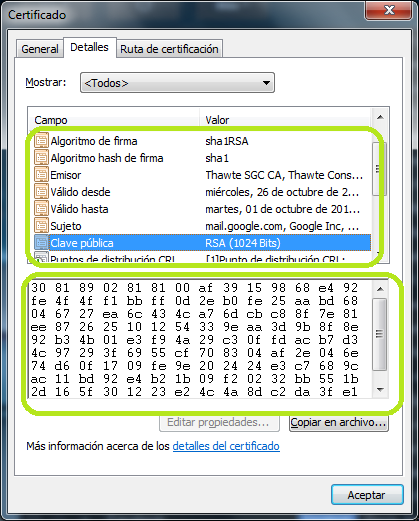
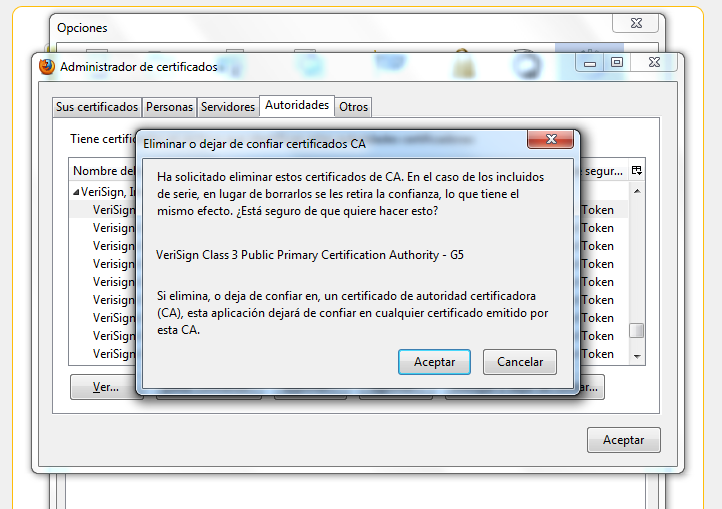
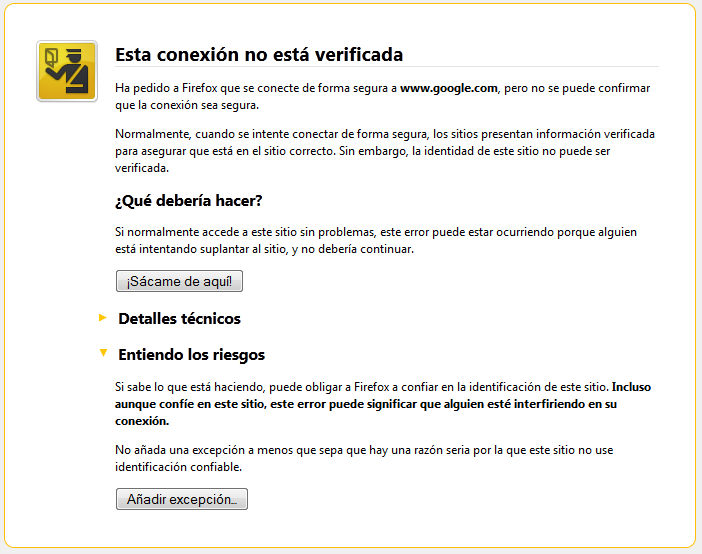
**Actividades con los navegadores (clientes):**

1. **Visita una Web que utilice HTTPS y comprueba los siguientes datos: Para quién se emitió el certificado, CA, algoritmo, fecha de caducidad y clave pública. Entregar datos obtenidos y pantallas de capturas.**
2. **Instala el certificado de una Web segura, visita la Web y comprueba que no vuelve a pedirte autorización, borra el certificado y visítala de nuevo. (Windows y Linux). Entregar paso a paso y pantallas de capturas.**

Entramos a google.com por ejemplo y luego vamos a borrar los certificados de google.com



Una vez borrado el certificado, volveremos a entrar a google.com y nos mostrara la siguiente pantalla. En la cual deberemos dar a **Añadir excepcion…**  Y ya tendremos instalado el certificado.

