

# Mundo real: cómo administrar un servidor subversion para un equipo pequeño

Estas son mis notas tras 4 años de administración de un servidor Subversion (SVN) en Linux, para un equipo pequeño de 4 o 5 personas que acceden desde la oficina, o desde casa, usando clientes Windows, Mac o Linux.

Si lo que te interesa alojar tu servidor SVN en windows, puedes echar un vistazo a [VisualSVN Server para Windows](#) (con versión gratuita y de pago).

Las ventajas de alojar subversion en Linux:

1. Se puede hostear en servidores (o VMs) más pequeños y baratos que si fuera en Windows
2. Se puede aprovechar scripting Bash/Perl ya existente
3. Dispondremos fácilmente del protocolo "svn+ssh" para tener transferencias seguras (encriptadas) de forma simple, sin tener que usar Apache con certificados SSL, etc.
4. El sistema de usuarios es el propio de cuentas de usuario de Linux

Desventajas de alojar subversion en Linux bajo el protocolo "svn+ssh":

1. No tenemos la misma granularización de permisos en los repositorios y directorios. Todos los usuarios pueden acceder a todo y escribir en todas partes, si bien siempre hay alternativas o pequeños "hacks".
2. No tenemos "de entrada" un visor HTML de los repositorios como cuando instalamos una típica instalación SVN con el servidor web Apache, etc. aunque siempre se pueden instalar aparte.

Cosas que necesitaremos:

- Instalación y actualización
  - Instalar la versión de subversion deseada, en Ubuntu o CentOS
  - Poder actualizar el servidor SVN para estar al día
- Gestión de usuarios
  - Añadir usuarios (un nuevo empleado, por ejemplo)
  - Dar de baja usuarios (expiración de usuarios, por ejemplo cuando un empleado deja la empresa)
  - Si el servidor es solo para SVN, los usuarios \*solo\* deben poder hacer subversion
- Decidir nuestro "esquema de repositorios" (layout), por ejemplo el de "uno o más proyectos en cada repo", más flexible que el de "un repo = un proyecto".
- Aprender a usar los "externals" de svn
- Scripts:
  - Script para añadir repositorios
  - Script para añadir proyectos a un repo
  - Script para crear el esqueleto de un proyecto nuevo (boilerplate/skeleton)
  - Script para listar repositorios/proyectos
  - Script para buscar cadenas en ciertos archivos de todos los repositorios
  - Script para "empaquetar" los repos
  - Script para hacer copias de seguridad (backup) de los repos, de diversos modos
- Extras:
  - Uso de una rama "trunk" estándar, y como mucho una rama "release" para proyectos

que lo requieren

- Svnnotify (enviar email con diferencias al equipo en cada commit)
- Usuario “deploy” para desplegar (deploy), uso de “phinx” para cambios en Base de datos (migrations)

From:

<https://juangacovas.info/> - **JuangaCovas.info**

Permanent link:

<https://juangacovas.info/doku.php/subversion?rev=1595703764>

Last update: **25/07/2020 21:02**

