

Sistemas operativos libres para servidores

Arquitectura de servidores con software libre

Miguel Vidal, José Castro

{mvidal,jfcastro}@libresoft.es

8 de abril de 2011

we study libre software

GSyc



Universidad
Rey Juan Carlos

GSyC

© 2009-2011 Miguel Vidal, Jose Castro.

Esta presentación se distribuye bajo
licencia Creative Commons Reconocimiento 3.0 España



<http://creativecommons.org/licenses/by/3.0/es>

¿Quiénes somos?

- **Miguel Vidal** (<http://gsyc.es/~mvidal>):
 - Desplegó la actual infraestructura HA de Morfeo y ha colaborado en la administración y mantenimiento a bajo nivel de la plataforma OSO-R.
 - Administró los servidores de barrapunto.com durante seis años.
 - Coordinador del Máster de Software Libre (URJC) y profesor en la Escuela de Negocios EOI.
 - Responsable del proyecto de traducción al español de la documentación de OpenBSD.
- **José Castro** (<http://gsyc.es/~jfcastro>):
 - Responsable de sistemas de la plataforma HA de Morfeo.
 - Parte del equipo técnico de la plataforma europea OSO-R.
 - Coordinador de la asignatura de “Implantación” en el Máster oficial de software libre de la URJC.
 - Miembro fundador de Madrid-OSUG (comunidad de usuarios de OpenSolaris en Madrid).

Agenda

- 1 Breve historia de Unix
- 2 Variantes de Unix





¿Qué es Unix?

- Sistema operativo multitarea y multiusuario. Muy portable (C).
- No hay un solo Unix, sino numerosas ramas.
- Probablemente cientos de variantes a lo largo de más de 40 años de historia.
- Se desarrolla al tiempo que Internet y es la base de la tecnología internet (TCP/IP).
- Los Unices comparten una estructura común, compatibilidad binaria (ELF), POSIX shell, servicios y utilidades como awk, echo, ed, vi y muchas otras.

Universo Unix

- **Universo**: nombre con el que tradicionalmente se conocen las variantes y entornos de Unix.
- Es muy raro que un sysadmin sea responsable de un solo SO.
- Unix es muy diverso: de móviles a supercomputadoras.
- Donde más se percibe esta diversidad es en la administración de sistemas.

[http://en.wikipedia.org/wiki/Universe_\(Unix\)](http://en.wikipedia.org/wiki/Universe_(Unix))

¿Qué es Unix? La marca

- Oficialmente Unix es una marca registrada, controlada por el consorcio Open Group: UNIX™
- El Open Group, formado por grandes corporaciones (Oracle, HP, IBM, Fujitsu...) concede el uso de la marca a quienes cumplen con la Single UNIX Specification (SUS), la versión 4 es también conocida como POSIX:2008 (Portable Operating System Interface [for Unix]).
- El uso de la marca cuesta dinero y solo los Unixes comerciales (y privativos) tienen la certificación: AIX, HP-UX, SCO, Solaris, Mac OS X, IRIX...
- El certificado no requiere el código fuente, por lo que pueden no tener código en común ni ser derivados del Unix original.
- Comparten POSIX shell, servicios y utilidades como awk, echo, ed, vi y muchas otras.

¿Qué es Unix?

- Para los modelos de desarrollo abiertos, la especificación es demasiado cara e insostenible.
- GNU: **GNU's Not Unix**. En la década de 1980 intentó desarrollar un sustituto libre de Unix (junto al kernel Linux es la base de los actuales sistemas GNU/Linux).
- Para los SOs que no cumplen la especificación, se suele usar el término *Unix-like* (“tipo Unix”), “*nix” o “Un*x” para sortear el problema del uso de la marca (aunque esto no gusta a sus propietarios).
- FreeBSD tiene una certificación “C99” (ISO 9899:1999) conforme POSIX, que cumple en gran parte con SUS.
- Linux usa una especificación LSB (Linux Standard Base), muy próximo a POSIX y que más o menos siguen todas las distribuciones.

