











































How does your OS compare?

	FreeBSD	vs. Linux	vs. Windows 2000
Reliability	 FreeBSD is extremely robust. There are numerous testimonials of active servers with uptimes measured in years. The new Soft Updates [*] file system optimizes disk I/O for high performance, yet still ensures reliability for transaction based applications, such as databases.	 Linux is well known for its reliability. Servers often stay up for years. However, disk I/O is non-synchronous by default, which is less reliable for transaction based operations, and can produce a corrupted filesystem after a system crash or power failure. But for the average user, Linux is a very dependable OS.	 All Windows users are familiar with the “Blue Screen of Death”. Poor reliability is one of the major drawbacks of Windows. Some of the major issues have been fixed in Windows 2000, but “code bloat” has introduced many more reliability problems. Windows 2000 uses a lot of system resources and it is very difficult to keep the system up for more than a couple of months without it reverting to a crawl as memory gets corrupted and filesystems fragmented.
Performance	 FreeBSD is the system of choice for high performance network applications. FreeBSD will outperform other systems when running on equivalent hardware. The largest and busiest public server on the Internet, at ftp.freebsd.com , uses FreeBSD to serve more than 1.2 terabyte/day of downloads. FreeBSD is used by Yahoo! , Qwest , and many others as their main server OS because of its ability to handle heavy network traffic with high performance and rock stable reliability.	 Linux performs well for most applications, however the performance is not optimal under heavy network load. The network performance of Linux is 20-30% below the capacity of FreeBSD running on the same hardware [*] . The situation has improved somewhat recently and the 2.4 release of the Linux kernel will introduce a new virtual memory system based on the same concepts as the FreeBSD VM system. Since both operating systems are open source, beneficial technologies are shared and for this reason the performance of Linux and FreeBSD is rapidly converging.	 Windows is adequate for routine desktop apps, but it is unable to handle heavy network loads. A few organizations try to make it work as an Internet server. For instance, barnesandnoble.com uses Windows NT, as can be verified by the error messages that their webserver produces, such as this recent example: Error Message: [Microsoft][ODBC SQL Server Driver][SQL Server] Can't allocate space for object 'queryHistory' in database 'web' because the 'default' segment is full. For their own “Hotmail” Internet servers, Microsoft used FreeBSD for many years.
Security	 FreeBSD has been the subject of a massive auditing project for several years. All of the critical system components have been checked and rechecked for security-related errors. The entire system is open source so the security of the system can and has been verified by third parties. A default FreeBSD installation has yet to be affected by a single CERT security advisory in 2000 [*] . FreeBSD also has the notion of kernel security levels, virtual server jails, capabilities, ACLs, and a very robust packet filtering firewall system. FreeBSD includes a very robust packet filtering firewall system and many intrusion detection tools.	 The open source nature of Linux allows anyone to inspect the security of the code and make changes, but in reality the Linux codebase is modified too rapidly by inexperienced programmers. There is no formal code review policy and for this reason Linux has been susceptible to nearly every UNIX-based CERT advisory of the year. This problem is compounded by the fact that distributions like Red Hat tend to turn on notoriously insecure services by default. However, Linux does include a very robust packet filtering firewall system and a competent administrator can remove unsafe services.	 Microsoft claims that their products are secure. But they offer no guarantee, and their software is not available for inspection or peer review. Since Windows is closed source there is no way for users to fix or diagnose any of the security compromises that are regularly published about Microsoft systems. Microsoft Windows has been affected by a very large number of known security holes that have cost companies millions of dollars [*] .