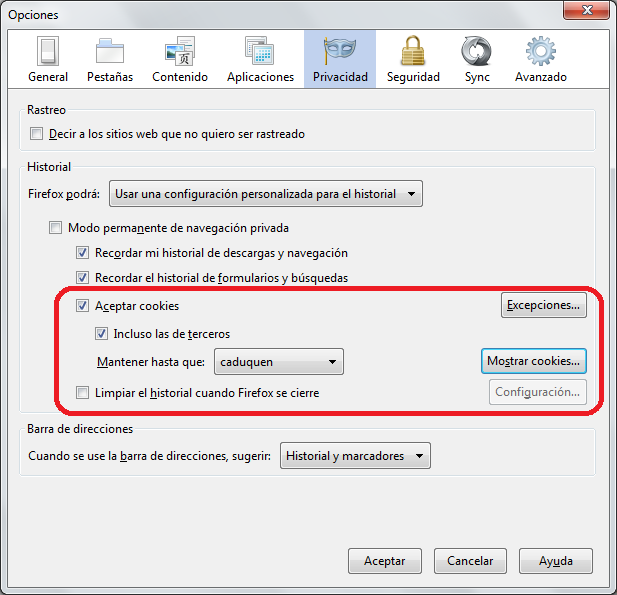


En Ubuntu haríamos lo siguiente:

En este caso vamos a la página terra.es.

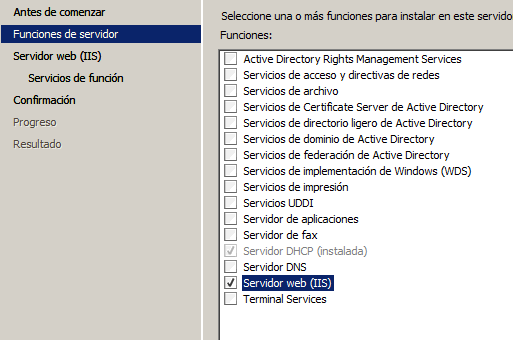
1. **Configura el navegador para que acepte todas las cookies, navega unos minutos por tus Web favoritas y consulta las cookies que se han guardado en tu equipo. Una vez comprobadas bórralas para no dejar rastros de tu uso de internet y configura el navegador a su estado anterior. Entrega listado de cookies y pasos realizados.**

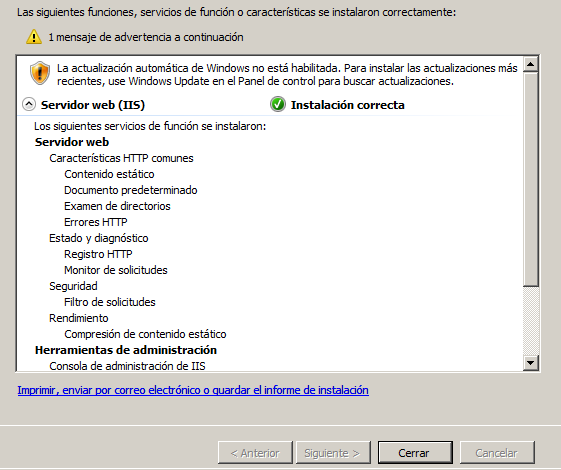
Vamos a opciones de firefox / privacidad/ y donde dice **Firefox podrá: Usar una configuración personalizada para el historial.**

