B3023886: Scientific review of web server technologies

Abstract

This essay is going to discuss modern web technologies from a scientific perspective. The outcomes of this essay will be an in depth review of web server technologies, how they are used and what features they have. We will also be reviewing how information is kept secure on a web server and why this is of utmost importance when implementing a server. Displaying of information is also important, and we shall look at the varying methods used for storing and displaying information from a server using both static and dynamic pages.

Lastly we shall briefly touch on the future of web servers and where current trends are headed. Where is server technology going to be in 10 years, and what are we expecting to see from new technologies and will this technology be more accessible or more restrictive?

Introduction

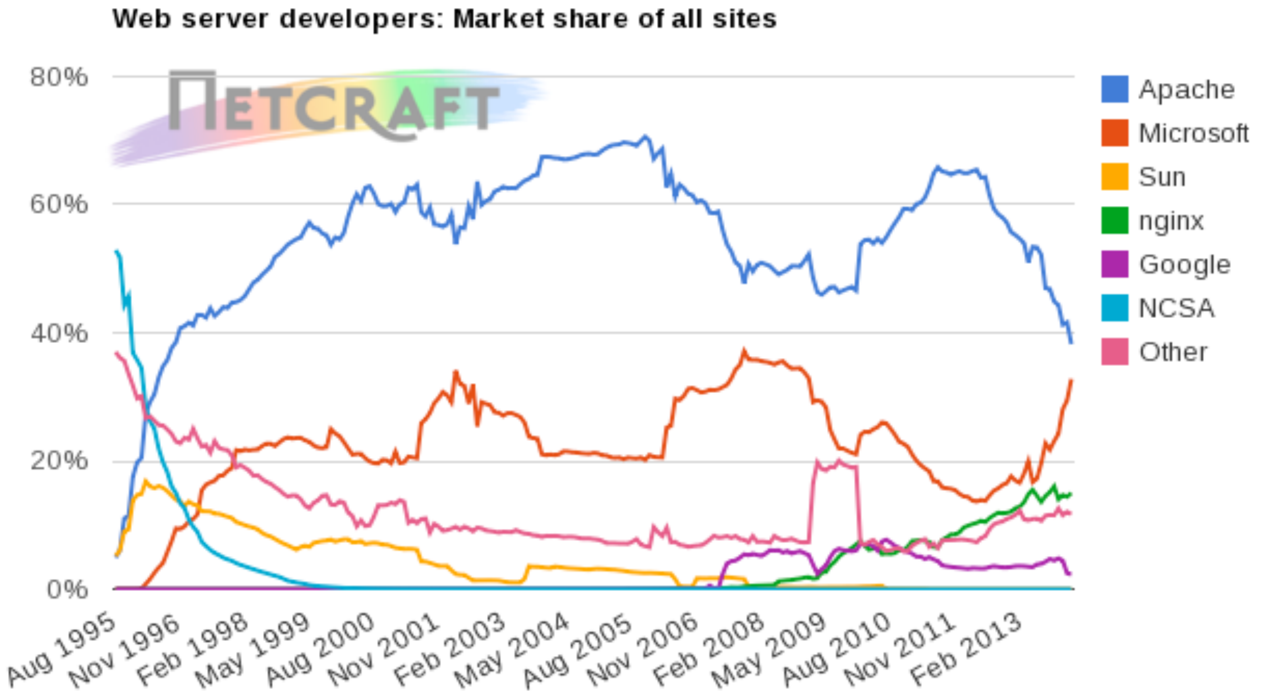
Different web servers are designed to do a multitude different things, and therefore must make use of many technologies to achieve multiple tasks asynchronously. Throughout the history of web server design emphasis has been put into creating a 'fit for all' solution to a problem, however with an emphasis on power output. Microsoft Server 2008 for example has a built in loader to make server which allows it to run different things at the same time. The open source Linux platforms such as CentOS, Debian and even Fedora[1] have opened up the chance for any machine to run asynchronous server profiles at once to include modules such as ssl encryption, php, and many other configurable profiles in server software such as apache.

This has seen a big development in accessible server hardware, and more recently access to online computing systems through services such as Amazon Web Services (AWS), offering access to hugely powerful servers for relatively affordable prices. And with the introduction of Wordpress in 2003 there was an insurgence of php, ruby rails and even python backends that are easy to use, and relatively easy to build. This increase in web content creators has created a culture of web developers that create content and distribute it as freeware, drastically improving the software available to site designers.

