UNIVERSITY OF WATERLOO

Faculty of Engineering

Department of Electrical and Computer Engineering

Load Analysis of the Magento Commerce Platform on Servers Running OpenVZ

Canfone.com Inc.

Montreal, Quebec

Prepared by

Michael A. Soares

ID [removed]

userid masoares

2B Computer Engineering

10 May 2010

[address removed]

10 May 2010

Dr. Manoj Sachdev, chair
Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario
N2L 3G1

Dear Sir:

This report, entitled "Load Analysis of the Magento Commerce Platform on Servers Running OpenVZ", was prepared as my 2B Work Report for Canfone.com Inc. This report is in fulfillment of the course WKRPT 300. The main purpose of this report is to quantitatively analyse the performance of the Magento Commerce platform, an online e-commerce platform available for creating a variety of online stores, running on the CentOS operating system using the OpenVZ kernel in terms of server load (i.e., CPU usage) as well as the simulated load time from a test client machine and to offer several recommendations which may help in improving the performance of this software running in a virtual container.

I was employed as an Internet Application Developer working with one other developer managed by Ben Blakley; my role was overseen by Tyler Potter. My primary responsibilities included scripting a custom wrapper in PHP for the Tucows OpenSRS API and integrating it with the Magento Commerce platform, as well as developing custom solutions for a variety of platforms (including, but not limited to Magento and osCommerce) for clients using a variety of programming languages, including PHP, MySQL, Perl, JavaScript, and bash scripts.

I would like to thank Mr. Tyler Potter for providing me with encouragement, valuable ideas and information over the work term which helped me in finalizing this report. I hereby confirm that I have received no further help other than what is mentioned above in writing this report. I also confirm this report has not been previously submitted for academic credit at this or any other academic institution.

Sincerely,

Michael A. Soares
ID [removed]

Contributions

For the past four months, I was employed at Canfone.com Inc. (“Canfone”) in Montreal, Quebec as an Internet Application Developer. I was one of the company’s three employees which included the Vice-President, who also developed various Internet applications, as well as the President himself. I worked and collaborated with both of them on a daily basis, dealing mostly with Internet application development and database manipulation, as well as some server management and troubleshooting.

My colleagues and I were primarily responsible for developing custom solutions as well as modifying and upgrading web applications (code-wise) in use by Canfone and its customers. Software is developed and modified to meet Canfone’s needs by working collaboratively with project leaders by using source code repositories to track and revise changes to all production code. All Internet applications at Canfone, either custom developed or modified for a customer’s needs, are run through a thorough code review process and are then tested by the President before being put into the production environment or on a customer’s site.

While at Canfone, my primary responsibilities involved working with new and existing software code, but more specifically encompassed:

- Scripting a custom wrapper in PHP: Hypertext Preprocessor (“PHP”) for the Tucows OpenSRS API and integrating it with the Magento Commerce platform,
- Creating a free and automatic deployment system for Magento via the cPanel/WebHost Manager (WHM) backend,
- Developing custom plug-ins and add-ons for customers using the Magento and osCommerce platforms,
- Debugging and properly commenting already-existing code,
- Manipulating database objects in MySQL databases,
- Managing Linux servers using the cPanel/WHM backend on CentOS running the OpenVZ kernel, and

- Researching new technologies to evaluate their use within Canfone's existing environment.

For the majority of the term, I was diligent in aiding in the development of a Secure Socket Layer (SSL) certificate reseller website, SSLTree.com, which required interacting with the Tucows OpenSRS API for purchasing and provisioning SSL certificates. At different times throughout the term, I also researched different methods of developing custom plug-ins and add-ons for the cPanel/WHM backend to aid in deploying software to customer websites. The result of my findings involved creating an automated installer for the Magento platform, available at MagentoInstaller.com, which any customer can use to deploy the software to his or her site(s). This involved using PHP to interact with a Perl script which then calls several bash commands on a Linux-based server.

The research that was done on this topic was quite insufficient in scope to be suitable for a work term report. However, because I had familiarized myself with the Magento platform as well as working with virtual servers on the OpenVZ kernel, Mr. Ben Blakley, one of my supervisors, gave me the opportunity to do a variety tests using Magento and virtual servers. This would lead to finding the optimal specifications for running Magento on a virtual server of the same nature without impacting the performance of the e-commerce stores or dynamic websites it would be serving over the Internet and without impacting other virtual servers running on the same physical hardware node. Thus, I took it upon myself to perform a variety of load tests using different virtual server configurations and offer a series of possible solutions, or rather, optimal configurations that Canfone can use when it eventually decides to deploy or upgrade the software across the majority of its customers' e-commerce sites.