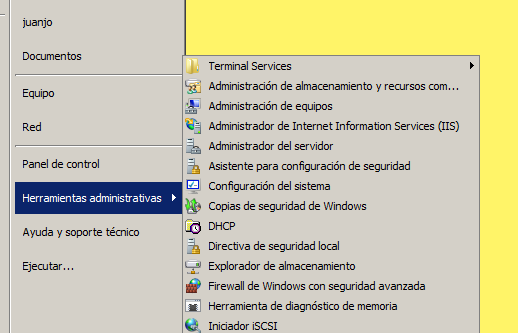


**Actividades con los servidores:**

1. Instalar IIS en Windows Server y Apache2 en Linux

Para la instalación de IIS en Windows Server 2008 debemos ir a **Herramientas del Servidor / Añadir nueva función** 



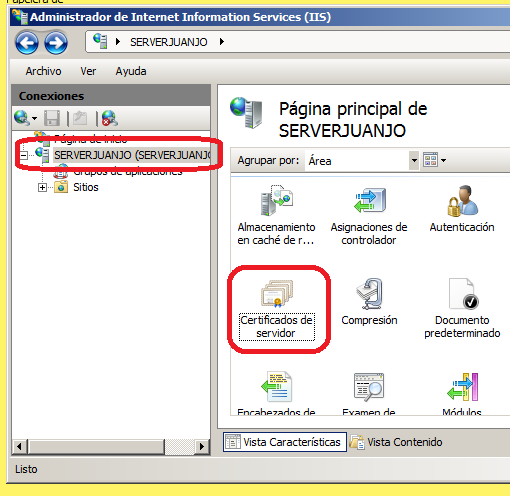


Ya tendremos instalado el servicio de IIS en Windows 2008.

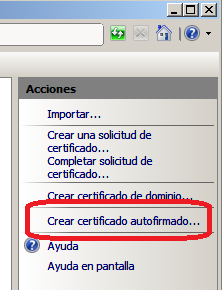
1. Crea un sitio Web seguro usando tu propio certificado digital (Windows y Linux).

Creación de certificado

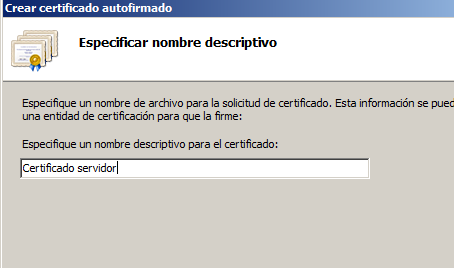
Para crear nuestro propio certificado vamos **a Administrador de Internet Information Services (IIS).** Y hacemos doble clic en **Certificados de servidor.**



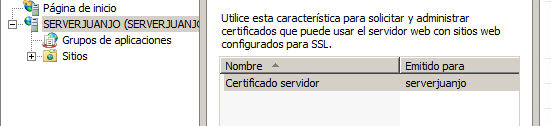
Ahora en el menú de la derecha le daremos a lo siguiente:

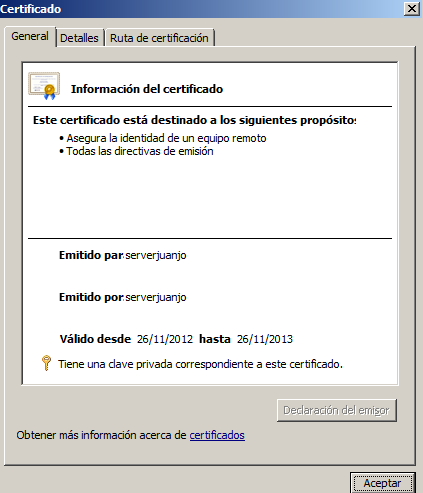


Ponemos un nombre descriptivo para el certificado.



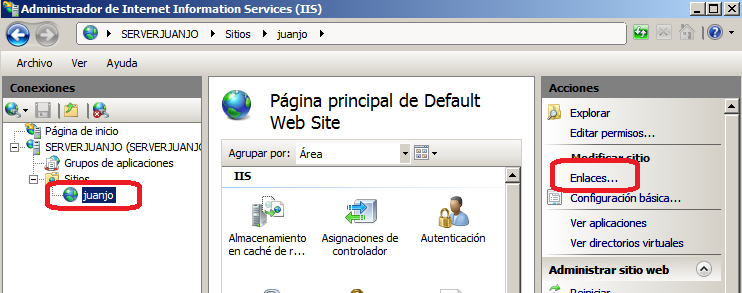
Veremos que el certificado se ha creado correctamente.



