UnixWorld Online: PC-Unix Connection Column No. 001

Selecting an OS for Internet services

By Tom Yager

Maxx.net--my personal network--is finally on-line. It was no cinch, but the process was more a test of my patience than my expertise. In coming articles, I'll be sharing with you the details of my entre into creating a well-rounded net presence. Just having finished the process leaves me in the mood to stage a Dennis Milleresque rant on the subject of choosing an operating system for Internet services.

Every magazine seems to be getting in on the net server act. I wouldn't be a bit surprised to find *Coal Age* running a twelve page treatise on its experiences setting up a bituminous Web server. As custom dictates, each rag preaches its pet operating system as the best one for the job. Of course I don't expect *Windows* magazine to advise its readers to grab a copy of Linux, but let's get real: UNIX is the only way to run an Internet server. Hey, Microsoft, are you listening?

Did you expect a columnist for UnixWorld Online to claim otherwise?

Why UNIX?

The reasons for UNIX's suitability as a server are obvious to most, but I must describe them lest the GUI camp accuse me of being a causeless UNIX dinosaur. My favorite reason for forsaking DOS, Windows, Windows NT and OS/2 as Internet servers? Filesystems. The structure of the DOS FAT filesystem, and those that model themselves after it, is one of the last remaining bits of outdated technology hampering advanced operating systems. Even NTFS and HPFS, Windows NT and OS/2 replacements for FAT, deliver better performance and long file names while clinging to FAT traditions and limitations.

Take drive lettering. All Windows and OS/2 filesystems use letters (A-Z) to identify mounted hard drives, network mounts and other random-access devices. Each drive letter has a hierarchy of directories and files beneath it, just like UNIX, but the difference is night and day. FAT and FAT-alikes make it difficult to manage storage, both local and remote. Applications get dependent on specific files existing on assigned drive letters. Users like me who employ removable drives and changing network configurations are forced to resort to contrivances. I've watched colleagues network mount drives on their own PCs to get around drive letter limitations.

What's more, FAT, NTFS and HPFS all lack support for links. You can create shadows and aliases in the graphical sense, but there's no equivalent for true UNIX links, hard or soft (symbolic).

Telemanagement

Once you put your system on-line for incoming Internet traffic, you're accountable. Unless your net presence is a mere hobby, you need to be prepared to minister to your system's ills no matter where you are. This presents a nearly insurmountable problem for users of GUI-centric operating systems. Remote control software lets you take control of your system's screen, keyboard and mouse through a modem, but it's slow and sometimes unreliable. On many trips, I've found flakey hotel phone systems limit modem traffic to 2400 baud. To make matters worse, many configuration changes to Windows and OS/2 systems require rebooting. TCP/IP parameters, for example, are locked up in a graphical configuration utility which, while attractive, lacks the capacity for on-the-fly changes that UNIX utilities like `ifconfig` provide. Imagine rebooting your Windows or OS/2 system through a modem link--talk about your Maalox moment!

Maxx.net's Internet link is a 28.8K dedicated PPP connection. If it fails, I can dial in with a slow modem and type ''pppcycle'' to force the system to reestablish the link. The Novell UnixWare operating system has cute point-and-click configuration utilities, too, but unlike Windows and OS/2, UnixWare stores pertinent data in ASCII files and manages the network with command-line utilities. It's a very remote-friendly environment.

The UNIX tools you can string together to make your system run also put UNIX solidly in the lead among Internet server environments. For example, on maxx.net I've got watchdogs that snoop for things like low disk space, crashed `inn` (Usenet news) and `httpd` (World Wide Web) servers, and emergency e-mail traffic. When it spots any of these conditions, my watchdog e-mails me and dials my numeric pager. Could you set up something like this in Windows or OS/2? Sure, but you'd be doing a lot of coding. Just wresting your modem from your fax server software long enough to dial your pager is a day's programming exercise. The latest Windows NT has an event manager that performs UNIX `cron`'s duty of launching applications at pre-arranged times, but be careful: if an application has to stop for input, it'll sit there until someone pulls up a keyboard and mouse. Most Windows and OS/2 apps are helpless when it comes to running unattended.

