Operating System

TAE - 1

Case Study on FreeBSD

-Shivam Tawari (A-49)

About the FreeBSD Operating System:

FreeBSD is a free and open-source Unix-like operating system descended from the Berkeley Software Distribution (BSD), which was based on Research Unix. The first version of FreeBSD was released in 1993. In 2005, FreeBSD was the most popular open-source BSD operating system, accounting for more than three-quarters of all installed simply, permissively licensed BSD systems.

Type of Operating System:

FreeBSD can be installed on a regular desktop or a laptop. A number of desktop environments such as GNOME, KDE, and Xfce, as well as lightweight window managers such as Openbox, Fluxbox, and dwm, bspwm, are also available to FreeBSD.

As of FreeBSD 12, support for a modern graphics stack is available via drm-kmod. A large number of wireless adapters are supported.

A FreeBSD installation requires a minimum of 96 MB of RAM and 1.5 GB of free hard drive space. However, such small amounts of memory and disk space are really only suitable for custom applications like embedded appliances. General-purpose desktop systems need more resources. 2-4 GB RAM and at least 8 GB hard drive space is a good starting point.

These are the processor requirements for each architecture:

amd64

This is the most common desktop and laptop processor type, used in most modern systems. Intel calls it Intel64. Other manufacturers sometimes call it x86-64.

Examples of amd64 compatible processors
include: AMD Athlon™64, AMD Opteron™, multi-core Intel Xeon™,
and Intel Core™ 2 and later processors.

i386

Older desktops and laptops often use this 32-bit, x86 architecture.Almost all i386-compatible processors with a floating point unit are supported. All Intel processors 486 or higher are supported.

powerpc

All New World ROM Apple Mac systems with built-in USB are supported. SMP is supported on machines with multiple CPUs. A 32-bit kernel can only use the first 2 GB of RAM.

Main Features of the FreeBSD Operating System:

FreeBSD's focus on performance, networking, and storage combines with ease of system administration and comprehensive documentation to realize the full potential of any computer.

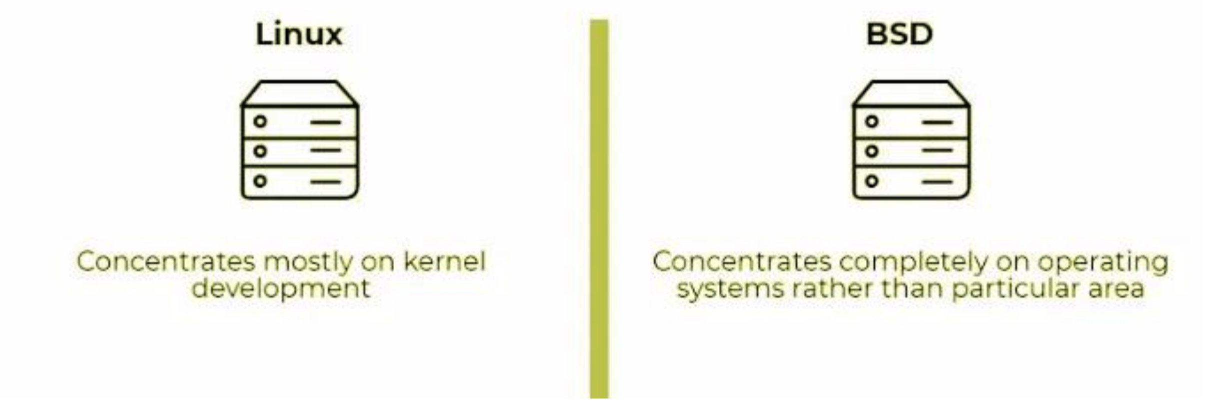
FreeBSD includes a number of other great features:

- ***Firewalls:*** The base system includes IPFW and IPFilter, as well as a modified version of the popular pf with improved SMP performance. IPFW also includes the dummynet feature, allowing network administrators to simulate adverse network conditions, including latency, jitter, packet loss and limited bandwidth.
- ***Jails*** are a light-weight alternative to virtualization. Allowing processes to be restricted to a namespace with access only to the file systems and network addresses assigned to that namespace. Jails are also Hierarchical, allowing jails-within-jails.
- ***Linux*** emulation provides a system call translation layer that allows unmodified Linux binaries to be run on FreeBSD systems.
- ***DTrace*** provides a comprehensive framework for tracing and troubleshooting kernel and application performance issues while under live load.
- ***The Ports Collection*** is a set of more than 23,000 third party applications that can be easily installed and run on FreeBSD. The ports architecture also allows for easy customization of the compile time options of many of the applications.

- ***Network Virtualization:*** A container ("vimage") has been implemented, extending the FreeBSD kernel to maintain multiple independent instances of networking state. Vimage facilities can be used independently to create fully virtualized network topologies, and jail(8) can directly take advantage of a fully virtualized network stack.
- ***New Binary Packaging System:*** FreeBSD now uses pkg, a vastly improved package management system that supports multiple repositories, signed packages, and safe upgrades. The improved system is combined with more frequent official package builds for all supported platforms and a new stable branch of the ports tree for better long term support.
- ***Unmapped I/O:*** The newly implemented concept of unmapped VMIO buffers eliminates the need to perform costly TLB shootdowns for buffer creation and reuse, reducing system CPU time by up to 25-30% on large SMP machines under heavy I/O load.
- ***bhyve:*** A new BSD licensed, legacy-free hypervisor has been imported to the FreeBSD base system. It is currently able to run all supported versions of FreeBSD, and with the help of the grub-bhyve port, OpenBSD and Linux.