This is the main relationship between this report, the knowledge I gained and the tasks I performed while working at Canfone. The data collected and the analyses performed in this work term report are beneficial to me in many different ways, primarily because it has given me the opportunity to learn well beyond what I thought I would as an Internet

Application Developer. This project and this subsequent report have also provided me with the ability to benchmark applications and evaluate the resultant quantitative data.

In the broader scheme of things, my research on this report topic should prove to be beneficial for Canfone. Since Canfone's user base is growing at a rather fast rate, it must use the resources it currently has to optimally run its applications and its customers' sites without sacrificing performance. In this report, I provide Canfone with several recommendations on how to properly configure its virtual servers in order to improve upon the performance of its current websites, its customers' current websites, and possibly even future websites which are/will be running the Magento Commerce platform.

Executive Summary

The main purpose and scope of this report is to qualitatively and quantitatively analyse the server load caused by accessing a Magento-based website and the resultant performance of said website via the Internet running on resource-limited or differently configured VPSs on a single physical hardware node. This report will suggest to Canfone ways of configuring its current and future set of virtual private servers so as to decrease the amount of time spent troubleshooting performance-related issues, all while keeping performance of said servers at their best and keeping their server loads at a minimum. I have identified several recommendations in this report that will optimize the performance of multiple virtual private servers running on a single physical hardware node which will allow Canfone to deploy or modify its virtual private servers without worry.

The major points in this report are memory allocation has little to no effect on server load or website performance, but CPU time allocation does. The first section sets out the scope, purpose, and outline of the report. The second section describes the Magento platform that was tested and the environment it was tested in. The third section describes the test methods and tools used to impose a load on and gather data from the virtual private server the Magento-based website was hosted on. The final sections provide conclusions and recommendations based on the analyses in the preceding sections.

The major conclusions of this report will confirm that a minimal amount of memory allocated to a virtual private server has little no effect on the performance/page load/request time of Magento-based websites hosted on said server, as well as on the server load of the server itself. In addition, it will also show that limiting the amount of CPU time allocated to a virtual private server can have drastic effects depending on the percentage of CPU time allocated to any one virtual private server.

Major recommendations in this report are also identified in that Canfone should allocate memory to any one virtual private server at its discretion since the effect on server and

website performance will be minimal or non-existent. Canfone can also change the secondary memory limit on PHP on any one VPS at its discretion, keeping in mind that there may be a possibility of increased memory usage on the server as a result. Lastly, Canfone should allocate at least 50% of a hardware node's CPU time to a single VPS so as not to affect the performance of the server, the performance of the website, and an end-user's browsing experience.

Table of Contents

Contributions ... iii
Executive Summary ... vi
List of Figures ... ix
List of Tables ... x

1 Introduction ... 1

1.1 Virtual Private Servers and Running Applications ... 1
1.2 Purpose ... 2
1.3 Scope ... 2
1.4 Outline ... 2

2 The OpenVZ Kernel and Magento Commerce ... 4

2.1 An Introduction to OpenVZ ... 4
2.2 The Magento Commerce Platform ... 4

3 Quantifying Performance ... 6

3.1 Introduction ... 6
3.2 Test Methods and Tools Used ... 6
3.3 Load Test Results ... 7
3.3.1 Memory-Limited Configurations ... 7
3.3.2 CPU-Limited Configurations ... 9
3.4 Optimal VPS Configurations ... 10

4 Conclusions ... 12

5 Recommendations ... 13

Glossary ... 14
References ... 15
Appendix A – Memory-Limited Configuration Test Results ... 16
Appendix B – CPU-Limited Configuration Test Results ... 18

List of Figures

Figure 1. A sample Magento store created using free sample data available online. 5

Figure 2. Graphed results of data obtained from Table 1. 8

Figure 3. Graphed results of data obtained from Table 2. 10

List of Tables

Table 1. Load test results from memory-limited VPSs... 8

Table 2. Load test results from CPU-limited VPSs. 9

1 Introduction

Canfone.com Inc. ("Canfone") specializes in providing fully customized e-commerce solutions to its customers, integrating secure ordering, secure payment gateways, and shipping into affordable "eStore" packages for many different types of businesses [1]-[3]. Canfone also provides web service consulting, custom web development services, and web hosting (including shared, virtual, and dedicated hosting) to many of its customers.

