Istanbul Kemerburgaz University

UNIX FreeBSD CPU Scheduling

Student Name:
Alaa Firas Jasim

Student NO:
163101031

Prof.Dr. Hasan Hussien Balik

OUTLINE:

1. Introduction
2. History
3. Features
4. Development
5. Logo
6. Derivatives
7. Summary
8. References

1. Introduction:

CPU Scheduling

CPU scheduling is a process which allows one process to use the CPU while the execution of another process is on hold(in waiting state) due to unavailability of any resource like I/O etc, thereby making full use of CPU. The aim of CPU scheduling is to make the system efficient, fast and fair.

FreeBSD

Is a free and open source Unix _ like operating system descended from research Unix via the Berkeley Software Distribution (BSD). Although for legal reasons FreeBSD cannot use the Unix trademark, it is a direct descendant of BSD, which was historically also called "BSD Unix" or "Berkeley Unix".

The first version of FreeBSD was released in 1993, and today FreeBSD is the most widely used open_ source BSD distribution, accounting for more than three-quarters of all installed systems running open-source BSD derivatives.

FreeBSD has similarities with Linux, with two major differences in scope and licensing.

FreeBSD's code base has become an integral part of other operating systems, such as Juniper JUNOS, Apple's macOS, the Nintendo Switch system software, and the operating systems running onSony's PlayStation 3 and PlayStation .

2. History

Background:

FreeBSD's roots go back to the University of California, Berkeley. The university acquired a UNIX source license from AT&T

Birth of FreeBSD

The first version of FreeBSD was released on November 1993 .

3. Features

1. Uses:

As a general purpose operating system, FreeBSD is used in various scenarios.

Servers: use FreeBSD as a mail server, web server, Firewall, FTP server, DNS server and a router

Desktop: Although FreeBSD does not install the X Window System by default, it is available in the FreeBSD ports collection. A number of Desktop environments such as GNOME, KDE and Xfce, and lightweight window managers such as Open box, Flux box and DWM are available to FreeBSD.

Embedded systems: FreeBSD supports such as ARM, PowerPC and MIPS to a lesser degree.

2. Networking:

FreeBSD's TCP/IP stack is based on the 4.2BSD implementation of TCP/IP which greatly contributed to the widespread adoption of these protocols. FreeBSD also supports IPv6, SCTP, IPSEC, and wireless networking (Wi-Fi). Also, FreeBSD supports IPX and AppleTalk protocols, but they are considered old and it is planned to drop support of them in the upcoming FreeBSD 11.0

3. Storage:

FreeBSD has several unique features related to storage. Soft updates can protect the consistency of the UFS file system (widely used on the BSDs) in the event of a system crash.

4. Security:

FreeBSD provides several security-related features including access control lists (ACLs), security event auditing, extended file system attributes, mandatory access controls (MAC) and fine-grained capabilities. These security enhancements were developed by the **Trusted BSD** project.

5. Portability:

FreeBSD has been ported to a variety of instruction set architectures.

The FreeBSD project organizes architectures into tiers that characterize the level of support provided. Tier 1 architectures are mature and fully supported.

Tier 2 architectures are undergoing major development. Tier 3 architectures are experimental or are no longer under active development and Tier 4 architectures have no support at all.

6. Third-party software:

Free BSD has a software repository of over 26,000 applications that are developed by third parties.

Examples include: windowing systems, web browsers, email clients, office suites

7. Jails:

First introduced in FreeBSD version 4, jails is a security mechanism and an implementation of operating-system-level virtualization that enables the user to run multiple instances of a guest operating system on top of a FreeBSD host.

8. Virtualization:

BHYVE, a new virtualization solution was introduced in FreeBSD 10.0. BHYVE allows a user to run a number of guest operating systems (FreeBSD, Open BSD, Linux, and Microsoft Windows) in a same time.

9. OS compatibility layers:

Free BSD provides compatibility layers for several other Unix-like operating systems, in addition to Linux, such as BSD/OS and SVR4,however, it is more common for users to compile those programs directly on FreeBSD.