Clases de Unix

Clasificación de Eric Raymond:

- **Unix genético**: descendientes del código Unix original de AT&T (muchos Unix comerciales y los actuales BSD).
- **Unix de marca**: los que tienen la especificación SUS (Solaris, AIX, HP-UX, MacOS X...)
- **Unix funcional**: los que se acercan a la especificación POSIX o se comportan de forma consistente como Unix (como Linux o Minix), pero no poseen la marca ni descienden del código del Unix original.

El surgimiento de Unix

El nacimiento de Unix fue una auténtica revolución del software:

- 1969: Ken Thompson inventó Unix (mismo año que Arpanet).
- Surge de los desechos de Multics, en AT&T (Bell Labs).
- Dennis Ritchie inventa un nuevo lenguaje llamado C para usarlo en el Unix de Thompson.
- Primer sistema operativo portable y modular (KISS), frente a anteriores sistemas incompatibles y costosos.
- Se extiende rápidamente y de forma no oficial por AT&T. Y por Arpanet (hardware distinto, gracias a C).
- Acuerdo judicial (*antitrust*) de 1956 impide a AT&T comercializar Unix: debe licenciarlo (con fuentes) a quien se lo solicite.

Años setenta: Unix y Berkeley

- CSRG (Computer Systems Research Group) de Berkeley:
 - Importancia de compartir fuentes (cultura Unix “original”).
 - Limitado por la licencia AT&T (poco desde el punto de vista práctico, todos la tenían).
 - Financiado por DARPA (DoD).
 - Utilizado por mucho software privativo (SunOS, Ultrix, etc.)
- Primera Internet:
 - Implementaciones de referencia, disponibles para todos: la base de los estándares actuales y servicios actuales.
 - La Red como herramienta de cooperación (news, ftp, e-mail).
 - La comunidad de usuarios proporciona el mejor soporte.

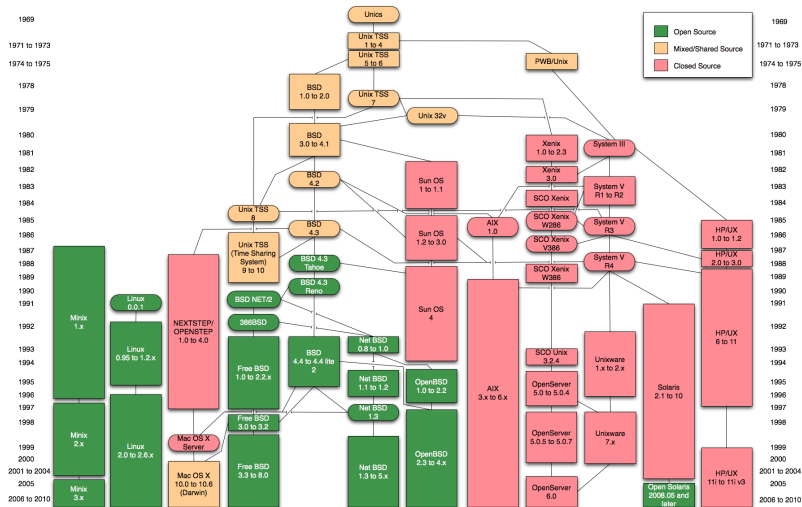


Figura: Historia de Unix. Fuente: Wikipedia

La herencia de BSD

- El Computer Systems Research Group (CSRG) libera la implementación de TCP/IP que desarrollaron ellos y todos los SO la adoptan (Net/1, 1989).
- Las distribuciones NetBSD, FreeBSD y OpenBSD surgen a partir de la adaptación original de 386BSD, basada en 4.4 BSD-Lite del CSRG (1992).
- Desde la distribución de 386BSD el desarrollo es rápido y se consigue un sistema estable.
- Mezcla de bazar y catedral, en paralelo al desarrollo de Linux.



Variantes de Unix

Dos grandes variantes históricas:

- 1 System V
 - 2 BSD
- Algunos sistemas mantenían las dos versiones en paralelo (con comandos, directorios, páginas *man* y librerías distintos). A estas variantes se les llamaba “universos”.
 - Esta división era problemática a la hora de portar aplicaciones y mantener los sistemas.
 - Cada universo fue adoptando lo mejor del otro.
 - En 1988, se produce una fusión entre ambas: **System R4**.
 - Hoy día quedan reminiscencias en algunos sistemas, que tienen un directorio separado con los comandos estilo BSD o System V.