Canfone is currently in the early planning stages of migrating many of its customers' e-commerce websites, currently running under a legacy platform called osCommerce, to the Magento Commerce platform, a more customizable, manageable, and overall, simple-to-use platform available, with community support, for free online [4]. Magento is increasingly becoming more popular on the web due to the amount of features it comes packaged with and due to the fact that, with the right amount of knowledge, it can be fairly simple to create new add-ons, or extensions as they are better known, and distribute them to the Magento community [4]. This platform has already been adopted by several well-known companies, including, but not limited to Samsung, 3M, and Lenovo, and its use is growing daily.

In this section, the purpose and scope of the report are both set out and essential background information is presented on the topic.

1.1 Virtual Private Servers and Running Applications

In general, virtual private servers ("VPS") can run independently from each other on the same physical hardware node ("node"), utilizing, sharing, and managing the same resources and hardware (i.e., processor ("CPU"), memory, disk space, etc.) among themselves [5]. The software applications on said VPSs run just as they would on a normal piece of hardware, however, without the constant amount of dedicated resources available to them. Canfone runs most of its clients' websites on VPSs to keep multiple instances of applications separate from each other and to make better use of its servers so that no resources go unused, all while making sure that server performance remains at its best and is unaffected.

1.2 Purpose

Because Canfone runs most of its customers' websites on VPSs running the OpenVZ kernel, or rather, individual server containers running on top of a software layer on the same physical node, a limit as to how many individual VPSs serving e-commerce websites, using Magento as their front- and back-ends, can run on the same node will have to be imposed. This report will analyse the load the Magento platform produces on a variety of differently configured VPSs and suggest different ways of allocating resources to said servers running on the same node so as not to affect the performance of the application itself (i.e., Magento) and any other VPSs.

1.3 Scope

This report will include both qualitative and quantitative analysis of the server load caused by a Magento-based website and its performance via the Internet running on resource-limited or differently configured VPSs on a single physical hardware node.

1.4 Outline

The sections in this report identify and summarize the use of both the OpenVZ Linux kernel and the Magento Commerce platform. This report also provides a qualitative and a quantitative analysis of the server load caused by accessing a Magento-based website via the Internet running on resource limited VPSs under the OpenVZ kernel on a single physical hardware node. A glossary has also been included for easy reference of technical terms used in this report. Section 2 introduces the OpenVZ Linux kernel in more detail as well as the Magento Commerce and outlines some of the key features included with these two pieces of software. Section 3 explains the methods used to impose a load on each of the resource-limited VPSs so as to gather data. It also shortly describes the tools used to gather the resultant data and attempts to provide justifications for the results obtained. Finally, conclusions and recommendations are outlined at the end of the report.

2 The OpenVZ Kernel and Magento Commerce

2.1 An Introduction to OpenVZ

OpenVZ, in itself, is simply a Linux kernel that allows for multiple instances or containers of Linux operating systems to run separate from each other on the same node utilizing and sharing the same resources among themselves as all VPSs do, as previously mentioned in section 1.1 [5]. Each container runs independently from the others, thus each one can have its own IP addresses, users, files, applications, disk quotas, etc. and can even be rebooted without affecting any of the other containers on the node [5].

The main advantages to using OpenVZ over other virtualization software include soft memory allocation, meaning that unused memory in one VPS can be used by other VPSs on the same hardware node, as well as the ability to directly access a VPS's file system through the hardware node without using any other proprietary software, since each VPS's file system is simply a directory of files on the node itself.

2.2 The Magento Commerce Platform

The Magento Commerce platform, as mentioned in section 1, is a very customizable, manageable, and overall, simple to use e-commerce platform for both customers and webmasters/administrators of websites alike. Fully functional online stores or dynamic websites can be put together in a very short period of time using Magento. The software only requires a basic Linux, Apache, MySQL, and PHP or "LAMP" installation in order to run [6]. Extensions or add-ons for the platform can either be downloaded from the Magento website (most are free) or they can be easily coded together if they don't already exist using a variety of tools and tutorials, also available online for free [4].

Once installed using the sample data from the Magento Commerce website, a basic installation will resemble something similar to that depicted in Figure 1 on the following page.

Figure 1. A sample Magento store created using free sample data available online.

