

LPIC-2 / Examen 208 - Servicios Web

208.2 Mantener un servidor web

Teoría

El mantenimiento de un servidor web es un proceso continuo para garantizar su disponibilidad, rendimiento, seguridad y estabilidad.

Tareas de Mantenimiento Clave:

1. Monitorización de Logs:

- Los logs de acceso y error son la fuente principal de información sobre lo que está sucediendo en el servidor web. Son esenciales para la resolución de problemas, el análisis de tráfico y la detección de actividad sospechosa.
- **Ubicación de Logs (Diferencias):**
 - **Apache (Debian/Ubuntu):** `/var/log/apache2/access.log`, `/var/log/apache2/error.log`.
 - **Apache (Red Hat/CentOS/Fedora):** `/var/log/httpd/access_log`, `/var/log/httpd/error_log`.
 - **Nginx (Ambas):** `/var/log/nginx/access.log`, `/var/log/nginx/error.log`.
- **Herramientas de Monitorización de Logs:**
 - `tail <archivo_log>`: Muestra las últimas líneas de un archivo.
 - `tail -f <archivo_log>` o `tailf <archivo_log>`: Sigue el archivo de log en tiempo real a medida que se añaden nuevas líneas.
 - `journalctl -u <servicio_web>`: Con systemd, muchos logs (incluyendo a veces los errores, dependiendo de la configuración de BIND) se pueden enviar al journal.
- **Interpretación de Logs:**
 - **Access Logs:** Cada línea registra una solicitud (IP cliente, fecha/hora, método HTTP, URL solicitada, código de estado HTTP - ej: 200 OK, 404 Not Found, 500 Internal Server Error, tamaño de la respuesta, agente de usuario).
 - **Error Logs:** Registran errores (ej: archivos no encontrados, problemas de permisos, fallos internos del servidor). El nivel de detalle se configura en el servidor web.

2. Rotación de Logs (logrotate**):**

- Los archivos de log pueden crecer muy rápido y llenar el espacio en disco. La rotación de logs archiva, comprime y elimina periódicamente los archivos de log antiguos.
- `logrotate` es la utilidad estándar. Se ejecuta típicamente como un trabajo diario de cron.
- **Archivos de Configuración de `logrotate`:**

- `/etc/logrotate.conf`: Configuración global de `logrotate`.
- `/etc/logrotate.d/`: Directorio que contiene archivos de configuración específicos para cada aplicación. Los paquetes de Apache y Nginx instalan sus propias configuraciones aquí.
- **Directivas Comunes en Archivos de Configuración:** `rotate <conteo>` (cuántas versiones antiguas mantener), `size <tamaño>` (rotar si alcanza un tamaño), `daily/weekly/monthly` (frecuencia), `compress` (comprimir versiones antiguas), `delaycompress` (comprimir en el siguiente ciclo), `missingok` (no reportar error si el archivo no existe), `notifempty` (no rotar si está vacío), `create <modo> <propietario> <grupo>` (crear un nuevo archivo de log con permisos específicos después de rotar), `prerotate/postrotate` (scripts a ejecutar antes/después de la rotación).
- **Nota:** Las configuraciones por defecto en `/etc/logrotate.d/` para Apache y Nginx son proporcionadas por los paquetes y suelen ser adecuadas, pero su contenido exacto y ubicación del archivo de configuración principal pueden tener ligeras variaciones entre distribuciones.

3. Gestión de Configuración:

- **Verificar Sintaxis:** Antes de recargar o reiniciar el servidor web, verifica que los archivos de configuración no contengan errores de sintaxis. Esto evita que el servidor no arranque.
 - Apache: `sudo apache2ctl configtest` (Debian/Ubuntu) o `sudo httpd -t` (Red Hat/CentOS/Fedora).
 - Nginx: `sudo nginx -t`.
- **Aplicar Cambios:** Después de modificar la configuración, recarga o reinicia el servicio.
 - `sudo systemctl reload <servicio_web>`: Recarga la configuración sin detener el servicio (si es posible, es preferible).
 - `sudo systemctl restart <servicio_web>`: Detiene y vuelve a iniciar el servicio. Necesario para algunos cambios (ej: añadir/eliminar módulos).

4. Monitorización de Rendimiento:

- Usa herramientas de monitorización del sistema (Ej. 200.1, 206.3) para ver el uso de CPU, memoria y red por parte del servidor web.
- **Páginas de Estado del Servidor Web:** Apache (`mod_status`) y Nginx (`stub_status`) pueden configurarse para proporcionar una página web con estadísticas internas (número de procesos/conexiones activas, tráfico, etc.). Esto es útil para ver la actividad del servidor web mismo.

5. Actualizaciones de Seguridad:

Mantener el paquete del servidor web y las librerías dependientes actualizados es crucial para protegerse contra vulnerabilidades conocidas.

6. **Gestión de Contenido:** Asegurarse de que los archivos en el DocumentRoot tengan los permisos y propiedad correctos para que el servidor web pueda leerlos (y a veces escribirlos). El usuario bajo el que corre el servidor web (ej: `www-data` en Debian, `apache` en Red Hat, `nginx` en Nginx) necesita leer los archivos.
7. **Páginas de Error Personalizadas:** Configurar páginas HTML amigables para errores comunes (404 Not Found, 403 Forbidden, 500 Internal Server Error) en lugar de mostrar las páginas por defecto del servidor o mensajes crípticos.

Resolución de Problemas Básicos de Mantenimiento:

- **Servidor no Inicia:** Verificar sintaxis de configuración (`configtest -t`). Revisar logs de `systemd` (`journalctl -u <servicio>`) o archivos de error del servidor web.
- **Página no Carga:** Verificar que el servicio está corriendo (`systemctl status`). Verificar reglas de firewall (puerto 80/443). Verificar logs de acceso y error. Verificar permisos de archivos y directorios en el Document Root.
- **Errores 404/403/500:** Revisar logs de error. 404: Archivo no encontrado (verificar ruta y nombre del archivo). 403: Permisos denegados (verificar permisos de archivo/directorio, configuración de Apache `<Directory>` o Nginx `location`, SELinux/AppArmor). 500: Error interno del servidor (verificar scripts dinámicos, configuración, logs de error detallados).