Un ejemplo: el comando 'ps' en Linux

```
PS(1)                                Linux User's Manual                                PS(1)

NAME
    ps - report a snapshot of the current processes.

SYNOPSIS
    ps [options]

DESCRIPTION
    ps displays information about a selection of the active processes. If
    you want a repetitive update of the selection and the displayed
    information, use top(1) instead.

    This version of ps accepts several kinds of options:
    1  UNIX options, which may be grouped and must be preceded by a dash.
    2  BSD options, which may be grouped and must not be used with a dash.
    3  GNU long options, which are preceded by two dashes.

    Options of different types may be freely mixed, but conflicts can
    appear. There are some synonymous options, which are functionally
    identical, due to the many standards and ps implementations that this
    ps is compatible with.

EXAMPLES
    To see every process on the system using standard syntax:
    ps -e
    ps -ef
    ps -eF
    ps -ely

    To see every process on the system using BSD syntax:
    ps ax
    ps axu
```

Los dos grandes “universos” de Unix

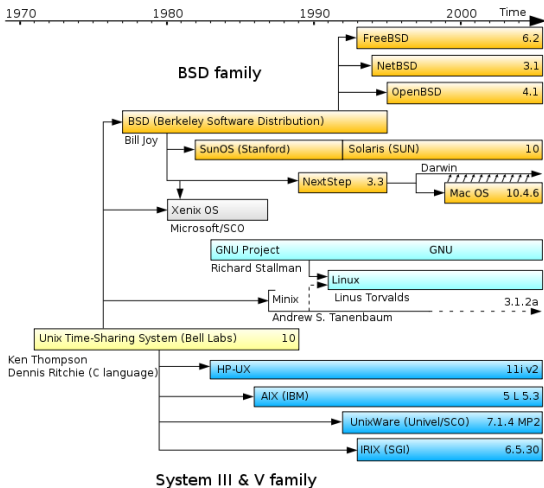


Figura: Los dos grandes “universos” de Unix. *Fuente: Wikipedia*

Unixes libres: los BSD

No son clones, son derivados del BSD Unix original. Principales proyectos:

- FreeBSD
- NetBSD
- OpenBSD: fork de NetBSD (1995)
- DragonFly BSD
- PC-BSD

Cada uno tiene, a su vez, numerosas variantes.

we study libre software
Lista de SOs basados en BSD:

http://en.wikipedia.org/wiki/List_of_BSD_operating_systems

Unixes libres: los BSD

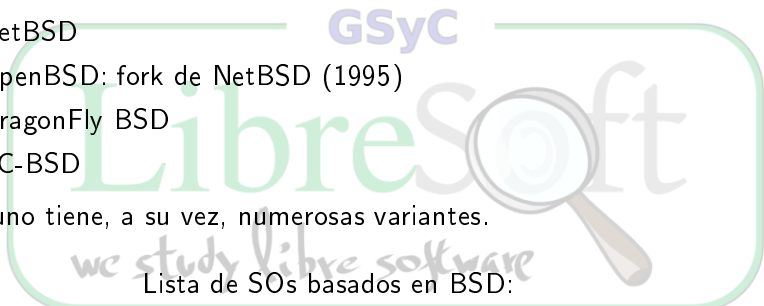
No son clones, son derivados del BSD Unix original. Principales proyectos:

- FreeBSD
- NetBSD
- OpenBSD: fork de NetBSD (1995)
- DragonFly BSD
- PC-BSD

Cada uno tiene, a su vez, numerosas variantes.

Lista de SOs basados en BSD:

http://en.wikipedia.org/wiki/List_of_BSD_operating_systems



Unixes libres: FreeBSD

- Es el BSD más popular. Rápido y optimizado para plataformas i386/amd64.
- Rápida incorporación de mejoras.
- Su kernel incorpora un sistema de virtualización ligera muy apreciado: las *jails*
- Ha portado el sistema de ficheros ZFS de OpenSolaris.

Unixes libres: OpenBSD (1)

- Se concentra en la corrección, seguridad proactiva, portabilidad (17 arquitecturas) y libertad.
- Código del sistema base auditado, características de seguridad y criptografía integradas.
- PF: el mejor firewall
- OpenSSH: la mejor shell segura.
- No intenta estar a la última, prioriza la sencillez y la estabilidad.