In addition to the free extensions available for Magento online, there are also a multitude of themes or skins available for companies to use to give their front-end stores or dynamic websites a different appearance other than the default one presented in Figure 1 above.

3 Quantifying Performance

3.1 Introduction

Before having measured the performance of the Magento platform on different VPS server configurations, it was initially believed that modifying the amount of memory available to the VPS would have little to no affect on Magento's performance, but modifying the percentage of CPU time available to the VPS would have a drastic effect. These two assumptions were based on the following facts, respectively:

1) The secondary memory limit set on PHP at runtime would impose the same upper limit on Magento's memory usage on all different VPS configurations, and
2) Since PHP is an interpreted language, code being run is interpreted at runtime, thus it would require quite a bit of processing time in order to output a result every time.

For the purpose of this report, only the amount of memory assigned to VPSs was changed during tests. The memory limit set on PHP was kept constant.

3.2 Test Methods and Tools Used

Load tests were conducted on a hardware node with approximately four (4) gigabytes (GiB) of available memory and a quad-core processor capable of running at 3.4 gigahertz (GHz), running the CentOS 5.4 operating system with the OpenVZ kernel. Disk space was not an important factor when conducting these tests and was not taken note of. A single VPS was setup on the hardware node with a basic LAMP installation serving a Magento-based store or website with sample data.

The tools used to conduct the load tests on the Magento platform included:

- Apache Bench, or `ab`, to simulate one thousand (1000) consecutive page requests and measure their load times from a remote computer for the homepage being served by the Magento-based sample store,

- `uptime`, sampled every five (5) seconds on the VPS to measure its CPU load average over the testing period for each VPS configuration,
- `vzsplit`, a tool included with OpenVZ to aid in equally distributing resources among VPSs [5], and
- `vzctl`, another tool included with OpenVZ to aid in managing multiple VPSs on a single hardware node (i.e., assigning resources to, starting, stopping, and restarting VPSs) [6].

The same tests were conducted on twenty (20) different VPS configurations, where each configuration differed either in the amount of memory assigned to the VPS or in the amount of CPU time dedicated to the VPS by the hardware node. All data were collected simultaneously from `ab`, `uptime`, and `free` and have been outlined in 3.3 below.

3.3 Load Test Results

As mentioned in 3.2, twenty (20) different VPS configurations were used to conduct load tests. The ten (10) different memory configurations for the VPSs were setup by using the `vzsplit` tool to equally assign the total amount of available physical resources (on the hardware node) to between one and ten VPSs all while keeping the maximum CPU time guarantee at 100%. In other words, VPSs were assigned 10%, 11.11%, 12.5%, 14.29%, 16.67%, 20%, 25%, 33.33%, 50%, or 100% of the physical memory available.

The ten (10) different CPU configurations for the VPSs were setup by using the `vzctl` tool to guarantee the VPSs' CPU times to a maximum of 10%, and in increments of 10%, up to 100%, all while keeping the available memory to the VPSs at 100%.

3.3.1 Memory-Limited Configurations

After having simulated the page requests against each of the memory-limited VPSs, the values for the mean of the top twenty-five (25) CPU five-minute average loads and the mean load time for the homepage being served by the Magento-based sample store on each of the VPSs were calculated using the complete set of data in Appendix A and have been presented in Table 1 below.

Table 1. Load test results from memory-limited VPSs.

Memory Allocation	"vzplit -n" Value	Average 5-Minute Load	Average Time Per Request (ms)
10%	**10**	0.6548	308.237
11.11%	**9**	0.6	310.365
12.50%	**8**	0.6756	306.914
14.29%	**7**	0.696	317.942
16.67%	**6**	0.616	314.201
20%	**5**	0.6836	316.495
25%	**4**	0.6696	308.055
33.33%	**3**	0.614	308.733
50%	**2**	0.6584	306.455
100%	**1**	0.6284	302.473

For the most part, the average five-minute loads and the average times per page request in Table 1 only differ up to a maximum of approximately 13% and 5%, respectively. It can also be visually observed that the difference is quite small when graphing this data in Figure 2 below.

