

# El software libre y la administración de sistemas

## Arquitectura de servidores con software libre

Miguel Vidal, José Castro

{mvidal,jfcastro}@libresoft.es

25 de marzo 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



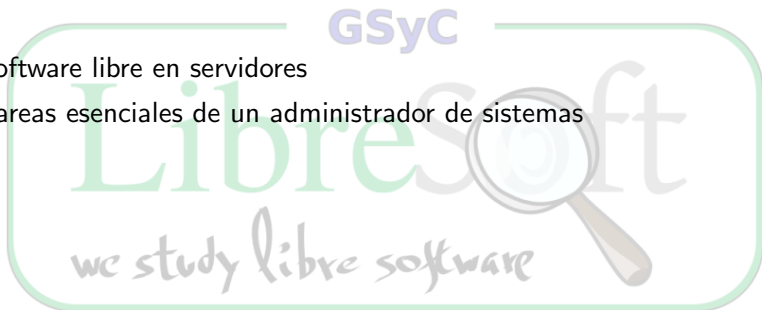
<http://creativecommons.org/licenses/by/3.0/es>

# ¿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 Software libre en servidores
- 2 Tareas esenciales de un administrador de sistemas





# Ventajas del software libre en servidores (I)

- Libertad de uso, modificación y redistribución:
  - podemos **instalarlo** en tantas máquinas como queramos.
  - podemos **adaptarlo** a nuestras necesidades o las del cliente.
  - podemos revisar el código y **corregir** errores sin esperar a que lo haga el fabricante.
  - podemos beneficiarnos de las mejoras y correcciones que hagan otros.
- Corrección mas rápida y eficiente de fallos, y rápida resolución de dudas y problemas, gracias al **modelo bazar** y a las fuertes comunidades que tiene detrás.

## Ventajas del software libre en servidores (II)

- **Independencia tecnológica:** no nos atamos a ningún proveedor en particular.
- Soporte y **compatibilidad** a largo plazo: el fabricante no está forzado a “vendernos” continuamente nuevas versiones.
- Fomento de la **libre competencia** al basarse en servicios y no en licencias.
- Ausencia de secretismo tecnológico y de patentes (seguridad jurídica).
- **Formatos estándar:** facilitan la interoperabilidad y evitan incompatibilidades.
- Métodos simples y unificados de gestión de software: las distribuciones evitan tener que acudir a buscar software de fuentes dudosas.

# Ventajas del software libre en servidores (y III)

- Inmensa **variedad** de soluciones muy **maduras**: el software libre nace en entornos de servidores.
- Demanda de técnicos FLOSS en expansión, gracias a la creciente adopción por parte de las AA.PP. y de grandes empresas tecnológicas (Google, IBM, Sun/Oracle, etc.).
- Sistemas **potencialmente más seguros**: hackers y empresas de seguridad de todo el mundo puedan auditar los programas.
- Aspectos económicos: más de mil millones de euros en licencias de Microsoft en España anuales (2006). Bajo TCO.
- Fiabilidad y rendimiento.



# Mercado de servidores con software libre

- El mercado suele medirse por unidades vendidas o por beneficios
- Difícil de evaluar para el caso del FLOSS: sistemas libres son a menudo obtenidos sin coste e instalados sin contratar soporte.
- Muchas veces se instalan en máquinas que no fueron compradas con software libre precargado.
- El método que se usa suele ser mediante acceso a máquinas públicamente accesibles (como servidores web).
- Problema: este método no contempla las máquinas no accesibles públicamente.