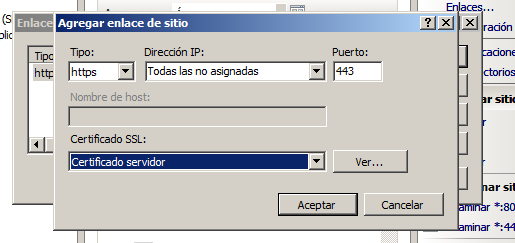


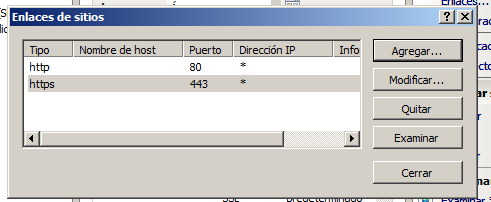
Instalación del certificado

Ahora lo añadimos a nuestro sitio web. Para ello:

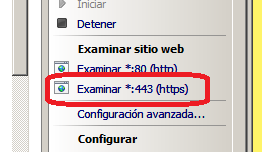


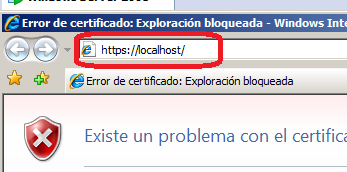
Seleccionamos **https y el certificado servidor.**





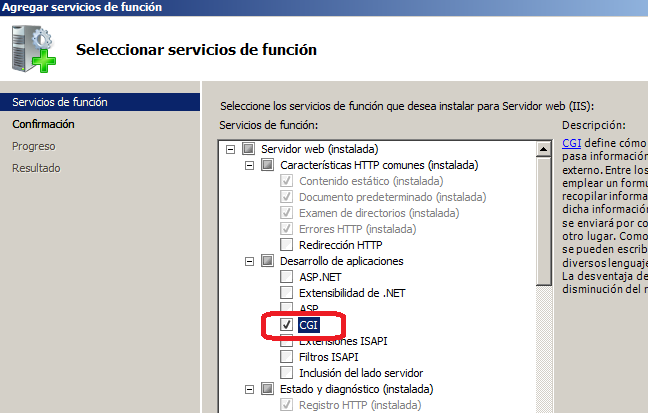
Para comprobarlo le daremos clic al siguiente enlace:





1. Configura IIS para que interprete correctamente las páginas creadas en PHP y comprueba que funciona correctamente.

Para ello vamos a **Administración del servidor / Funciones / Servidor web (IIS)** y hacemos clic sobre **Agregar servicios de función. Instalamos CGI.**

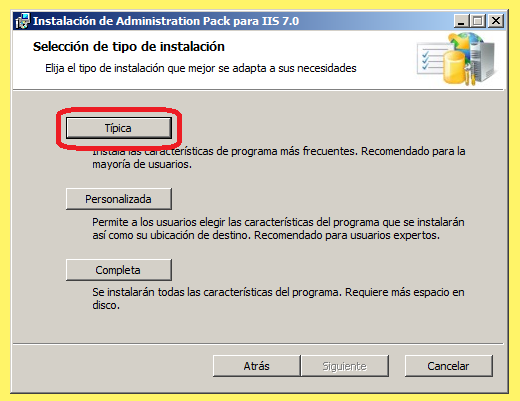
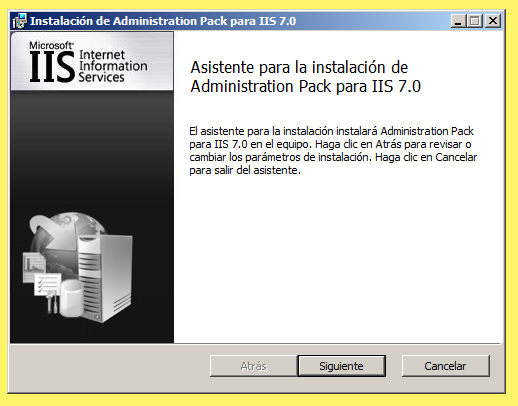
****

