



**SystemStarter**

## Acerca de este documento

---

En este reportaje vamos a comentar como podemos configurar servicios de Mac OS X para que se ejecutan automáticamente al arrancar la máquina.

Este documento está escrito para Mac OS X 10.3 o anteriores. Debido a los cambios introducidos por Apple en el proceso de arranque de Mac OS X 10.4, si lo que quiere es configurar Mac OS X 10.4 o posterior debe consultar el reportaje "Configuración del sistema de arranque de Mac OS X".

El objetivo de esta sección no es aprender a lanzar aplicaciones de usuario al arrancar el ordenador, sino otro tipo de aplicaciones conocidas como servicios o demonios, que son aplicaciones no visuales (normalmente procedentes del mundo de UNIX) que permanecen lanzadas todo el tiempo y que dan servicios (p.e. servicios de red) como puedan ser: Apache, NFS, SSH, cron, etc.

Este reportaje está dirigido a usuarios familiarizados con sistemas UNIX, con lo que no vamos a explicar los detalles más comunes o conocidos de estos sistemas.

## Nota Legal

---

Este tutorial ha sido escrito por Fernando López Hernández para MacProgramadores, y de acuerdo a los derechos que le concede la legislación española e internacional el autor prohíbe la publicación de este documento en cualquier otro servidor web, así como su venta, o difusión en cualquier otro medio sin autorización previa.

Sin embargo el autor anima a todos los servidores web a colocar enlaces a este documento. El autor también anima a cualquier persona interesada en conocer el funcionamiento del SystemStarter a bajarse o imprimirse este reportaje.

Madrid, Septiembre 2005

Para cualquier aclaración contacte con:

[fernando@DELITmacprogramadores.org](mailto:fernando@DELITmacprogramadores.org)

## Tabla de contenido

---

Los ficheros de arranque .....	4
SystemStarter .....	5

## Los ficheros de arranque

---

Mac OS X, como sistema UNIX que es, dispone de unos ficheros de lotes en el directorio `/etc/` que se ejecutan siempre que arrancamos la máquina, estos ficheros son:

Fichero	Descripción
<code>/etc/rc.boot</code>	Fichero ejecutado al arrancar la máquina cuando todavía está en modo monousuario.
<code>/etc/rc</code>	Fichero ejecutado al pasar a modo multiusuario
<code>/etc/rc.common</code>	Fichero con rutinas comunes compartidas por <code>/etc/rc.boot</code> y <code>/etc/rc</code>

Si miramos en el fichero `/etc/rc` vemos que contiene una serie de inicializaciones básicas y que al final del script aparece una llamada de la forma:

```
SystemStarter ${VerboseFlag}
```

Esta llamada lo que hace es lanzar un programa llamado `SystemStarter` que es el que lanza todos los servicios del sistema. Este programa arranca la mayoría de los servicios que tradicionalmente se ponían en `/etc/rc`, con lo que el tamaño de `/etc/rc` en Mac OS X es moderadamente pequeño.

Como veremos, `SystemStarter` permite controlar mejor las dependencias que hay entre los distintos servicios (p.e. NFS no se puede lanzar hasta que este lanzado el portmap) y además permite la localización de los mensajes de arranque que dan estos programas al arrancar.

## SystemStarter

---

Al arrancar `SystemStarter` busca que servicios queremos arrancar en dos directorios:

Directorio	Descripción
<code>/System/Library/StartupItems/</code>	Contiene servicios que vienen instalados "de fabrica" en Mac OS X
<code>/Library/StartupItems/</code>	Directorio para colocar servicios adicionales que nosotros queremos arrancar.

Si miramos el contenido de `/System/Library/StartupItems/` podremos ver que estructura siguen estos servicios. En primer lugar el lector puede comprobar que se crea un subdirectorio por cada servicio que queramos arrancar. La siguiente tabla es un resumen de los servicios que podemos encontrar en este directorio.