# Mercado de servidores

Source	Date	Method	Linux	Microsoft Windows	Unix				Other	References
					All Unix	BSD	Solaris	Other Unix		
W3Techs	September 2010	<b>Units</b> (Web)	63.7%	33.7%	2.7%	2.4%	0.1%	0.2%	<0.1%	[21][22][23]
IDC	Q1 2010	<b>Revenue</b>	16.2%	48.9%	22.2%					[24]
Security Space	July 2009	<b>Units</b> (Web)	74.29%	20.36%	5.35%	5.35%				[25][26]
Netcraft	Jan 2009	<b>Units</b> (Web)	41.02%	41.59%	5.54%	3.30%	1.90%	0.34%	11.83%	[27]
Gartner	2007	<b>Revenue</b>	23.2%	66.8%	6.8%					[28][29]

## Notes:

- Netcraft survey in January 2009 checked 1,014,301 publicly accessible **Web servers** with valid **SSL certificates**.
- Security Space survey in August 2009 checked 38,549,333 publicly accessible **Web servers**.
- W3Techs survey in September 2010 checked the top 1 million **Web servers** (according to **Alexa**).

# Compañías de hosting más fiables

## Most Reliable Hosting Company Sites in February 2011

Rank	Company site	OS	Outage hh:mm:ss	Failed Req%	DNS	Connect	First byte	Total
1	Datapipe	FreeBSD	0:00:00	0.004	0.100	0.021	0.051	0.076
2	Swishmail	FreeBSD	0:00:00	0.009	0.098	0.027	0.056	0.162
3	New York Internet	FreeBSD	0:00:00	0.009	0.129	0.029	0.081	0.218
4	www.westhost.com	Linux	0:00:00	0.009	0.291	0.076	0.163	0.502
5	ServInt	Linux	0:00:00	0.013	0.157	0.031	0.065	0.163
6	www.serverbeach.com	Linux	0:00:00	0.017	0.071	0.010	0.024	0.084
7	iWeb Technologies	Linux	0:00:00	0.017	0.098	0.044	0.089	0.089
8	www.singlehop.com	Linux	0:00:00	0.026	0.135	0.074	0.498	0.676
9	www.navisite.com	Windows Server 2003	0:00:00	0.030	0.188	0.043	0.220	0.380
10	www.poundhost.com	Linux	0:00:00	0.030	0.259	0.089	0.189	0.369

# ¿No hay desventajas?

- Necesidad de técnicos especializados (la gente se forma con SO privativos)
- Interfaces visuales (suelen ser privativos)
- No siempre hay soporte para todo tipo de hardware (patentes, drivers y especificaciones privativas).
- Suele ser necesario hacer *advocacy* y plantear migraciones.
- ¿Mayor mercado laboral en sistemas privativos? (depende del sector)



## ¿No hay desventajas?

- Necesidad de técnicos especializados (la gente se forma con SO privativos)
- Interfaces visuales (suelen ser privativos)
- No siempre hay soporte para todo tipo de hardware (patentes, drivers y especificaciones privativas).
- Suele ser necesario hacer *advocacy* y plantear migraciones.
- ¿Mayor mercado laboral en sistemas privativos? (depende del sector)

# ¿Y qué hay de las GUIs?

- Muchas distros traen GUIs o herramientas visuales propias.
- Son útiles y facilitan las tareas, sobre todo para sysadmins noveles.
  - Suelen ser propietarias
  - O nos hacen dependientes de una distro en concreto
  - A veces poseen oscuros detalles en la forma de gestionar los recursos
- Nosotros veremos siempre en las tecnologías y métodos subyacentes
- Estos suelen ser comunes a todas las distros, incluso a todos los Unixes.
- La configuración manual es mejor: más rápida, más flexible, más fiable, más potente y más *scriptable*.



# ¿Y qué hay de las GUIs?

- Muchas distros traen GUIs o herramientas visuales propias.
- Son útiles y facilitan las tareas, sobre todo para sysadmins noveles.
  - Suelen ser propietarias
  - O nos hacen dependientes de una distro en concreto
  - A veces poseen oscuros detalles en la forma de gestionar los recursos
- Nosotros veremos siempre en las tecnologías y métodos subyacentes
- Estos suelen ser comunes a todas las distros, incluso a todos los Unixes.
- La configuración manual es mejor: más rápida, más flexible, más fiable, más potente y más *scriptable*.