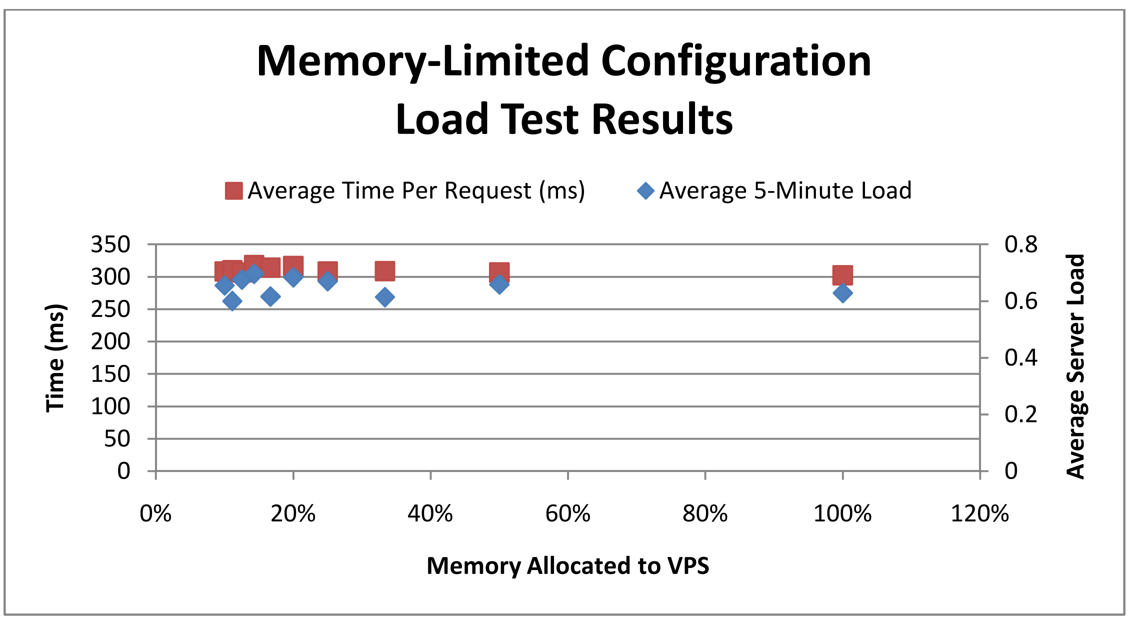

Figure 2. Graphed results of data obtained from Table 1.

Despite the fact that the page load times in Table 1 and Figure 2 only differ by about 5%, the actual observed time difference is within a few hundredths of a second which is still quite minuscule in reality; it is barely noticeable. The CPU loads in the same table and figure give the sense that not much other processing is being done on the server itself when changing memory configurations, so no noticeable difference will be seen by the end-user browsing to a Magento-based store or website running on a server with a minimal amount of memory.

3.3.2 CPU-Limited Configurations

After having simulated the page requests against each of the CPU-limited VPSs, the values for the mean of the top twenty-five (25) CPU five-minute average loads and the mean load time for the homepage being served by the Magento-based store on each of the VPSs were calculated using the complete set of data in Appendix B and have been presented in Table 2 below.

Table 2. Load test results from CPU-limited VPSs.

CPU Allocated	Average 5-Minute Load	Average Time Per Request (ms)
10%	2.574	2001.565
20%	1.892	986.347
30%	1.896	653.319
40%	1.55	491.076
50%	1.2064	394.676
60%	0.774	330.664
70%	0.6436	309.829
80%	0.6172	306.691
90%	0.5892	307.721
100%	0.6916	306.72

As was mentioned in 3.1, it was believed that limiting the CPU time would have the most drastic effects on performance. In this case, the average five-minute load differs by up to approximately 125% and the average time per page request by almost 147%. When the data from Table 2 is plotted on the graph in Figure 1, the corresponding data for both the average five-minute load and the average five-minute request almost resembles a power function, decreasing as CPU allocation is increased.