10. Kernel:

FreeBSD's kernel provides support for some essential tasks such as managing processes, communication, booting and file systems. FreeBSD has a monolithic kernel, with modular design.

11. Documentation and support:

FreeBSD's documentation consists of its handbooks, manual pages, mailing list archives, FAQs and a variety of articles, mainly maintained by The FreeBSD Documentation Project.

12. Installers:

From version 2.0 to 9.0, FreeBSD used the sysinstall program as its main installer.

4. Development

Free BSD is developed by a volunteer team located around the world.

Governance structure: The FreeBSD Project is run by around 500 committers, or developers who have commit access to the master source code repositories and can develop.

Branches: FreeBSD developers maintain at least two branches of a same time development. The -CURRENT branch always represents the "bleeding edge" of FreeBSD development. A -STABLE branch of FreeBSD is created for each major version number.

5. Logo

Free BSD's logo was the generic BSD daemon, also called Beastie.

6. *Derivatives*

***There are a number of software distributions based on FreeBSD including:**

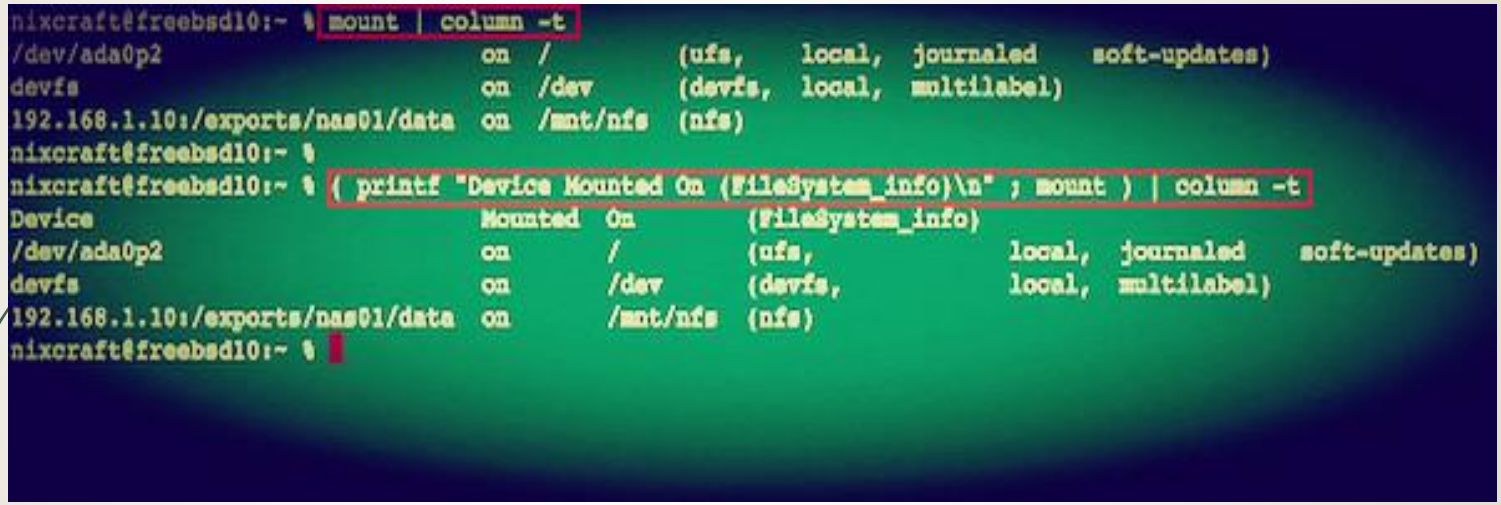
((Open Server 10, True OS, Desktop BSD, Ghost BSD, Free SBIE, Frenzy, Hardened BSD, m0n0wall, pf Sense, Free NAS, NAS4Free)).