My UnixWare watchdogs are Korn Shell scripts, as are some of the BBS-style programs with which my dial-up users interact. Here's one area where OS/2 has it all over Windows: IBM's REXX interpreter holds its own against most UNIX shells as a utility language. Recent revisions have added sockets support; so you can create Internet applications entirely in REXX. Microsoft still dares to ship a server environment without a decent shell language.

Finally, cost is a factor which can't be brushed aside. You can set up a very capable Internet server on a $1000 computer with $50 of software. Any 486-class machine with 8 MB of RAM will do--get more RAM if you plan to run the X Window System--and grab a copy of NetBSD, FreeBSD or Linux. Any of these can be had for $50 or less, and Linux bundles now routinely include pre-compiled Internet server tools.

Commercial UNIXes, like my chosen UnixWare, are more costly but offer features free UNIX-alikes do not. I don't wish to start a range war with free UNIX users. It's a matter of individual choice, and some invest more confidence in a commercial product when business is riding on it. The helpful administrative tools and the Veritas journaling filesystem are but two factors that lead me to trust UnixWare for commercial work.

Windows NT and OS/2, running in reasonable server configurations, both need 16 MB of RAM to run. You'll need more if you want to do any development. With Windows' lack of a usable scripting language, a C/C++ compiler is a must. Add to that the cost of optional tools, including both graphical remote access and backup software. Windows NT does come with a decent graphical backup tool, but you’re on your own in Windows or Windows 95 unless you’re backing up to floppies or using a "floppy tape" drive.

It’s possible you’ll choose a UNIX variant that doesn’t have all the neat toys built in. In my case, UnixWare includes only basic TCP/IP servers (`telnetd, ftpd` and the like). Novell’s FTP site (ftp.novell.com) has both binaries and sources for most types of services you’ll want on-line. Linux has, by far, the most impressive and easily-accessible array of freeware Internet servers. It also has the distinction of being the only PC UNIX variant supported by Netscape’s unbeatable Web browser.

UNIX and UNIX-alikes do have a downside: for the non-UNIX crowd, the learning curve can be pretty intimidating. Fortunately, there are lots of good books on the subject. Take a trip to a local technical bookstore and check out the tutorial titles. Interestingly, you’ll find that the best books on setting up Internet services (like O’Reilly and Associates’ *Managing Internet Information Services*) focus on UNIX hosts.

Buy or build?

If you don’t want to take the time to learn, there’s an alternative: hire a wizard. Corporate downsizing and the move to Windows systems has left lots of UNIX-heads scrambling for things to do. Many of these folks have hung their own shingles, setting up Internet services for needy clients. The exploding popularity of the Internet could be alternately titled, ‘‘the full employment act for UNIX geeks.’’ Clearly, the fastest path to offering commercial data via Internet is to hire it done.

Sometimes it’s cheaper, too. My 28.8K link is perfect for my research, but were I planning to launch any commercial enterprise this bandwidth would prove wholly insufficient. The next step up in bandwidth is quite expensive by comparison. Corporations looking for exposure on the Web or who want their files made available through FTP are usually best served, at least to start, by purchasing space and services from Internet providers. The provider that maintains my link, FastLane Communications in Hurst, Texas, is typical. FastLane not only sells Internet links to those who want to set up their own machines. They also provide services, everything from storage space to HTML authoring, to clients who want to supply content and let someone else worry about its delivery. FastLane, like many providers, relies on a loose-knit association of specialists to service its clients.

Those with an inventor’s patience and a boundless drive to acquire new knowledge will find setting up an Internet server to be an exciting exercise. In the burgeoning electronic universe, as elsewhere, there are those who draw pleasure from consuming, and those who enthusiastically create that which is consumed. If you fall in the latter category, I’ll be providing information in coming columns which may help you along your way.

Don’t I know you?

Through the beneficence of my editor, Becca Thomas, I’m resuming the ‘‘PC-Unix Connection’’ column that ran during 1994 in *Open Computing* print magazine. I’m extremely pleased to be invited to participate in UnixWorld Online. I’m confident that--through this medium--I’ll be able to reach a unique core of bright and involved readers. As always, I’m interested in your comments. Send mail to tyager@maxx.net.

Edited by Becca Thomas / Online Editor / UnixWorld Online / beccat@wcmh.com

Go To Content
Search

Last Modified: Wednesday, 23-Aug-95 15:54:28 PDT