Servicio	Descripción
<code>SystemTuning</code>	Ajusta valores de rendimiento del sistema en base a factores como la memoria RAM disponible.
<code>Cleanup</code>	Libera ficheros temporales que no fueron borrados durante el anterior arranque del sistema.
<code>Network</code>	Configura los interfaces de red en base al contenido del fichero <code>/etc/iftab</code> , fija en nombre de red de la máquina, configura el routing de red (si es necesario), fija el machine host ID del kernel, y carga los Shared IP kernel extension para poder compartir una dirección IP entre todos los entornos de aplicación (incluido Classic).
<code>Accounting</code>	Inicia el sistema de accounting ( <code>accton</code> ), el cual guarda información de log de cada proceso lanzado en el fichero <code>/var/account/acct</code> . Para que se haga este accounting debemos de crear primero el fichero.
<code>Portmap</code>	Inicia el demonio del portmapper de RPC
<code>Disks</code>	Ejecuta el demonio <code>autodiskmount</code> , que monta y comprueba las unidades de disco locales.
<code>AppleTalk</code>	Ejecuta el programa de arranque de AppleTalk en modo router, multihoming (sin router) o en un único puerto (en función de lo que diga el fichero de configuración <code>/etc/hostconfig</code> ).
<code>SystemLog</code>	Inicia el demonio de lo <code>syslogd</code> .
<code>DirectoryServices</code>	Inicia el NetInfo master server ( <code>nibindd</code> ), los

	Network Information Services, y el name-resolver daemon ( <code>lookupd</code> )
NFS	Arranca el servicio de Network File System que realiza asynchronous block I/O ( <code>nfsiod</code> ), monta el remote file systems, arranca el automounter, y, si la base de datos de NetInfo indica que el ordenador debe exportar un sistema de ficheros usando NFS, arranca el NFS server.
Apache	Arranca el servidor web Apache (si así está indicado en el panel de control).
AppleShare	Si se detecta conexión de red local, arranca el AppleShare service.
AppServices	Lanza el demonio de base de datos de escritorio ( <code>DesktopDB</code> ), actualiza la cache de tipos de letra y arranca los servicios de impresión.
IPServices	Arranca los servicios de TCP/IP, <code>inetd</code> , el servicio de arranque en red (BOOTP), y el netboot client management server.
NetworkTime	Arranca el cliente de Network Time Protocol (NTP) usado para sincronizar el reloj del sistema con un servidor de tiempo de Internet.
QTServer	Arranca el QuickTime Streaming Server.
SendMail	Arranca el servicio de salida de emails ( <code>sendmaild</code> ).
AuthServer	Arranca el servidor de autenticación.
Cron	Ejecuta el demonio <code>cron</code> para poder programar la ejecución de tareas periódicas.

El directorio `/Library/StartupItems/` puede estar vacío, o incluso no existir si no se han instalado servicios adicionales al sistema. Aunque si añadimos nosotros servicios deberíamos de añadirlos aquí y no en el directorio del sistema. Los ficheros que guardemos en esta carpeta (y la carpeta) deben de tener permisos de sólo lectura para los usuarios que no sean administradores, con el fin de evitar que un usuario sin permiso de administración modifique estos ficheros. Apple pide que tengan permiso de lectura para el grupo `wheel`, con el fin de poder ser lanzados. El usuario que tengan como dueño debe tener permiso de lectura y escritura (p.e. en mi caso el dueño sería el usuario `fernando`). Los ficheros del sistema (los de la carpeta `/System/Library/StartupItems/`) tienen como dueño a `system`, que es el dueño de todos los ficheros bajo la carpeta `/System`.

Cada servicio nuevo que queremos crear debe tener un subdirectorio con el nombre del servicio dentro del cual deben de aparecer al menos dos ficheros:

1. Un shell script con los comandos que arrancan este servicio, donde el nombre del script debe de coincidir con el nombre del subdirectorio.

Por ejemplo si miramos el fichero:

```
/System/Library/StartupItems/NFS/NFS
```

vemos que contiene algo tal que así:

```
#!/bin/sh

. /etc/rc.common

CheckForNetwork

if [ "${NETWORKUP}" = "-NO-" ]; then exit; fi

.....
.....
```

Todos estos script suelen empezar llamando a `/etc/rc.common` antes de ejecutar las operaciones necesarias para arrancar el entorno, de esto forma se fijan variables de entorno esenciales como `PATH`.