# ¿Es gratis el software libre? Algunos consejos

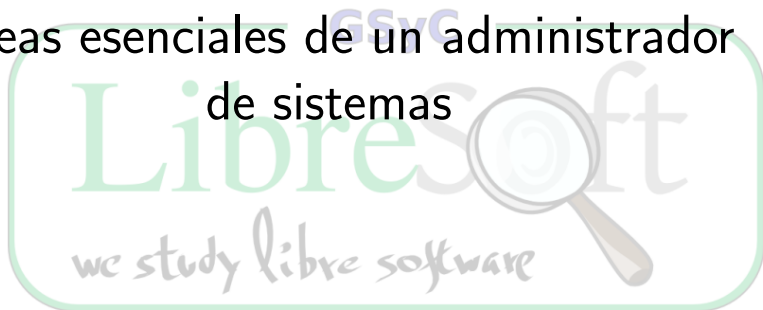
- La gratuidad no es el punto fuerte del software libre
- Insistir en la gratuidad supone minusvalorar el resto de ventajas (y es injusto para la gente que lo crea y lo mantiene).
- No comiences hablándoles de dinero a los que toman decisiones.
- No hablar del FLOSS en abstracto ("Linux es mejor"): estudia costes de migración y trata de cubrir necesidades concretas que no están cubiertas o mejorar lo que hay.
- No seas impaciente: deja que el software libre crezca con los clientes, introduciendo mejoras de forma progresiva.



## ¿Es gratis el software libre? Algunos consejos

- La gratuidad no es el punto fuerte del software libre
- Insistir en la gratuidad supone minusvalorar el resto de ventajas (y es injusto para la gente que lo crea y lo mantiene).
- No comiences hablándoles de dinero a los que toman decisiones.
- No hablar del FLOSS en abstracto (“Linux es mejor”): estudia costes de migración y trata de cubrir necesidades concretas que no están cubiertas o mejorar lo que hay.
- No seas impaciente: deja que el software libre crezca con los clientes, introduciendo mejoras de forma progresiva.

# Tareas esenciales de un administrador de sistemas



# ¿Qué es un administrador de sistemas?

*“Un administrador de sistemas es aquel profesional que tiene la responsabilidad de ejecutar, mantener, operar y asegurar el correcto funcionamiento de un sistema informático y/o una red de ordenadores.”* (Wikipedia).

También llamado **sysadmin**, debe demostrar una mezcla de **cualidades técnicas** y de **responsabilidad** para desempeñar bien su trabajo.

we study libre software

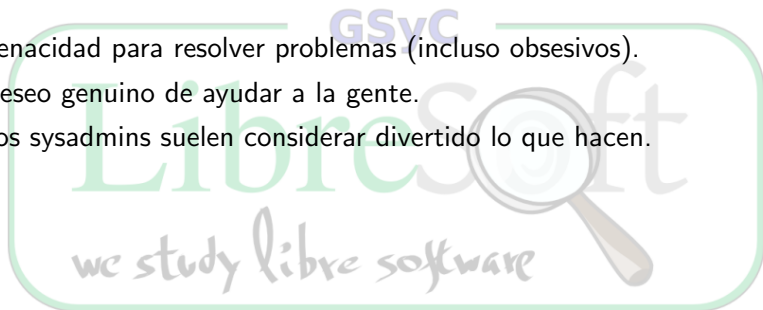
# Tareas esenciales de la administración de sistemas

- Instalación, soporte y mantenimiento de servidores o de otros sistemas informáticos.
- *Scripting* o programación ligera.
- Gestión de proyectos en proyectos relacionados con sistemas.
- Supervisión y formación de operadores.
- mantenimiento: Monitorización del sistema, ejecutar backups, actualizar software, añadir y retirar hardware...
- Creación, organización y mantenimiento de la documentación.
- Soporte a usuarios.

Todas estas tareas no necesariamente las lleva a cabo una sola persona. Pero al menos una persona debe conocerlas y asegurarse de que alguien las hace.

# Habilidades

- Tenacidad para resolver problemas (incluso obsesivos).
- Deseo genuino de ayudar a la gente.
- Los sysadmins suelen considerar divertido lo que hacen.