Filesystem	 FreeBSD uses the Berkeley Fast Filesystem, which is a little more complex than Linux's ext2. It offers a better way to insure filesystem data integrity, mainly with the “softupdates” option. This option decreases synchronous I/O and increases asynchronous I/O because writes to a FFS filesystem aren't synced on a sector basis but according to the filesystem structure. This ensures that the filesystem is always coherent between two updates. The FreeBSD filesystem also supports file flags, which can stop a would-be intruder dead in his or her tracks. There are several flags that you can add to a file such as the immutable flag. The immutable (schg) flag won't allow any alteration to the file or directory unless you remove it. Other very handy flags are append only (sappnd), cannot delete (sunlnk), and archive (arch). When you combine these with the kernel security level option, you have a very impenetrable system.	 The Linux ext2 filesystem gets its performance from having an asynchronous mount. You can mount FreeBSD FFS filesystems as asynchronous but this is very dangerous and no seasoned UNIX admin would do this. It's amazing that Linux is designed this way by default. Often a hard crash permanently damages a mount. FreeBSD or Solaris can sustain a very hard crash with only minor data loss, and the filesystem will be remountable with few problems. There are several new journaling filesystems in development for Linux that will fix some of these issues, but these will not be ready for the 2.4 release of Linux.	 The Microsoft FAT filesystem and the newer NTFS are both plagued by over 15 years of backwards compatibility with the earliest of PC-based filesystems. These filesystems were not designed for today's demanding server applications, they weren't even designed with a multi-user OS or networking in mind!
Device Drivers	 The FreeBSD bootloader can load binary drivers at boot-time. This allows third-party driver manufacturers to distribute binary-only driver modules that can be loaded into any FreeBSD system. Due to the open-source nature of FreeBSD, it is very easy to develop device drivers for new hardware. Unfortunately, most device-manufacturers will only release binaries for Microsoft operating systems. This means that it can take several months after a hardware device has hit the market until a device driver is available.	 The Linux community intentionally makes it difficult for hardware manufacturers to release binary-only drivers. This is meant to encourage hardware manufacturers to develop open-source device drivers. Unfortunately most vendors have been unwilling to release the source for their drivers so it is very difficult for Linux users to use vendor supplied drivers at all.	 Microsoft has excellent relationships with hardware vendors. There are often conflicts when using a device driver on different versions of Microsoft Windows, but overall Windows users have excellent access to third party device drivers.
Commercial Applications	 The number of commercial applications for FreeBSD is growing rapidly, but is still below what is available for Windows. In addition to native applications, FreeBSD can also run programs compiled for Linux, SCO UNIX, and BSD/OS.	 Many new commercial applications are available for Linux, and more are being developed. Unfortunately, Linux can only run binaries that are specifically compiled for Linux. It is unable to run programs compiled for FreeBSD, SCO UNIX, or other popular operating systems without significant effort.	 There are thousands of applications available for Windows, far more than for any other OS. Nearly all commercial desktop applications run on Windows, and many of them are only available on Windows. If you have an important application that only runs on Windows, then you may have no choice but to run Windows.
Free Applications	 There are many, many gigabytes of FREE software available for FreeBSD. FreeBSD includes thousands of software packages and an extensive ports collection, all with complete source code. Many people consider the FreeBSD Ports collection [*] to be the most accessible and easiest to use library of free software packages available anywhere.	 There are huge numbers of free programs available for Linux. All GNU software runs on both Linux and FreeBSD without modification. Some of the free programs for Linux differ between distributions, because Linux does not have a central ports collection.	 The amount of free Windows software is much less than what is available for UNIX. Many Windows applications are provided as “shareware”, without source code, so the programs cannot be customized, debugged, improved, or extended by the user.
