503
The longest run test in sub-sequences	0.345	0.493
The non-overlapping test	0.787	0.605
The overlapping test	0.323	0.418
The poker test	0.566	0.645
The autocorrelation test	0.846	0.808
The linear-complexity test	0.733	0.629
The Binary Matrix Rank Test	0.423	0.597
The random excursions test	0.535	0.601

Table. 1 The Average P-value of Zend

From this table, we can conclude that for the Ioncube encoder, the results for both long length output and short length output are still similar and have a consistent trend. The frequency test in sub-sequences, the non-overlapping test and the linear-complexity test have the best results this time, which mean that the output of the Ioncube encoder possesses the highest randomness in term of these three tests. As we mentioned before, we prove that our test suite can be carried out in a reasonable computing time. In fact, they ran as fast as we imagine. Table 2 shows the output for each test case, run time of two PHP encoders. The frequency test and the run test cost the least running time for both Zend Guard encoder and the Ioncube encoder. These two tests can be implemented easily in software.

Also, the autocorrelation test and the linear-complexity test require the most running time, which suggests that they need a more complicated implementation in software.

The tests	Running Time for Zend(s)	Running Time for Ioncube(s)
The frequency test	0.007	0.007
The run test	0.008	0.008
The frequency test in sub-sequences	0.013	0.012
The longest run test in sub-sequences	0.024	0.023
The non-overlapping test	0.078	0.079
The overlapping test	0.034	0.033
The poker test	0.056	0.054
The autocorrelation test	0.086	0.088
The linear-complexity test	0.099	0.096
The Binary Matrix Rank Test	0.035	0.041
The random excursions test	0.047	0.051

Table. 2 The Average P-value of PHP Encoders

Summary and Conclusion

PHP is used more and more widely by the Web designers due to its open source property. One of the problems about the PHP security is that it can be plagiarized and copied after finishing. With the rapid progress of Web developments, people concern more about the copyright protection now. So we focus on how to protect PHP source code. Usually the PHP codes are encrypted with PHP encoders before handing them out. Different kinds of PHP encoders possess different features and performances. In this research report, we conduct investigation on the PHP websites’ safety. First is the Zend Guard and second is the Ioncube. From a different aspect, we propose a new method for analyzing and comparing their output. We originally exploit the random statistical tests as a tool for measuring PHP encoders and demonstrate the reasonableness of our idea. We give a test suite which is proper for PHP encoders. Finally, experiments are taken to support our point of views. In the later research, we plan to conduct more complicated classification work such as kernel classification [4-6] to classify the different unsafe elements.

References

[1] P. Hellekalek, S. Wegenkittl, Empirical evidence concerning AES. ACM Transactions on Modeling and Computer Simulation (TOMACS) Volume 13, Issue 4 2003, 322 – 333, 2003.

[2] I. J. Good, The serial test for sampling numbers and other tests for randomness. In: Proceedings of Cambridge Philosophical Society, vol. 49, pp. 276–284, 1953.

[3] G. Marsaglia, J. Marsaglia, Evaluating the Anderson- Darling distribution. Journal of Statistical Software 9(2), pp. 1–5, 2004.

[4] Wang, Haoxiang, and Jingbin Wang. "An Effective Image Representation Method using Kernel Classification."

[5] Zhu, Changming, and Daqi Gao. "Improved multi-kernel classification machine with Nyström approximation technique." Pattern Recognition (2014).

[6] Mangasarian, Olvi L. "Unsupervised classification via convex absolute value inequalities." Data Mining Institute Technical Report (2014): 14-01.