***Some subscription services that are directly based on FreeBSD are:**

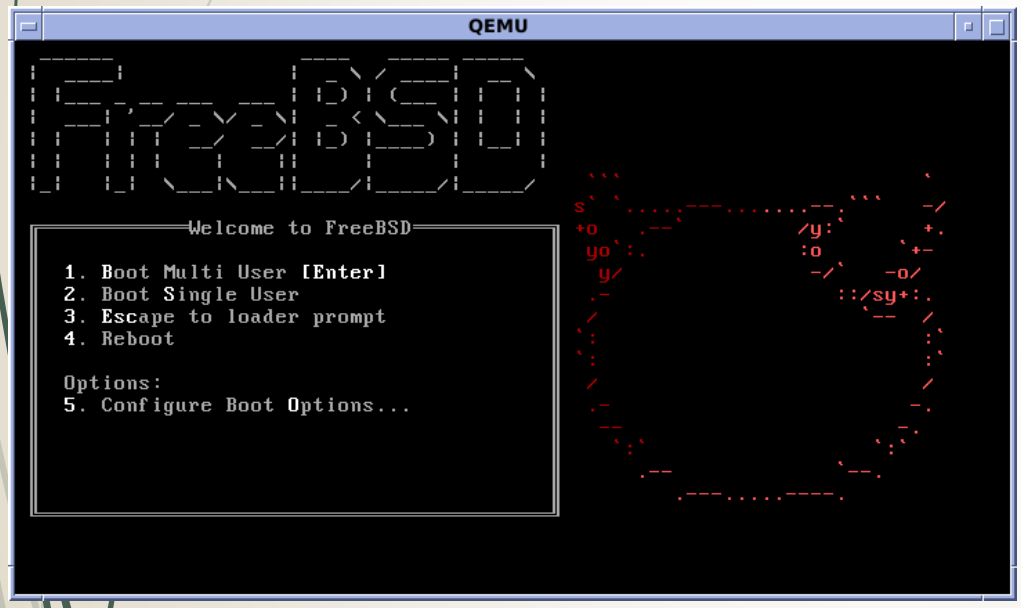
((Whats App, EMC Isilon, NetApp, Netflix, PlayStation 4)).

7. Examples

This photo below represents example of FreeBSD.