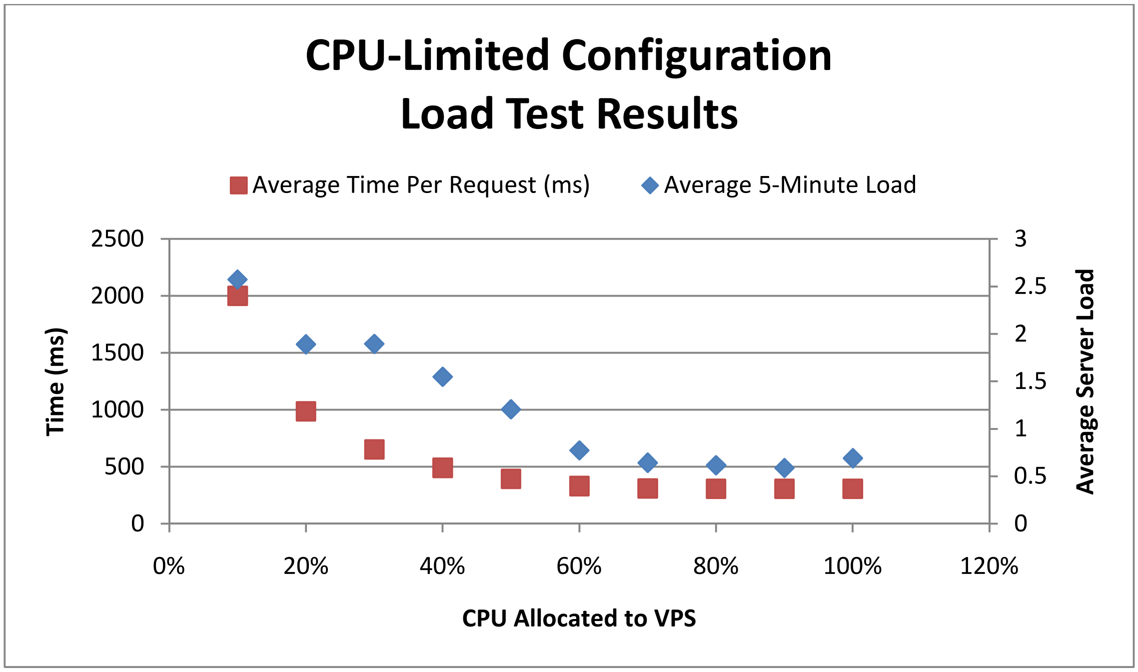

Figure 3. Graphed results of data obtained from Table 2.

Thus, from the data in Table 2, plotted in Figure 3, there is without a doubt a drastic change in server load and performance, as a result, of the Magento Commerce platform when CPU allocation is changed, as was expected.

3.4 Optimal VPS Configurations

With there being no difference in page load/request time or server load when limiting memory in a VPS hosting a Magento-based store or website, the memory on said VPS should be allocated at the discretion of Canfone's server administrators as the difference in load and request time is minimal. The secondary memory limit on PHP on the same VPS can also be changed at the discretion of the administrators all well knowing that there may be a possibility of increased memory usage as a result.

In terms of CPU, a minimum amount of 50% CPU time should be allocated to any one VPS hosting a Magento-based store or website. Said configuration may increase the page load/request time by approximately 25%, but in terms of time, would only represent an increase of a maximum of approximately nine one-hundredths of a second, which may be barely noticeable when other times, such as rendering times, are taken into account. This would most likely limit any one VPS from hosting no more than one or two Magento-based stores or websites.

These VPS configurations are sure to help Canfone provide their customers with the best Magento-based website hosting possible with minimal or no performance loss whatsoever. Although Canfone may be limiting their hardware to hosting only a few Magento-based websites, it will pay off in the long run for them and their customers when minimal troubleshooting will be needed due to performance-related issues.

4 Conclusions

From the analysis in the report body, it is concluded that limiting the memory allocated to a virtual private server has a minimal effect on the performance/page load/request time of Magento-based websites hosted on said server, as well as on the server load of the server itself. Because there is a secondary memory limit imposed by PHP itself, a virtual private server will never have all of the memory allocated to it be used up solely by PHP.

Lastly, limiting the amount of CPU time allocated to a virtual private server can have drastic effects depending on the percentage of CPU time allocated to any one virtual private server. The server load and the page load/request time of Magento-based websites can both be increased by up to 147% in some cases, drastically decreasing the performance of the server, the performance of the website, and an end-user's browsing experience.

5 Recommendations

Based on the analysis and conclusions put forth in this report, it is recommended that Canfone implement the following recommendations:

1) For ease of deployment, memory should be allocated to any one virtual private server at the discretion of the server administrator since the effect on server and website performance will be minimal or non-existent.
2) The secondary memory limit on PHP on any one VPS can also be changed at the discretion of the administrator, keeping in mind that there may be a possibility of increased memory usage on the server as a result.
3) Any one virtual private server should have at least 50% of its hardware node's CPU time allocated to it so as not to affect the performance of the server, the performance of the website, and an end-user's browsing experience.

By implementing one or more of the above recommendations, these VPS configurations are sure to help Canfone provide their customers with the best Magento-based website experience possible. Canfone will be able to benefit from minimal troubleshooting due to performance-related issues by limiting their hardware to hosting only a few Magento-based websites.

