System Administration for Beginners

Week 2 Notes

February 8, 2010

1 Introduction to UNIX

1.1 What is UNIX?

UNIX used to refer to an operating system. The term and definition of "UNIX" has been used and defined rather loosely in the past, and still continues today. Formally, UNIX is a trademark held by The Open Group and refers to a system that complies with the "Single UNIX Specification". Ideally, developers of systems can call their product UNIX or UNIX-compliant if they follow the specification put forth originally by The Open Group and IEEE (now maintained by the Austin Group). If followed, the standard maintains that programs will be able to run on all UNIX systems without non-trivial modifications.

While many operating systems are registered as UNIX compliant, a number of them do not certify themselves as UNIX due to the time and costs required. For example, the trademark costs 2,500$USD with an additional 1,000$USD license fee every year thereafter. While not UNIX-compliant, these operating systems are generally referred to as Unix-variants or *nix. From here on out, UNIX will refer to systems compliant with the specification while Unix will refer to Unix-like systems or UNIX variants.

1.2 History of UNIX

UNIX was an operating system that was designed in the 60's at Bell Labs by a group of AT&T employees. It's in the same category as other operating systems as Microsoft Windows and Mac OS X. UNIX, however, is not a Windows or Mac OS X 'clone'. Mac OS X, for example, is actually derived from a Unix-like implementation called Berkeley Software Distribution (BSD).

UNIX turned out to be quite popular because it was well-designed and could be ported to various platforms, or different types of servers. It also had a set of very innovative features that no other operating system had at the time.

Unfortunately, UNIX was proprietary. It was like Microsoft Windows and Mac OS X today: you had to buy a license to use the software (as opposed to owning it), and you couldn't modify the software in any way. As a result, companies and universities began making UNIX-clones.

Eventually enough derivations and clones were developed that the name UNIX could no longer be applied to a single operating system, but to a category of operating systems. Several companies attempted to get together and form standards, which resulted in other groups of companies getting together to form rival standards, also known as the "Unix wars." In 1988, IEEE's POSIX specification[1] was published, creating a standard that was able to be implemented on different Unix-like platforms. With this standard in place, software written for one operating system that followed this standard could be used on another operating system without too much work in porting, or changing the program, the code to work with that system.

1.3 Unix/*nix Variants

From the mass of UNIX clones, the most popular and available ones now is Solaris, BSD, Windows, Mac OS, and Linux and their variants. Windows and Mac OS follow the UNIX's proprietary operating type. Solaris until recently, was proprietary but in recent years, they have been heading towards becoming a free software.

1.3.1 Solaris

Solaris is the descendent of SunOS, a Unix variant derived from BSD and has been proven to be a reliable and a powerful operating system for many companies today. It was once exclusive and specific to Sun Microsystem's SPARC computer architecture, they now support most of the major computer architecture platforms today. It is well known for it's scalability, allowing administrators to use Solaris in conjunction with powerful server hardware (*e.g.*, multiple central processing units and large amounts of memory). Administrators of the Solaris operating system are among the highest paid in industry.

1.3.2 BSD

BSD is the Berkeley Software Distribution, and was developed here at UC Berkeley in the 1970s. BSD's main benefit is that it is completely free, in both senses of the word; there is no license fee, and you can modify and redistribute it with little restrictions. Some of Microsoft Windows and Mac OS X can trace parts of their software to BSD. While BSD once referred to an operating system, it is now a term used to refer to it's descendants: FreeBSD, NetBSD, and OpenBSD.

1.3.3 GNU and Linux

Linux is a common name given to the variety of distributions and variations of systems out there. Formally, it is known as GNU/Linux as it represents the combination of the kernel (Linux) and it's tools (GNU system). For simplicity, many people refer to this combination as just Linux which has caused much

---

[1]IEEE Std 1003.1-1988

controversy. One should take note of when it is referencing the combination as opposed to just the kernel.

GNU is the set of utilities that comes with the system, such as file/directory creation, search, programming language compilation, etc. These originated from Richard Stallman. He is the founding member of the GNU Project (An effort to maintain and create new tools) and has helped shaped what are the core tools used in distributions today.

The Linux kernel serves as the heart of many systems which people refer to as Linux distributions today. Systems with the kernel are pseudo-UNIX clones of which the kernel is due to the efforts of a Finnish student by the name of Linus Torvalds (hence the name). Since it's creation, there have been many variants and overall usage has been rising in popularity in recent years. It departs from traditional UNIX design in many areas and has acquired a rapidly growing community of users and contributors. Like BSD, Linux is free to everyone, and anybody can modify it according to their needs (licensing, however, differs from distribution to distribution). You now find Linux in all sorts of devices, like PDAs, cell-phones, and routers. However, Linux generally sacrifices stability for flexibility, and, as a result, is not viewed as a very reliable UNIX platform (compared to Solaris). Nevertheless, Linux is favored by computer scientists and system administrators because of its rapid development and bleeding-edge (i.e., very up-to-date) feature set.

Popular variants are Ubuntu, Debian, Red Hat Linux, OpenSUSE, and Gentoo. There are many different ways for Linux to work, Ubuntu is a derivative of Debian, Red Hat Linux and OpenSUSE are owned by companies, Red Hat and Novell respectively. Gentoo is a source based distribution.

1.4 Why Use UNIX?

Superficially, UNIX isn't too different from Microsoft Windows or Mac OS X. Most UNIX clones have Graphical User Interfaces (GUI), and doing tasks like writing email or documents, browsing the internet, and chatting on instant messenger have little difference across operating systems.

Technically speaking, UNIX offers many advantages over Windows and OS X in regards to operating as a server, since UNIX was designed from the beginning to be used as a a server operating system: some of it's features aren't as friendly to a person new to the UNIX operating systems. It was designed with the system administrator in mind instead of the end-user. Things like the command line, and the sometimes lack of the familiar features that Windows and Mac OS X offer aren't always available in the same manner in UNIX.

Practically UNIX is preferred by many users because it's features allow it to be more secure, stable and faster than Windows or OS X as a server. User control in UNIX is fine grained, to such a point that you can specify exactly what a user can and can't do, unlike in OS X and Windows where there are generic and less-specific settings for users. Upgrades in UNIX systems do not require reboots; in fact, it is not uncommon for a UNIX system to be running for years. Depending upon the application, UNIX is also many times faster.

1.5   Why not to Use UNIX?

- If you depend on softare that run only on Windows or Mac OS X. Gamers and graphic artists are almost limited to these operating systems for their needs.

- Your hardware is not supported. Microsoft Windows is the world's most popular operating system, companies tends to design drivers for only Windows. However, in recent years with the rise of personal UNIX systems many companies has begun to provide drivers for UNIX. If not, there is a large user-driven community out there that will write drivers for your hardware.

- You don't like the command-line. Although tedious at times, the command-line is generally optimized for speed, with shortcuts and scripting languages available to automate repetitive commands and tasks. Back then, the lack of color and syntax highlighting in the terminal certainly was an eyesore. Now, most terminals and programs are provide support and capability for colors and highlighting, making the system administrator's job a lot easier.