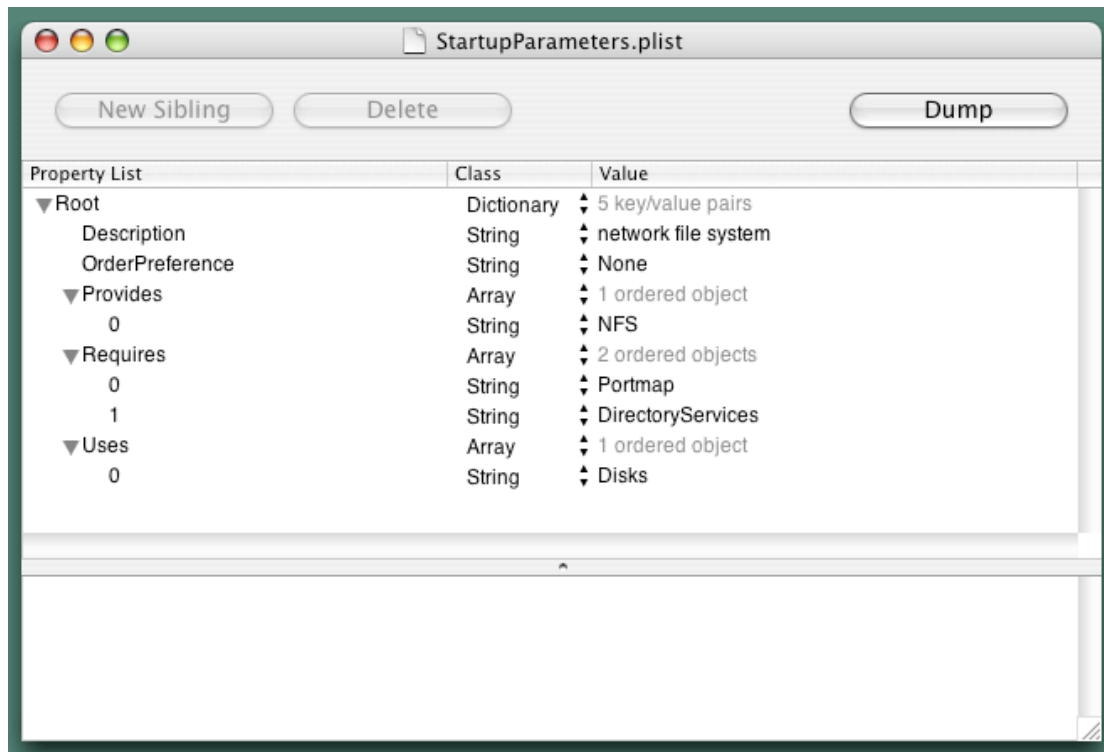
2. Un fichero llamado `StartupParameters.plist` en el que se indican los servicios que deben de estar arrancados antes de arrancar este servicio, así como los mensajes que se dan al arrancar y parar el servicio.

El contenido de este fichero tiene la forma:

```
{
  Description      = "Sun network file system";
  Provides         = ("NFS");
  Requires         = ("Portmap", "Resolver");
  OrderPreference = "None";
  Messages =
  {
    start = "Starting network file system";
    stop  = "Stopping network file system";
  };
}
```

Donde se indica que antes de poder lanzar este servicio deben de estar lanzados `Portmap` y `Resolver`.

Este fichero también puede ser editado desde el `PropertyListEditor`, una herramienta visual que se distribuye con las herramientas de desarrollo de Mac OS X:



3. Opcionalmente, en el directorio del servicio puede aparecer un subdirectorio llamado `Resources` dentro del cual puede haber mensajes para los distintos idiomas. Si aparece dentro de él encontraremos a su vez otro subdirectorio por cada idioma con un fichero `Localizable.strings`, que es un fichero de la forma:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist SYSTEM
"file://localhost/System/Library/DTDs/PropertyList.dtd"
>
<plist version="0.9">
<dict>
  <key>Starting network file system</key>
  <string>Iniciando sistema de archivos de red</string>
</dict>
</plist>
```

El fichero puede contener varios pares clave (`<key>`), valor (`<string>`), donde se indica la traducción de cada mensaje.

Por ejemplo en el caso de NFS que estamos estudiando, si el sistema está configurado en las preferencias del sistema para usar Español, el mensaje "Starting network file system" del fichero `StartupParameters.plist` será sustituido por "Iniciando sistema de archivos de red".