Difference with another Operating system of the same type:

1.


Linux
BSD
Concentrates mostly on kernel development
Concentrates completely on operating systems rather than particular area

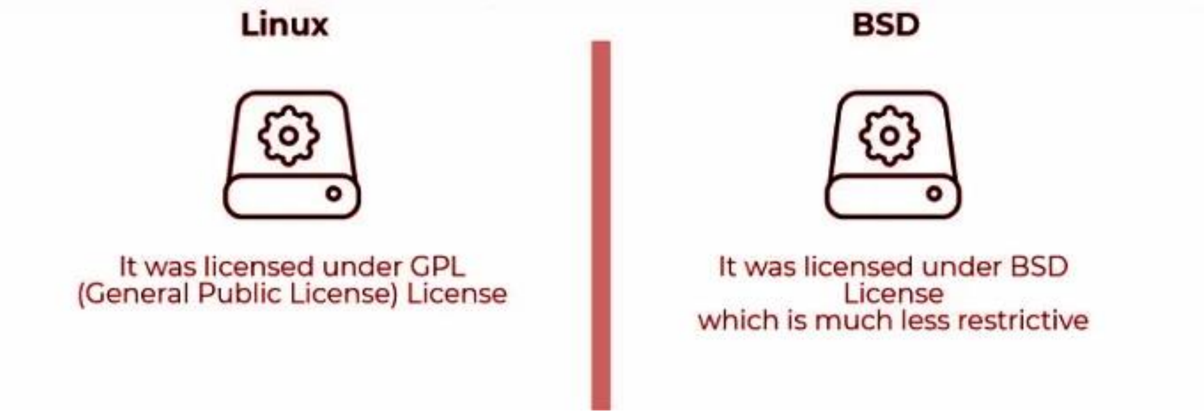
BSDs use something called a ports system. This system is what allows the installation of software packages. The software is kept in source form, meaning that your computer will need to compile the data each time before the software will run.

A silver lining in this is that packages can be installed in a pre-installed binary state which allows your system to forego the pre-run data compilation step.

The basic difference between both is that Linux distributions come with different sets of programs and repositories, allowing the user to download additional different programs pertaining to the requirements of the distribution.

When you install a BSD operating system, you just get the programs that BSD offers. This is not true for software packages as they are available to both as you will discover.

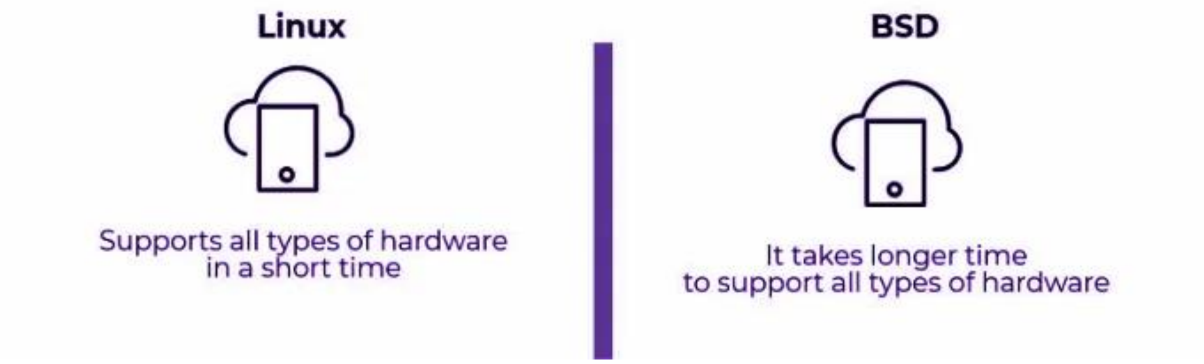
2.


Linux
BSD
It was licensed under GPL (General Public License) License
It was licensed under BSD License which is much less restrictive

Most people may not care but the difference in licensing is actually significant. Linux uses the GNU General Public License, or GPL. This means that developers can modify or add new features to the Linux kernel as they please. The only catch being that all newly-developed source code must be released to the public whether they want it to or not.

BSDs use their own unique BSD license which allows developers to modify and add new features to either the BSD kernel or distribution, without the need to release the source code. This means an open-source BSD can be declared closed-source if the developer so chooses. They have no obligation to release the source code to anyone.

3.


Linux
BSD
Supports all types of hardware in a short time
It takes longer time to support all types of hardware

Where Linux is concerned, it is easier for developers to write code that can be made available to users in pre-compiled binary packages for installation. The packages can be installed using apt, yum and other similar package managers. The open-source nature of Linux is what makes this possibility easier.

For BSD users, the task is not as simple. Users will have to download the source codes for the programs from the thousands of ports available to them. Then, after the source codes have been downloaded, they would have to compile them on their system.

This creates a headache for both BSD users and developers, as the lack of popularity among general users can be attributed to the extra hassle of compiling the source codes. The pre-compiled binary packages can be seen as the only saving grace to eradicate the hassle but still falls short in the availability of applications programs.