Unixes libres: OpenBSD (y 2)

- Comunidad preocupada por la libertad del software: no NDAs, no blobs, la licencia más permisiva de todas (ISC).
- La calidad de su documentación es legendaria.
- Introdujo el uso de CVS y el registro de *commits*, luego adoptado por todas las comunidades de software libre.
- Ha logrado que muchos fabricantes de tarjetas de red liberen especificaciones de sus drivers.

we study libre software

Unixes libres: NetBSD

- Orientado a la portabilidad: se propone funcionar en tantas arquitecturas de hardware como sea posible (actualmente 57 plataformas distintas).
- Gracias a su licencia permisiva y su portabilidad, es muy usado en sistemas empotrados.
- Como todos los BSD actuales, deriva del BSD-lite del CSGR de Berkeley.
- Es el antecesor de OpenBSD.

Unices libres: derivados de OpenSolaris

Principales proyectos:

- OpenSolaris
- illumos
- Nexenta
- OpenIndiana
- SchilliX

Todos ellos comparten el kernel SunOS 2.x.



OpenSolaris

- **Service Manager Facility (SMF)**: sistema de gestión de servicios que reemplaza a los scripts `init.d` (SVR4).
- **ZFS** (Zettabyte File System): sistema de ficheros nativo de OpenSolaris que provee administración simplificada, cifrado transparente, volúmenes lógicos, snapshots y *copy-on-write*, chequeo de integridad, RAID-Z, NAS/SAN y una escalabilidad inmensa. Bajo licencia CDDL, por tanto no compatible con Linux (hay *workarounds*).
- **DTrace**: Herramienta de instrumentación para depurar problemas y errores en el SO y sus aplicaciones en producción y en tiempo real, sin apenas impacto.

OpenSolaris (2/2)

- **Solaris Containers** (aka **Zonas**): virtualización ligera. Entornos aislados con una sola instancia del SO. Equivalente a las *jails* de FreeBSD.
- **LDOMs**: Paravirtualización para arquitectura Sparc (estilo Xen, pero con las ventajas del soporte multi-hilo de las CPUs Sparc).
- **Crossbow**: virtualización de redes y recursos para virtualizar el stack completo y las NICs en torno a cualquier servicio.

Unixes libres: Linux

- Linux es un kernel escrito desde cero.
- Es un clon, no un derivado de Unix: pero Dennis Ritchie lo considera un “Unix de facto”.
- El proyecto lo inicia Linus Torvalds en 1991, y *just for fun*
- Incorpora aspectos de las variantes System V y BSD.
- Contiene mucho software con origen BSD.
- Modelo bazar: desde que liberó la primera versión (0.01) se van uniendo cientos de desarrolladores en un esquema innovador (*release early, release often*).
- Se adopta la licencia GPLv2.
- Marzo 1994: versión 1.0

Unixes libres: Linux

- Debian y derivados: Ubuntu, Knoppix
- Red Hat y derivados: RHEL, CentOS, Fedora
- Gentoo y derivados: Sabayon
- Ubuntu y derivados: Xubuntu, Kubuntu, Edubuntu, gnewSense, Chrome OS, distros regionales (Guadalinex, LliureX)...
- Mandriva, SuSE, Slackware...

we study libre software

http://en.wikipedia.org/wiki/List_of_Linux_distributions

Unixes libres: Linux

- Debian y derivados: Ubuntu, Knoppix
- Red Hat y derivados: RHEL, CentOS, Fedora
- Gentoo y derivados: Sabayon
- Ubuntu y derivados: Xubuntu, Kubuntu, Edubuntu, gnewSense, Chrome OS, distros regionales (Guadalinex, LliureX)...
- Mandriva, SuSE, Slackware...

Lista de distribuciones Linux:

http://en.wikipedia.org/wiki/List_of_Linux_distributions

El caso de MacOS X

- En 1997, Apple Computer refunda su sistema operativo a partir de NeXTSTEP.
- NeXTSTEP es un SO privativo desarrollado por NeXT a finales de los 80 y primeros 90.
- El núcleo del SO está basado en BSD y en el kernel Mach: pasó a llamarse Darwin después de que Apple lo adquiriera.
- Darwin es casi todo software libre (Apple Public Source License), pero Mac OS X **NO** lo es.
- Darwin y Mac OS X son el sistema Unix más usado en el mercado de los sistemas de escritorio.