Glossary

Apache – In the context of this report, Apache refers to a web server, unless otherwise specified.

Gigabyte (GiB) – 1024^3 bytes.

Gigahertz (GHz) – 10^9 Hertz.

Hardware Node – A host computer or server on which OpenVZ is installed and running which allows virtual private servers to directly access its hardware.

Kernel – Connects applications to the hardware of a computer.

MySQL – A database language for managing data in a database.

OpenVZ – A container-based virtualization solution for Linux.

PHP – A hypertext preprocessor for interpreting and rendering coded web applications.

Server Load – For single-processor systems, the server load can be thought of as a percentage of system utilization over a period of time. For a system with multiple processors or processor cores, one must divide the number by the number of total processor cores, and it too represents system utilization over a period of time.

Virtual Private Server / VPS / Virtual Container – A server that runs virtually on top of a software layer on a physical piece of hardware.

References

[1] Canfone.com Web Services, “Company,” 2010. [Online]. Available: http://can-en.canfone.com/company/index.php. [Accessed: Apr. 23, 2010].

[2] Canfone.com Web Services, “Our History,” 2010. [Online]. Available: http://can-en.canfone.com/company/history.php. [Accessed: Apr. 23, 2010].

[3] Canfone.com Web Services, “Values,” 2010. [Online]. Available: http://can-en.canfone.com/company/values.php. [Accessed: Apr. 23, 2010].

[4] *Magento User Guide*, Varien Inc., Los Angeles, CA, 2009, pp. 11-14, 189

[5] *OpenVZ User’s Guide*, SWsoft, Inc., 2005, pp. 11-12, 27, 92, 114. [E-Book]. Available: http://download.openvz.org/doc/OpenVZ-Users-Guide.pdf. [Accessed: May 1, 2010].

[6] Magento, “System Requirements,” 2010. [Online]. Available: http://www.magentocommerce.com/system-requirements. [Accessed: May 1, 2010].

Appendix A – Memory-Limited Configuration Test Results

Average values, with minimum and maximum request times for 1000 page requests:

Memory Allocation	"vzplit -n" Value	Average 5-Minute Load	Average Time Per Request (ms)	Minimum Time Per Request (ms)	Maximum Time Per Request (ms)
10%	**10**	0.6548	308.237	285	698
11.11%	**9**	0.6	310.365	288	634
12.50%	**8**	0.6756	306.914	287	529
14.29%	**7**	0.696	317.942	284	681
16.67%	**6**	0.616	314.201	287	659
20%	**5**	0.6836	316.495	284	625
25%	**4**	0.6696	308.055	285	601
33.33%	**3**	0.614	308.733	286	488
50%	**2**	0.6584	306.455	285	560
100%	**1**	0.6284	302.473	283	532

Top 25 server load values for VPSs:

10%	11.11%	12.50%	14.29%	16.67%
0.72	0.67	0.74	0.78	0.69
0.71	0.66	0.73	0.77	0.67
0.69	0.65	0.72	0.75	0.66
0.69	0.64	0.72	0.74	0.65
0.69	0.63	0.71	0.72	0.64
0.68	0.62	0.71	0.72	0.64
0.67	0.62	0.7	0.71	0.64
0.67	0.62	0.69	0.71	0.63
0.66	0.61	0.69	0.71	0.63
0.66	0.61	0.69	0.7	0.62
0.66	0.6	0.68	0.7	0.62
0.66	0.6	0.68	0.7	0.62
0.65	0.6	0.67	0.69	0.61
0.65	0.59	0.67	0.68	0.61
0.64	0.59	0.66	0.68	0.6
0.64	0.58	0.66	0.68	0.6
0.64	0.58	0.66	0.68	0.6
0.64	0.58	0.65	0.68	0.59
0.63	0.57	0.65	0.67	0.59
0.63	0.57	0.64	0.67	0.59
0.62	0.57	0.64	0.66	0.58

0.62	0.57	0.64	0.65	0.58
0.62	0.56	0.64	0.65	0.58
0.62	0.56	0.63	0.65	0.58
0.61	0.55	0.62	0.65	0.58