Una vez instalado tendremos que instalar una actualización para evitar problemas de incompatibilidad. Para ello vamos a la siguiente página: [Actualización KB954946](http://www.microsoft.com/es-es/download/details.aspx?id=9453)

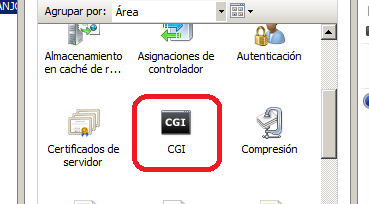
Descargamos la actualización y la instalamos.

Para facilitar la administración descargaremos **Administración pack for IIS.**

Instalamos Administración pack for IIS.



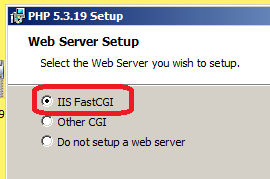
Una vez instalado nos saldrá una opción nueva en la **Administración de ISS**

****

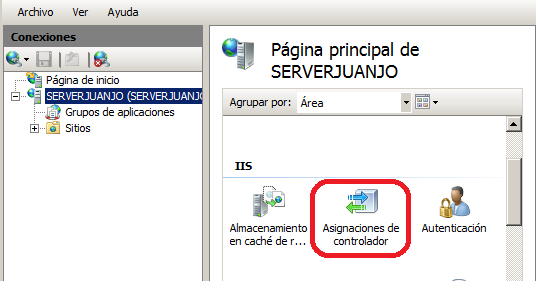
Esta herramienta nos facilitara la instalación de los binarios de PHP. Ahora toca descargar PHP desde la página oficial. A nosotros nos interesa el **Non Thread Safe** ya que será más eficiente

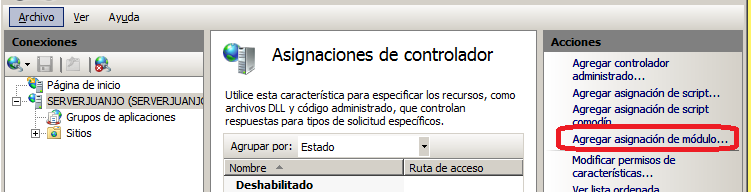


Una vez descargado lo instalamos.

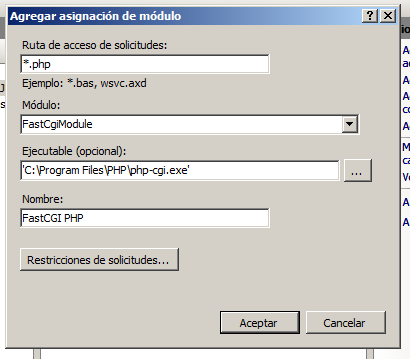


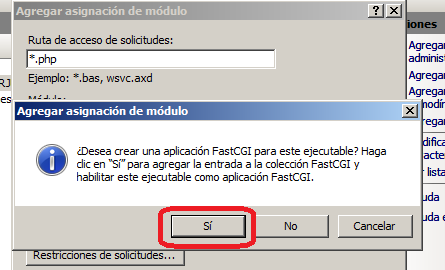
Ahora nos queda configurar IIS para que las peticiones que vayan a los ficheros \*.php pasen por el ejecutable que acabamos de instalar de PHP. Para ello nos vamos a **Administración de IIS / Asignaciones de controlador.**