Development Environment	 FreeBSD includes an extensive collection of development tools. You get a complete C/C++ development system (editor, compiler, debugger, profiler, etc.) and powerful UNIX development tools for Java, HTTP, Perl, Python, Tcl/Tk, Awk, Sed, etc. All of these are free, and are included in the basic FreeBSD installation. All come with full source code.	 Linux includes all the same development tools as FreeBSD, with compilers and interpreters for every common programming language, all the GNU programs, including the powerful GNU C/C++ Compiler, Emacs editor, and GDB debugger. Unfortunately due to the very splintered nature of Linux, applications that you compile on one system (Red Hat 7) may not work on another Linux system (Slackware).	 Very few development tools are included with Windows 2000. Most need to be purchased separately, and are rarely compatible with each other.
Development Infrastructure	 FreeBSD is an advanced BSD UNIX operating system. The source code for the entire system is available in a centralized source code repository running under CVS. A large team (200+) of senior developers has write access to this repository and they coordinate development by reviewing and committing the best changes of the development community at large. FreeBSD is engineered to find elegant solutions for overall goals, rather than quick hacks to add new functionality. Since FreeBSD is a complete open-source operating system, rather than just a kernel, you can recompile and reinstall the entire system by simply typing one command, ‘make world’.	 Linux is a UNIX-like kernel that must be combined with the GNU system to make a complete operating system. Linux does not use any version control system so all bug-fixes and enhancements must be emailed back and forth on mailing lists and ultimately submitted to the one person (Linus) who has authority to commit the code to the tree. Due to the overwhelming amount of code that gets written, it is impossible for one person to adequately quality control all of the pending changes. For this reason there is a lot of code in Linux that was hastily written and would never have been accepted into a more conservative operating system.	 Microsoft Windows is a closed-source operating system driven by market demand rather than technical merit. New technologies are rushed into the product before they have been properly designed or fully implemented. Very little is known about the internal development infrastructure of Microsoft but the “blue-screen of death” speaks for itself.
Support	 Several organizations, including BSDi, offer a wide range of support options for FreeBSD. In addition to 24x7 professional support, there is a large amount of free, informal support available through Usenet newsgroups and mailing lists, such as questions@freebsd.org . Once a problem is found, source code patches are often available within a few hours.	 Many organizations provide professional support for Linux. All the major Linux vendors offer some level of support, and several offer full 24x7 service. There are many forums where Linux questions are answered for free, such as newsgroups and mailing lists. As a last resort, you can always use the source to track down and fix a problem yourself.	 Although support is available for Windows 2000, you should be prepared to spend as long as an hour on hold, with no guarantee that your problem will be resolved. Because of the closed source nature of Windows, there is no informal, free support available, and bugs are fixed on Microsoft's schedule, not yours. Since Win 2000 is not updated frequently, you may wait years for bugs to be fixed.
Price and Total Cost of Ownership	 FreeBSD can be downloaded from the Internet for FREE . Or it can be purchased on a four CDRom set, along with several gigabytes of applications, for \$40. All necessary documentation is included. Support is available for free or for very low cost. There is no user licensing, so you can quickly bring additional computers online. This all adds up to a very low total cost of ownership.	 Linux is FREE. Several companies offer commercial aggregations at a very low cost. Applications and Documentation is available for little or no cost. There are no licensing restrictions, so Linux can be installed on as many systems as you like for no additional cost. Linux's total cost of ownership is very low.	 The server edition of Windows 2000 costs nearly \$700. Even basic applications cost extra. Users often spend many thousands of dollars for programs that are included for free with Linux or FreeBSD. Documentation is expensive, and very little on-line documentation is provided. A license is required for every computer, which means delays and administrative overhead. The initial learning curve for simple administration tasks is smaller than with UNIX, but it also requires a lot more work to keep the system running with any significant work load.
T O T A L	 = 9  = 2  = 0	 = 4  = 4  = 3	 = 2  = 1  = 8