20%	**25%**	**33.33%**	**50%**	**100%**
0.79	0.74	0.67	0.72	0.69
0.77	0.73	0.66	0.71	0.67
0.75	0.72	0.66	0.71	0.67
0.74	0.71	0.65	0.69	0.66
0.73	0.7	0.64	0.69	0.66
0.72	0.7	0.64	0.68	0.65
0.72	0.69	0.63	0.68	0.64
0.72	0.68	0.63	0.68	0.64
0.71	0.68	0.63	0.66	0.63
0.7	0.68	0.62	0.66	0.63
0.69	0.68	0.62	0.66	0.63
0.69	0.67	0.61	0.66	0.63
0.68	0.67	0.61	0.66	0.62
0.68	0.66	0.61	0.65	0.62
0.67	0.66	0.6	0.65	0.62
0.66	0.65	0.6	0.65	0.62
0.66	0.65	0.6	0.64	0.61
0.65	0.65	0.6	0.64	0.61
0.64	0.65	0.6	0.63	0.61
0.63	0.64	0.59	0.63	0.61
0.63	0.64	0.58	0.63	0.6
0.62	0.63	0.58	0.62	0.6
0.62	0.62	0.58	0.62	0.6
0.61	0.62	0.57	0.62	0.6
0.61	0.62	0.57	0.62	0.59

Appendix B – CPU-Limited Configuration Test Results

Average values, with minimum and maximum request times for 1000 page requests:

CPU Allocated	Average 5-Minute Load	Average Time Per Request (ms)	Minimum Time Per Request (ms)	Maximum Time Per Request (ms)
10%	2.574	2001.565	1428	3441
20%	1.892	986.347	952	1480
30%	1.896	653.319	524	1014
40%	1.55	491.076	413	797
50%	1.2064	394.676	343	667
60%	0.774	330.664	300	1027
70%	0.6436	309.829	288	508
80%	0.6172	306.691	286	456
90%	0.5892	307.721	286	453
100%	0.6916	306.72	283	613

Top 25 server load values for VPSs:

10%	20%	30%	40%	50%
2.67	2.08	2.07	1.79	1.36
2.66	2	2.06	1.75	1.33
2.65	1.99	2.06	1.72	1.31
2.64	1.99	2.06	1.68	1.28
2.63	1.99	1.99	1.64	1.26
2.63	1.97	1.98	1.62	1.26
2.62	1.92	1.98	1.61	1.22
2.62	1.91	1.98	1.59	1.22
2.61	1.9	1.98	1.58	1.2
2.6	1.89	1.98	1.57	1.2
2.6	1.88	1.91	1.57	1.2
2.59	1.87	1.9	1.57	1.19
2.58	1.86	1.9	1.56	1.19
2.57	1.86	1.89	1.54	1.18
2.57	1.85	1.83	1.53	1.17
2.56	1.85	1.82	1.49	1.17
2.56	1.85	1.81	1.49	1.17
2.53	1.85	1.8	1.45	1.17
2.52	1.84	1.8	1.45	1.16
2.51	1.83	1.78	1.44	1.16
2.5	1.83	1.78	1.44	1.16
2.5	1.83	1.76	1.43	1.16

2.48	1.82	1.76	1.42	1.15
2.48	1.82	1.76	1.41	1.15
2.47	1.82	1.76	1.41	1.14

60%	**70%**	**80%**	**90%**	**100%**
0.86	0.77	0.69	0.68	0.78
0.86	0.75	0.68	0.65	0.77
0.84	0.71	0.66	0.64	0.75
0.83	0.7	0.65	0.64	0.72
0.82	0.68	0.65	0.63	0.72
0.81	0.68	0.64	0.62	0.71
0.81	0.65	0.64	0.61	0.71
0.8	0.65	0.63	0.61	0.7
0.8	0.65	0.63	0.6	0.7
0.78	0.65	0.62	0.59	0.69
0.77	0.65	0.62	0.59	0.69
0.77	0.65	0.62	0.59	0.68
0.77	0.64	0.61	0.58	0.68
0.76	0.62	0.61	0.58	0.68
0.76	0.62	0.61	0.58	0.68
0.75	0.62	0.6	0.57	0.68
0.75	0.62	0.6	0.57	0.67
0.75	0.61	0.59	0.56	0.67
0.74	0.61	0.59	0.56	0.66
0.73	0.6	0.59	0.56	0.66
0.73	0.6	0.59	0.55	0.66
0.72	0.6	0.58	0.55	0.66
0.72	0.59	0.58	0.54	0.66
0.71	0.59	0.58	0.54	0.66
0.71	0.58	0.57	0.54	0.65