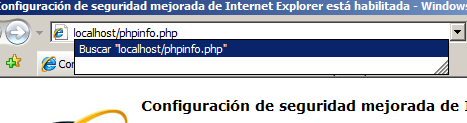
Configuramos para que todos los **\*.php** lo utilice el modulo **FastCgiModule**



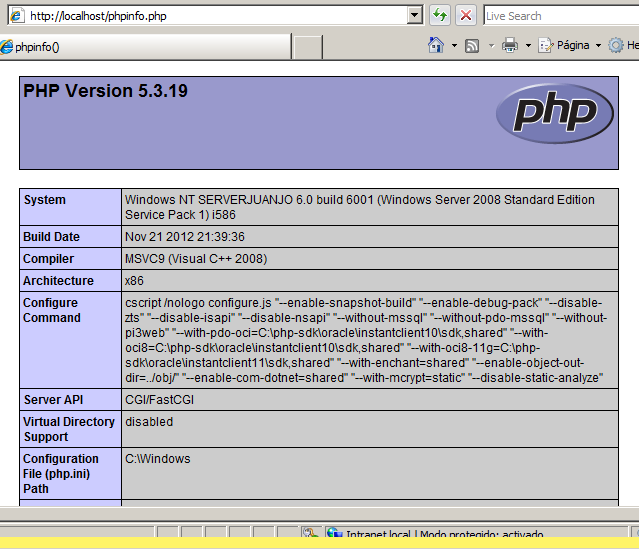


Para comprobar que todo funciona correctamente vamos a crear un archivo en el directorio **C:\inetpub\wwwroot** llamado **phpinfo.php**

Seguidamente nos vamos al browser y ponemos la siguiente dirección:



Nos mostrara la página de la información de PHP.

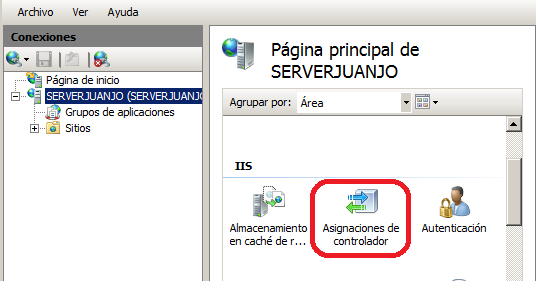


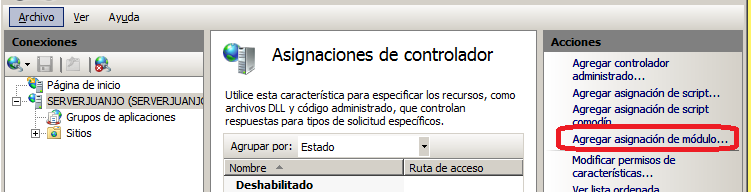
Tendremos configurado en nuestro servidor el PHP.

APACHE en Ubuntu

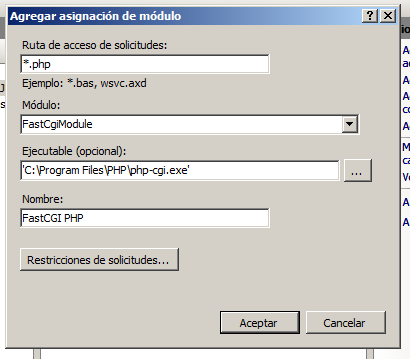
1. Configura Apache para que interprete correctamente las páginas creadas en PHP y comprueba que funciona correctamente.