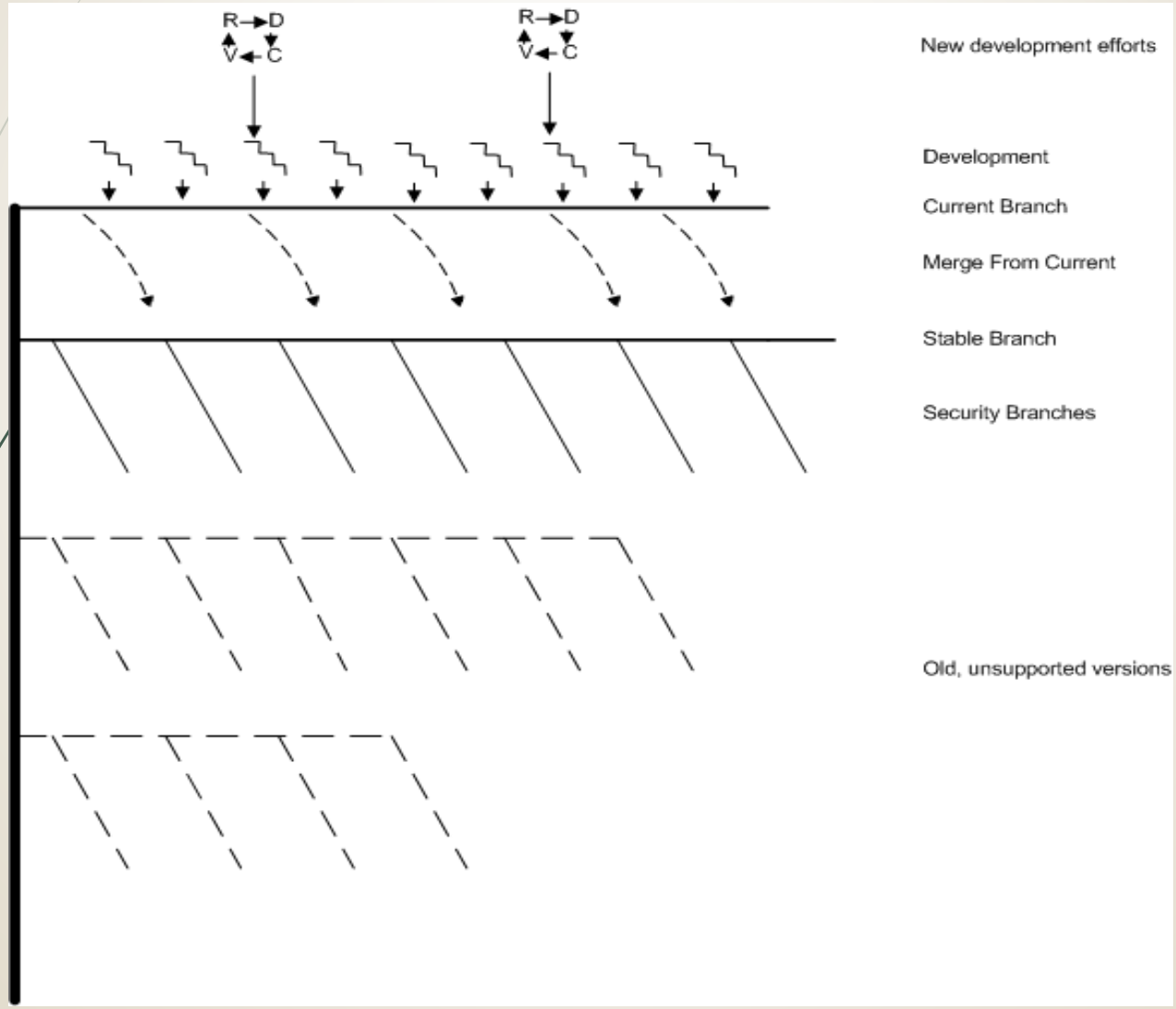
Another Examples of FreeBSD




```
root@freebsd10:~ # dd if=/dev/random of=/root/en.swap0 bs=1m count=64
64+0 records in
64+0 records out
67108864 bytes transferred in 2.086772 secs (32159171 bytes/sec)
root@freebsd10:~ # mdconfig -a -t vnode -f /root/en.swap0
md0
root@freebsd10:~ # geom eli init md0
Enter new passphrase:
Reenter new passphrase:

Metadata backup can be found in /var/backups/md0.eli and
can be restored with the following command:

        # geli restore /var/backups/md0.eli md0

root@freebsd10:~ # geom eli attach md0
Enter passphrase:
root@freebsd10:~ # swapon /dev/md0
md0%     md0.eli%
root@freebsd10:~ # swapon /dev/md0.eli
root@freebsd10:~ # swap
swapctl  swapinfo swapoff  swapon
root@freebsd10:~ # swapinfo -k
Device          1K-blocks     Used    Avail Capacity
/dev/ada0p3       1048540        0  1048540     0%
/dev/md42          131072        0   131072     0%
/dev/md0.eli        65532        0    65532     0%
Total             1245144        0  1245144     0%
root@freebsd10:~ #
```

8. Summary

The development model of FreeBSD can be seen as the following tree:



9. References

https://lists.freebsd.org/pipermail/freebsd-stable/2012-June/068129.html

https://www.freebsdfoundation.org/press/2013Jul-newsletter#whatsapp

http://flightaware.com/about/code/freebsd.rvt

http://doc.dl.playstation.net/doc/ps4-oss

http://doc.dl.playstation.net/doc/ps3-oss

https://en.wikipedia.org/wiki/FreeBSD

https://www.freebsd.org

Thank You