Promiscuidad de los Unixes libres

Mezclas de proyectos y código **solo** posible con el software libre:

- Debian kFreeBSD (kernel FreeBSD en Debian)
- FreeBSD + ZFS
- Gentoo/*BSD: *userland* GNU manejado por Portage (el árbol de paquetes) con un kernel {Net,Free,Open}BSD.
- Nexenta: Kernel Solaris y *userland* estilo Ubuntu/Debian (paquetes deb, dpkg y apt).

Consejos a la hora de elegir una versión de Unix

- ¿Cuál es el propósito? (no tiene nada que ver un servidor web que un FW)
- ¿Usas hardware no estándar?
- ¿Prefieres definir el sistema a tu gusto o buscas un inicio sencillo?
- ¿Estás interesado en la seguridad?
- ¿Eres desarrollador? (soportes nativos o extensos para Java, Flash, etc.)
- ¿Necesitas tecnologías especializadas? (almacenamiento, virtualización...)
- ¿Tienes requisitos de licencias?

Distribuciones Unix-like

- Familias BSD, Linux y OpenSolaris se dividen en “distribuciones” (*aka* “distros”)
- Cada “distro” es un sistema operativo específico, con diferencias más o menos significativas, con una marca con la que se conoce y se hace *advocacy* o publicidad.
- Las distros son mantenidas por *desarrolladores*, que pueden ser voluntarios o profesionales, interconectados desde cualquier parte.
- Suele haber comunidades de usuarios en torno a cada distribución.
- Casi siempre están disponibles para descarga.

Diferencias entre distribuciones

- Sistemas de paquetes
- Hardware soportado
- Instalador / proceso de instalación
- Configuración/administración/operación
- Aplicaciones disponibles
- Comunidad (tamaño, foco más o menos especializado, etc.)
- Sistema base homogéneo (BSD)
- Esquema de licencias (permisivo vs. copyleft)

Diferencias entre distribuciones: empaquetado

- Sistemas de paquetes de Linux
 - **dpkg**: formato original de Debian, portado también a Nexenta/OSol
 - **apt**: gestor de paquetes .dpkg
 - **rpm**: formato de RedHat, hoy día portado a muchas distros (incluso a otros Unixes)
 - **yum**: gestor de paquetes .rpm
 - **portage**: Gentoo Linux (portado a FreeBSD)

we study libre software

Diferencias entre distribuciones: empaquetado

- Sistemas de paquetes de BSD
 - **pkg/ports** (FreeBSD, OpenBSD)
 - **pkgsrc** (NetBSD)
- Sistemas de paquetes de OpenSolaris
 - **pkg** (SVR4 package)
 - **IPS** (aka **pkg(5)**). Soporte ZFS, rollbacks.

we study libre software

Diferencias entre distribuciones: ports

- Colecciones de ports: FreeBSD y OpenBSD.
- Un `Makefile` descarga los fuentes, los descomprime, parchea, compila y da de alta en la base de datos del sistema de paquetes.
- Permiten definir las opciones de compilación
- Muy costoso en tiempo con programas grandes.
- La mayoría de ports tienen paquete precompilado (opción recomendada para OpenBSD)

Diferencias entre distribuciones: operaciones

Distintas shells por defecto:

- Linux: `bash`
- FreeBSD: `tcsh` (`csch` con completado y edición de comandos)
- OpenSolaris/illumos: `ksh93` (Korn Shell sde AT&T, liberada en 2005)
- OpenBSD: `pdksh` (Public Domain Korn Shell)

we study libre software

Diferencias entre distribuciones: scripts de inicio

- Unix histórico (Bell Labs): `/etc/rc` para demonios estándar y `/etc/rc.local` para demonios añadidos localmente
- Unix System V de AT&T: esquema de directorios en `/etc/rc.d`.
- S- (start) o K- (kill). Ejemplo: `S671pr`
- Linux: System V con runlevels. Scripts en `/etc/init.d`, symlinks con S- y K- situados en `/etc/rc0.d`, `rc1.d`, `rc2.d`, etc. que corresponden a los runlevels.
- BSD mantiene ficheros únicos: `/etc/rc.conf` y `/etc/rc.local`
- FreeBSD 8: más de 150 ficheros para arrancar demonios en `/etc/rc.d`.
- OpenBSD, todo en un fichero único `/etc/rc.local`. A partir de la versión 4.9, variable `rc_scripts` en `/etc/rc.conf.local`.