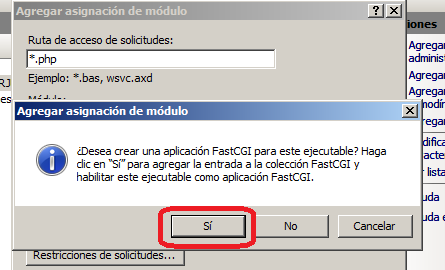
Ahora nos queda configurar IIS para que las peticiones que vayan a los ficheros \*.php pasen por el ejecutable que acabamos de instalar de PHP. Para ello nos vamos a **Administración de IIS / Asignaciones de controlador.**





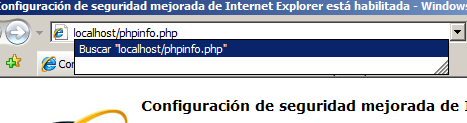
Configuramos para que todos los **\*.php** lo utilice el modulo **FastCgiModule**



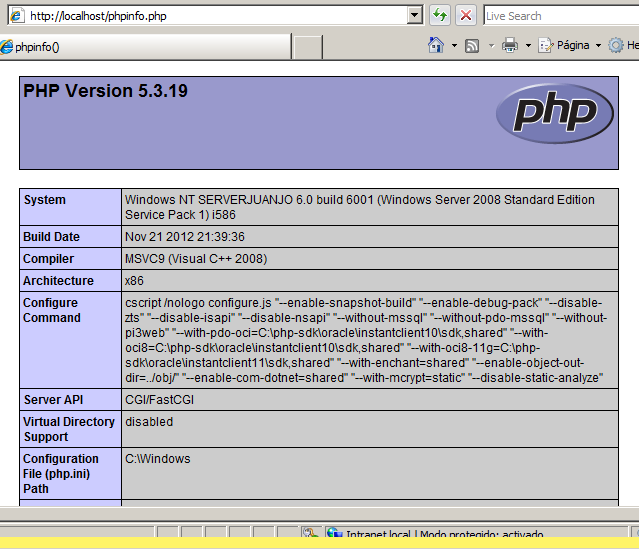


Para comprobar que todo funciona correctamente vamos a crear un archivo en el directorio **C:\inetpub\wwwroot** llamado **phpinfo.php**

Seguidamente nos vamos al browser y ponemos la siguiente dirección:



Nos mostrara la página de la información de PHP.



Tendremos configurado en nuestro servidor el PHP.

1. Cambia la configuración de Apache para que cumpla los siguientes requisitos:
   1. El puerto de escucha será el 8080
   2. La dirección de correo del administrador será [administrador@ejemplo.com](mailto:administrador@ejemplo.com)
   3. El número máximo de clientes conectados sea 50
   4. Cuando se acceda al directorio de una web devuelva por defecto el fichero principal.htm o el fichero indice.htm
2. Configura el servidor Apache para que de acceso a tres sitios. El primero se ubicará en el directorio /var/www/first.com, el segundo en /var/www/second.com y solo permitirá el acceso al usuario “curro”. Finalmente, el tercero se ubicará en el directorio /var/www/third.com, pero hará referencia a modo se directorio virtual a una carpeta ubicada en el equipo de un compañero.
3. ¿Cómo se podría configurar un servidor con Linux instalado para utilizar como directorio raiz el utilizado por un servidor Windows?; ¿Y cómo se podría configurar un servidor Windows para utilizar como directorio raiz el utilizado por un servidor Linux?

**Actividades con Servidores Web y gestores de contenido:**

1. Instalación en Windows y en Linux de un gestor de contenidos utilizando como soporte de información MySQL.
   * Deberá documentarse el proceso de instalación de los módulos/servicios necesarios, por ejemplo (MySQL, Apache, PHP, etc.)
   * Cada persona deberá utilizar un gestor de contenidos diferente (Moodle, Joomla, …)
   * Se valorará la elección de un gestor de contenidos Java como Liferay.
2. Instalar un servidor web java, por ejemplo TomCat en Windows y Linux. Deberá documentarse los módulos/servicios necesarios.
3. Configurar Apache y TomCat (u otro servidor web java) para que se comuniquen entre ellos. Deberá documentarse el proceso seguido.

**Formato de entrega:**

Documento en formato xhtml 1.0 ó pdf con enlaces a elementos multimedia, que resuelvan las cuestiones planteadas.