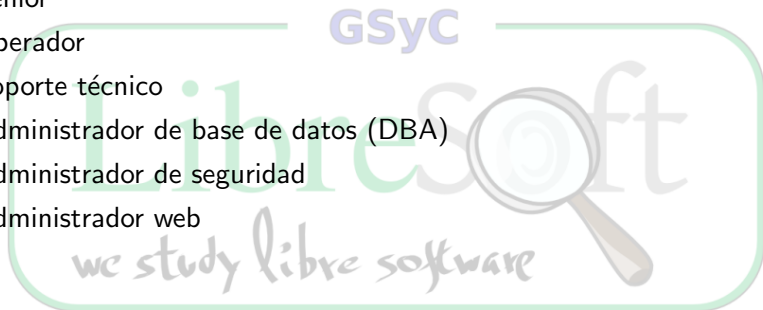


# Habilidades, formación

- La administración de sistemas implica más cambios de contextos en un solo día que la mayoría de trabajos en un año.
- Un sysadmin necesita habilidad para organizarse y gestionar su tiempo eficientemente.
- Habilidad para mantener felices a los usuarios en una situación *win-win*.
- El “queme” en el trabajo de un sysadmin es creciente. La mayoría de los administradores duran solo unos cuantos años.
- A diferencia de otras profesiones, no existe una única vía para convertirse en sysadmin.

# Tipos de sysadmin

- senior
- operador
- soporte técnico
- administrador de base de datos (DBA)
- administrador de seguridad
- administrador web







# Documentación

- Lo último que quiere hacer un sysadmin es crear o mantener documentación.
- Tarea ardua y poco valorada.
- Tampoco suelen querer aprender herramientas como LaTeX, SGML o groff.

*we study libre software*

# Importancia de documentar

- La documentación ayuda a la reproducibilidad.
- La documentación ahorra tiempo.
- La documentación facilitan el aprendizaje de nuevos administradores (algo que beneficia a todos).
- Lo principal: la documentación mejora la inteligibilidad de un sistema y permite que las modificaciones se hagan de un modo consistente.
- Escribe documentos cortos: de una página que cubran un solo tema.
- La documentación local debe guardarse en un solo punto bien definido y conocido (wiki, repo, sección de páginasman...).

# Recursos documentales

- Páginas man: tradicional doc. online.
  - Están organizadas por secciones.
  - Una misma orden puede estar en varias secciones.
  - No son howtos.
- GNU Texinfo (reemplazo del formateador nroff –privativo– usado en AT&T). Hoy tiene poco sentido, pero GNU las sigue apoyando.
- Guías y documentación específica de cada sistema (ej. *FreeBSD Handbook* o `docs.sun.com`)
- Documentación específica del paquete: (ej. `/usr/share/doc`)
- Libros en papel (O'Reilly)
- Linux Documentation Project
- RFCs

# Procedimientos

Algunas tareas comunes que suelen necesitar procedimientos:

- Añadir un *host*
- Añadir un usuario
- Configurar los backups para una nueva máquina
- Securitizar una nueva máquina
- Actualizar el sistema operativo
- Hacer respaldo y restauración de datos
- Ejecutar apagados de emergencia

# Políticas

## Políticas habituales:

- Políticas de seguridad
- Políticas para los administradores (login, sudo, pfexec...)
- Acceso y políticas de usuario
- Política de privacidad
- Cuestiones legales: copyright (licencias y datos almacenados), cifrado, protección de datos personales. . .