For details on the footnotes, please see the reverse side of this document.

Footnotes

1. Filesystem

FreeBSD uses FFS, the Berkeley Fast File System, with the addition of “Soft Updates” for performance and consistency. FreeBSD 5.0-CURRENT (the development branch), gives users the ability to automatically snapshot file systems, as well as the ability to store extended attributes on files, in turn supporting other features such as Access Control Lists (ACLs). Fsch-less booting is currently a work in progress, but will be included with 5.0-RELEASE. In fact, several sites, including Yahoo!, have already deployed fschless booting internally. A paper titled “Journaling Versus Soft Updates: Asynchronous Meta-data Protection in File Systems” presented at the USENIX 2000 Technical Conference discusses the performance and consistency differences between journaled and soft updates consistency mechanisms. This paper is available online from <http://www.ece.cmu.edu/~ganger/papers/usenix2000.ps>. This paper also discusses two different journaling implementations based on FFS and developed on FreeBSD. Clearly, FreeBSD is at the forefront of filesystem research and this is the source of many of its performance and reliability advantages.

For more information about Soft Updates, please see:

“Soft Updates: A Technique for Eliminating Most Synchronous Writes in the Fast Filesystem” by Marshall Kirk McKusick and Gregory R. Ganger.
<http://www.ece.cmu.edu/~ganger/papers/mckusick99.ps>

2. Performance Benchmarks

Numerous benchmarks have continued to show FreeBSD’s clear advantage for network performance. **Yahoo!**, **Xoom.com**, **QWest**, and some of our other largest customers have published results showing the clear case for using FreeBSD in the enterprise. It’s important to understand that benchmarking is just a game, and that for real performance comparisons you need to perform real-world tests. However, it’s easy to find published benchmarks on the Internet which show FreeBSD with a commanding lead over the competition :

Gartner Group Comparisons showing FreeBSD 20-30% faster than Linux on identical hardware: http://advisor.gartner.com/n_inbox/hotcontent/hc_2121999_3.html#h8

Filesystem Benchmarking with PostMark from Network Appliance:
<http://www.shub-internet.org/brad/FreeBSD/postmark.html>

Is FreeBSD a Superior Server Platform to Linux? by Nathan Boeger:
<http://www.webtechniques.com/archives/2001/01/infrevu/>

“Flexibility, in-house expertise, price/performance, and manageability,” says Filo. “Those are just a few of the reasons why the Intel Architecture and FreeBSD appeal to us. By combining these platforms, we’re able to deliver better, faster and more innovative solutions than our competitors.” — David Filo, Chief Yahoo
<http://www.intel.com/ebusiness/casestudies/yahoo/buscase.htm>

3. Security

The Computer Emergency Response Team, or CERT, studies Internet security vulnerability, provides incident response services to sites that have been the victims of attack, publishes a variety of security alerts, does research in wide-area-networked computing, and develops information and training to help improve security at Internet sites.

CERT Advisories in 2000 that affected Linux:

CA-2000-22 — Input Validation Problems in LPRng
CA-2000-21 — Denial-of-Service Vulnerability in TCP/IP Stacks
CA-2000-20 — Multiple Denial-of-Service Problems in ISC BIND
CA-2000-17 — Input Validation Problem in rpc.statd
CA-2000-13 — Two Input Validation Problems In FTPD
CA-2000-06 — Multiple Buffer Overflows in Kerberos Authenticated Services
CA-2000-03 — Continuing Compromises of DNS servers

CERT Advisories in 2000 that affected Windows:

CA-2000-16 — Microsoft ‘IE Script’/Access/OBJECT Tag Vulnerability
CA-2000-14 — Microsoft Outlook and Outlook Express Cache Bypass Vulnerability
CA-2000-12 — HHCtrl ActiveX Control Allows Local Files to be Executed
CA-2000-10 — Inconsistent Warning Messages in Internet Explorer
CA-2000-07 — Microsoft Office 2000 UA ActiveX Control Incorrectly Marked “Safe for Scripting”
CA-2000-04 — Love Letter Worm

For more information about CERT and potential security exploits for your operating system, please see <http://www.cert.org>.

For more information about some of the enhanced security features of FreeBSD, please see <http://www.trustedbsd.org>.

4. Ports Collection

Even with all the hype about open standards, getting a program to compile on various UNIX platforms can be a tricky task. Occasionally, you might be lucky enough to find that the program you want compiles cleanly on your system, install everything into all the right directories, and run flawlessly “out-of-the-box”, but this behavior is somewhat rare. Most of the time, you find yourself needing to make modifications in order to get the program to work. This is where the FreeBSD Ports collection comes to the rescue.

The general idea behind the Ports collection is to eliminate all of the messy steps involved with making things work properly so that the installation is simple and very painless. With the Ports collection, all of the hard work has already been done for you, and you are able to install any of the Ports collection ports by simply typing ‘make install’.

The ports collection provides a makefile skeleton that describes where to download the software and how to compile and install it. There are currently over 4,400 applications in the FreeBSD Ports tree and with a single command the ports mechanism will automatically download the software source code, perform a checksum, uncompress the software, apply any FreeBSD-specific patches, configure the software, run the compilation, install the software, and clean up after itself! Ports can of course depend on other ports which are built automatically in the same fashion and binary packages can be used as well. The FreeBSD Ports Collection was recognized very early on as an elegant method to deal with a complex problem so it’s functionality has been shared with the other BSD UNIX systems.