Scripts de inicio en Linux (System V)

```
mvidal@beagle2:~$ ls -l /etc/rc2.d/
total 4
-rw-r--r-- 1 root root 677 2010-11-01 17:36 README
lrwxrwxrwx 1 root root 20 2010-08-11 19:40 S20fancontrol -> ../init.d/fancontro
l
lrwxrwxrwx 1 root root 17 2010-08-11 20:33 S20hddtemp -> ../init.d/hddtemp
lrwxrwxrwx 1 root root 20 2010-08-11 18:28 S20kerneloops -> ../init.d/kernelloop
s
lrwxrwxrwx 1 root root 17 2011-01-05 03:42 S20postfix -> ../init.d/postfix
lrwxrwxrwx 1 root root 27 2010-08-11 18:28 S20speech-dispatcher -> ../init.d/sp
eech-dispatcher
lrwxrwxrwx 1 root root 17 2010-09-05 20:57 S20vboxdrv -> ../init.d/vboxdrv
lrwxrwxrwx 1 root root 25 2010-10-31 00:56 S20vboxweb-service -> ../init.d/vbox
web-service
lrwxrwxrwx 1 root root 17 2010-08-12 00:42 S20winbind -> ../init.d/winbind
lrwxrwxrwx 1 root root 19 2010-08-11 18:28 S25bluetooth -> ../init.d/bluetooth
lrwxrwxrwx 1 root root 20 2010-08-11 18:28 S50pulseaudio -> ../init.d/pulseaudi
o
lrwxrwxrwx 1 root root 15 2010-10-11 00:25 S50rsync -> ../init.d/rsync
lrwxrwxrwx 1 root root 15 2010-08-11 18:28 S50saned -> ../init.d/saned
lrwxrwxrwx 1 root root 19 2010-08-11 18:28 S70dns-clean -> ../init.d/dns-clean
lrwxrwxrwx 1 root root 18 2010-08-11 18:28 S70pppd-dns -> ../init.d/pppd-dns
lrwxrwxrwx 1 root root 14 2011-01-23 11:57 S75sudo -> ../init.d/sudo
lrwxrwxrwx 1 root root 24 2010-08-11 18:28 S90binfmt-support -> ../init.d/binfm
```

Scripts de inicio en OpenBSD 4.9

Nuevo /etc/rc.d en OpenBSD 4.9

```
/etc/rc.d/FOO start  
/etc/rc.d/FOO stop  
/etc/rc.d/FOO reload  
/etc/rc.d/FOO check
```

```
# En /etc/rc.conf.local:  
rc_scripts=freshclam clamd cupsd FOO"
```

Diferencias entre distribuciones: Licencias

- La licencia determina lo que podemos hacer con el software.
- La distribución (recopilación) puede tener una licencia distinta a los programas por separado, incluso privativa.
- **Licencias BSD:**
 - Esquema permisivo y minimalista (BSD, MIT e ISC)
 - 2, 3 y 4 cláusulas. FreeBSD (2-clauses). OpenBSD (ISC)
 - Preserva únicamente los derechos morales (la autoría y el copyright)
- **Licencia Linux (copyleft):**
 - Kernel y *userland* (GNU) es generalmente copyleft (aunque contiene también herramientas BSD).
 - Si se compila o se combina cualquier cosa con código GPL, el resultado debe ser GPL.
 - Solo si hay redistribución de los cambios, hay que mantener la GPL.
 - Espacio de usuario: cualquier licencia.

Sistemas operativos libres para servidores

Arquitectura de servidores con software libre

Miguel Vidal, José Castro

{mvidal,jfcastro}@libresoft.es

8 de abril de 2011

we study libre software

GSyc



Universidad
Rey Juan Carlos