# Sistemas de seguimiento de incidencias

LibreSoft  
we study libre software

# Sistemas de seguimiento de incidencias

- Software para crear, actualizar y resolver listas de incidencias. Similar a una "bugtracker".
- Contiene una base de conocimientos con soluciones a problemas comunes: recurso de valor incalculable para el personal administrador de sistemas.
- **Ticket/incidencia:** una ficha que contiene información sobre las intervenciones de soporte realizadas por el personal técnico.
- Trac (python), RT (Perl), Redmine (RoR), OTRS, Mantis...:  
[http://en.wikipedia.org/wiki/Comparison\\_of\\_issue\\_tracking\\_systems](http://en.wikipedia.org/wiki/Comparison_of_issue_tracking_systems)

# Funciones comunes de un sistema de gestión de incidencias

Los responsables de proyecto pueden extraer valiosa información de alto nivel como:

- El número de tickets abiertos
- El tiempo medio en cerrarse un ticket
- La productividad de los sysadmins
- El porcentaje de tickets no resueltos
- Posibles desequilibrios en la distribución de la carga de trabajo



# Flujo

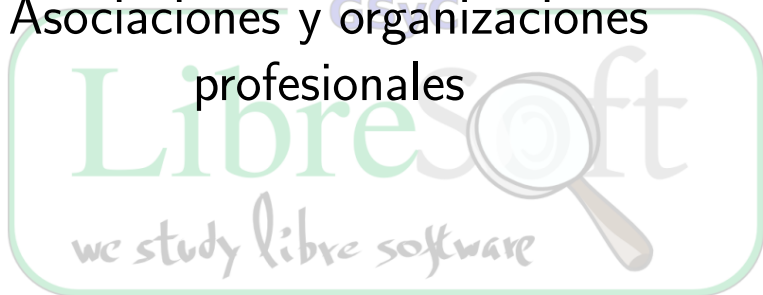
- El usuario (o el *helpdesk*) reporta un problema.
- El operador verifica que el problema es real y no solo una impresión.
- EL operador se asegura de obtener suficiente información sobre el problema por parte del usuario.
- La incidencia se asigna a la persona adecuada, que la marca como resuelta/cerrada/wontfix/feedback

# Redmine

## Principales características:

- Soporte multi-proyecto
- ACLs: acceso basado en roles muy flexibles.
- Wiki por proyecto
- Integración con SCM (SVN, CVS, Git, Mercurial, Bazaar y Darcs)
- Soporte para auto-registro
- Diagrama de Gantt y calendario
- Feeds y notificación por e-mail.

# Asociaciones y organizaciones profesionales



# SAGE

- Es la primera organización internacional para sysadmins.
- Es un grupo de interés dentro de Usenix.
- Promueve la administración de sistemas como profesión y patrocina conferencias y programas informales.
- Organiza el mayor evento para sysadmins: la conferencia USENIX LISA (Large Installation System Administration) en otoño.
- SAGE se enfoca más a la investigación.

# LOPSA

- LOPSA, League of Professional System Administrators.
- Se creó en 2005 por parte de algunos miembros de SAGE.
- Misión:
  - promover la práctica de la administración de sistemas;
  - apoyar, reconocer, educar y alentar a los sysadmins;
  - servir al público por medio de la educación y divulgación en temas relacionados con la administración de sistemas.
- LOPSA busca brindar apoyo legislativo a los temas que afectan a la profesión.
- SAGE y LOPSA cooperar en objetivos comunes, como el Código de Ética y la conferencia LISA.

# Herencia de Unix

- “KISS”, “Small is beautiful”, “Haz que cada programa haga bien una sola cosa”, “Construye un prototipo tan pronto como sea posible”, “Escoge la portabilidad sobre la eficiencia”, “Usa scripts de shell scripts para incrementar la portabilidad”, “Evita interfaces que hagan cautivos a los usuarios”, “Haz de cada programa un filtro”...
- Usenet, Internet jargon...
- System Administrator Appreciation Day (último viernes de julio)
- Bastard Operator From Hell (BOFH)

# Código ético (1)

LOPSA, USENIX y SAGE animan a que todo administrador se guía por un código ético:

- Profesionalidad
- Integridad personal
- Privacidad
- Leyes y políticas
- Comunicación



# Código ético (2)

- Integridad de sistema
- Educación
- Responsabilidad social
- Responsabilidad ética

<http://lopsa.org/CodeOfEthics>





# Referencias

- Nemeth, Snyder, Hein *UNIX and Linux System Administration Handbook*
- Limoncelli, Thomas A. *Time Management for System Administrators*

we study libre software