

System Administration for Beginners

Week 7 Notes

October 27, 2008

1 Lecture

So far we have covered and mastered the basics of UNIX, the history of and how the Internet works, and different ways to obtain and install software, most notably server daemons. These topics together form a foundation for general UNIX system administration. Additionally, the labs and hands-on experience have been specifically focused on familiarizing oneself with commands and applying concepts learned. In the upcoming weeks, we will be building upon this with a focus on multi-user Internet server administration.

1.1 Internet Servers

Internet servers are the most common type of servers. Many companies have their own specialized internal servers that handle functions such as user authentication, but practically every company uses a web server to host their own Internet presence.

In general, Internet servers are composed of four components: an operating system, a way for users to interact with the server, a place to store user data, and a programming language to tie all the components together. In this class, we have been and will continue to use GNU/Linux as the operating system.

The most common way of allowing users to interact with a remote server is through a web server. Every user connected to the Internet has access to a web browser, and most people are familiar with browsing web pages and buying products online. Since companies want to make their Internet presence accessible to the largest audience, they generally use a web site. Other forms of allowing interaction with Internet users are mail servers and remote login interfaces, such as SSH.

The most popular web server daemon is Apache, but there are many other free and commercial daemons available. Of particular note is IIS, or Internet Information Server, produced by Microsoft; it is the second most popular web server, but is neither free nor open-source.

Interaction with customers produces a lot of data, such as purchase orders, shopping carts, and customer information, which must be stored in some form. Companies must also provide customers with easy access to data such as product information and availability. The most popular way of managing large

amounts of data is with a database. A generic database can be thought as a table with rows and columns. These databases are managed by server daemons that provide special features, such as sorting, fast updates and retrieval, and transactions. For smaller amounts of data, companies can use low-tech solutions such as simple text files, sometimes known as flat files.

Databases come in many forms, but most databases follow the Structured Query Language (SQL) standard. SQL is a language which is recognized across different platforms and software. There is a common set of commands that each SQL daemon recognize, so that knowing the basics of it will allow you to interact with most databases. The most commonly used SQL database, MySQL, is free and open-source. Other common SQL databases are PostgreSQL, Oracle, and Microsoft SQL Server.

Each of the aforementioned components operate independent of each other. If you install them on a server, they will typically operate by themselves and will not interact. Programming languages are used to create programs that tie the components of a server together. For a example, a PHP program might be used to produce a dynamic website for a company. Customers may use this web site to browse information on a company's products, which the program retrieves from a certain database. If a customer wishes to purchase a product, the program may accept their order and store it into a database for future processing.

PHP: Hypertext Processor (PHP) is the most popular Internet programming language. It is easy to learn and supports many different types of web servers and databases. Other popular Internet programming languages include Perl and Python.

As stated in the beginning of this course, we will not be doing any programming. However, we will be looking at the software that must be installed to allow Internet programs to run, and we will be installing some of these programs to demonstrate how all the components of a web server work together.

1.2 The LAMP Model

LAMP (Linux, Apache, MySQL, PHP) is a popular combination of open-source tools used to build Internet servers. Linux is used as the operating system, Apache as the web server daemon, MySQL as the database daemon, and PHP as the programming language. Together, these programs provide practically everything a company needs to develop and offer web services. Additionally, the 'P' in LAMP can mean Python or Perl, depending on your preferences.

We'll be focusing upon the LAMP package of server components in this course, but you'll find that many companies substitute other pieces of software depending on their own needs. For example, Microsoft Windows is commonly substituted for Linux in companies that have a Microsoft Windows infrastructure.

1.3 Choosing a Server Daemon

As a system administrator, the software you use will generally be mandated by the company that hires you. Unless you are working for a smaller or startup company, you will not have much choice or freedom regarding choice in software. However, if you have a choice, the following are important considerations to keep in mind:

1.3.1 Customers and Clients

The most important point to consider is what your customers and clients want to use. If they want to use an Oracle database, it would be best to provide them with an Oracle database. As is often said, the customer is king.

1.3.2 Personal Preference and Experience

If you are familiar or comfortable with a certain piece of software, it's probably best to use it. When something breaks, fixing things is a lot easier if you have some experience with the software.

1.3.3 Popularity and Support

Barring any personal preference, if you have to randomly choose a piece of software, choose the most popular one. Bugs are identified and fixed faster in software that many people are using, and you can be generally be assured that the software will be maintained. Many times programmers have released a server daemon and never bothered to update or fix the code when bugs were identified.

Another reason for using popular software is that there will generally be a diverse community of users. If you plan on using the software in a very specialized environment and need help, you're more likely to find someone with a similar configuration when using popular software.

1.3.4 Speed and Efficiency

In case of high-traffic servers or servers with limited resources, the speed and efficiency of different pieces of software can be very important. Some server daemons are not very scalable – they can not handle a very large amount of users. Others place too many demands upon the server and end up slowing everything else down. In this case, it is up to you as the system administrator to think of different solutions that would scale well or provide some sort of fail-safe support.