And not only have servers been used to create websites and database services, but they have been used to create virtual machines that can be used from anywhere, bitcoin rigs with dedicated mining processors, gaming consoles and multiplayer servers, and even by some as deep learning machines or security testing units.

Figure(1) shows the popularity of different server technologies from Aug 1995 to Feb 20143 Notice that apaches is the most widely used with a market share of almost 70% in 2008 compared to around 55% in 2014. Nginx has increased it's market share massively in this time however, fast becoming a rival to apache as it offers a very similar range of services as well as easy to code modules that has generated it a large following in the online open source community.

Figure 1: http://news.netcraft.com/archives/2014/02/03/february-2014-web-server-survey.html[4]



In future I expect that we will see a large spike in the use of 'other' technologies as there are many new backends such as Jetty, IIS and Varnish which offer easy to implement web servers that are extremely fast and versatile. I believe this will lead to a balancing market share as each creates it's own dedicated team of developers and users.

Feature Comparison

There are many important factors to consider when implementing a web server. Security is imperative, as well as expandability and speed. With so many websites built on a small number of platforms, when a security hole is found developers must respond immediately to avoid risk to user data. For example when a Wordpress exploit was discovered in April 2014 a mass amount of bots were built, targeting any Wordpress site and automatically DDosing the wp-login panel. This lead to web hosting services such as HeartInternet implementing extra security measures when accessing their custom apache control panels preventing the automated attack from accessing the files[2]. Wordpress promptly released an update preventing the sites from being affected, improving security and closing similar exploits at the same time.

The modules provided in an apache backend are feature rich and versatile. The mod_auth module provides a simple yet very secure authentication system for user logins, whilst the available sql plugin provides a secure database system with customisable secure data hashing algorithms. The mod_ssl module provides HTTPS security through use of an SSL Certificate from a recognised Certificate authority, and guarantees that the connection is securely encrypted using RSA.

See below for a breakdown of web server technologies and the types of features that they offer[3].

Server	Access Auth	CGI	SSI	Virtual Hosting
Apache	X	X	X	X
IBM	X	X	X	X
IIS	X	X	X	X
Jetty	X	X		X
nginx	X		X	X

This versatility has allowed developers to build web servers that do almost anything, providing reliable web services for very reasonable prices. Another important feature created by developers is ftp, a protocol that allows users to login to any remote server and read/write files to and from a local machine. Other protocols such as ssh allow for remote login to a web serve with superuser access, providing a shell for the user to use, running commands on the remote machine. Many power users now have a desktop at home with ssh or remote viewing capabilities, allowing access to applications on their own powerful build from anywhere.

In the future it is very likely that we will see the number and capability of these modules increase rapidly as new innovations are made in the ways that we store, display and digest information. With further increases in the ability and number of software developers, more and more freeware will be developed giving anyone access to this technology.

Discussion

As you can see from the above table, there are very few differences in the modules that are offered by different servers. One feature of a server is the language it is written in, for example php or ruby on rails. Groups of these languages and their respective features can be used in tandem with each other depending on the server setup and the feature cross compatibility.

As we move forward with available technology I expect there will be constant change in the most commonly used languages. In recent years specific JavaScript languages such as node.js & angular.js. Both of these Javascript languages are feature rich, with built in styling and server-side-scripting capabilities. Node.js is particularly interesting because it is not a client-side language but actually runs directly on the web server.

Citations

Citation	
1	Schroder, C. (2011, August 8). Linux.com. Retrieved December 18, 2015, from https://www.linux.com/learn/tutorials/479960:the-six-best-linux-community-server-distributions
2	Telfer, M. (2013, April 12). Important information for WordPress users - Web Hosting Blog. Retrieved December 18, 2015, from https://www.heartinternet.uk/blog/important-information-for-wordpress-users/
3	Comparison of Web servers. (2015, February 16). Retrieved December 18, 2015, from http://socialcompare.com/en/comparison/comparison-of-web-servers
4	February 2014 Web Server Survey. (2014, February 3). Retrieved December 18, 2015, from http://news.netcraft.com/archives/2014/02/03/february-2014-web-server-survey.html