

## Caso práctico

Julián, el director del centro IES Montes Claros, mantiene una animada conversación con Alberto, el encargado de mantenimiento.

-Me gustaría poder tener una página web del instituto a disposición de todos los miembros del centro –dice Julián.

-Eso sería muy interesante, Julián. Se podría pensar en utilizar la página web, por ejemplo, como plataforma de todas las actividades que se realizan en el centro.

-Sí, Alberto, también estaba pensando en poder compartir fechas del calendario escolar o normas de convivencia del centro.

-Existe un montón de información que se puede exponer a la comunidad educativa. Y yo creo que la manera más atractiva sería a través de una página web –aclara Alberto.

-¿Cómo habrá que hacer para poder tener una página web? ¿Dónde reside físicamente? – Pregunta Julián.

-Pues tenemos dos opciones, Julián. Por un lado, se puede tener un servidor web en uno de los equipos del centro. Y por otro lado, se puede contratar los servicios externos de una empresa que nos aloje la página web -respondió Alberto.

-Como ya tenemos un servidor en el centro, que además tiene conexión a Internet, ¿se podrá alojar ahí la página web?

-Es bastante probable que sí se pueda. Pero para asegurarme voy a llamar a Maria, que ella me echará una mano.

-Cuando sepas algo en firme, me vas informando. ¿De acuerdo, Alberto?

-Así lo haré, Julián.



[jmerelo \(CC BY-NC-ND\)](#)

Como has leído en el caso práctico, es muy útil poder tener una página web al alcance de todo el mundo para dar a conocer toda la información necesaria sobre un tema. Supongo que sabrás que, una página web no es más que un conjunto de ficheros relacionados entre sí a través de enlaces o hipervínculos. Para que esta página web pueda ser accedida por cualquier equipo conectado a una red, estos ficheros han de permanecer guardados en un ordenador que cumpla el rol de servidor web.

Precisamente en este tema, vas a aprender a instalar y manejar un servidor web. Para ello, vas a estudiar cómo se instalan y configuran distintos servidores web, concretamente, uno en la plataforma Microsoft Windows y otro en la plataforma Linux.

## Debes conocer

Antes de comenzar con el grueso del tema, te recomiendo que visites esta presentación sobre los conceptos generales del protocolo HTTP

**Presentación general de la Unidad 5.**

[https://www.youtube.com/embed/wHXRmq75N\\_Y](https://www.youtube.com/embed/wHXRmq75N_Y)



[Ministerio de Educación y Formación Profesional](#) (Dominio público)

**Materiales formativos de FP Online propiedad del Ministerio de Educación y Formación Profesional.**

[Aviso Legal](#)

# 1.- Alojamiento web.

## Caso práctico

Como sucede en otras ocasiones, Alberto no se ve capaz de encarar la tarea de instalar y configurar un servidor web por su cuenta. Es por esto que recurre a la inestimable ayuda de María. Ella es técnica superior en Administración de Sistemas Informáticos en red y trabaja en la empresa que realiza el mantenimiento de los equipos del centro.



[Stockbyte](#). (Uso educativo nc)

-Hola María.-Hola Alberto, cuanto tiempo sin hablar.-Pues sí. Como te puedes imaginar -dijo Alberto-, te llamo porque se nos ha ocurrido a Julián y a mí que sería interesante tener una página web del centro con la información más relevante de toda la comunidad educativa.-Sí, desde luego que es muy buena idea. ¿Habéis pensado que alojamiento web necesitáis? -preguntó María.-Pues precisamente, para eso mismo te llamaba. En este tema estoy un poco perdido y no sé que opciones reales contamos para llevar a cabo esta tarea, -respondió Alberto.-No te preocupes. Lo primero que tenéis que decidir es si queréis que el alojamiento web se lleve a cabo en un servidor del instituto, o preferís contratar los servicios externos de una empresa por un precio al año -planteó María.-¿Te importa que quedemos un día -sugirió Alberto- y me expliques todo esto más despacio? Creo que es demasiada información la que tengo que procesar y veo que tú controlas del tema.-Desde luego, ¿qué día te viene bien para que me pase por el centro? -preguntó María.

Como acabas de leer en el anterior caso práctico, la elección de un alojamiento web es muy importante. En primer lugar, un proveedor de alojamiento web es una empresa que alquila **espacio web** y **ancho de banda** para la publicación de sitios web. Se trata un ordenador remoto donde guardaremos los archivos que forman parte de nuestra web. La comunicación entre el ordenador donde creamos la página web con el servidor web se realizará a través de una herramienta **FTP**, siglas en inglés de Protocolo de Transferencia de Ficheros.



Photodisc. (Uso educativo nc)

A la hora de decidir que tipo de alojamiento web se necesita, es necesario saber los requerimientos de la página en cuestión. No es lo mismo alojar una página web personal que la página web de una gran empresa. En este último caso, el volumen de información y la cantidad de accesos al servidor será mucho mayor. Por tanto, el servidor que maneje dicha información deberá disponer de más recursos.

Como puedes imaginar, una web con miles de visitas al día no puede permitirse la no disponibilidad del servicio durante unas horas porque las pérdidas serían inadmisibles. El alojamiento a contratar deberá ofrecer más calidad que, por ejemplo, una página personal que diseñes para colgar tus apuntes. Por todo esto, existen diferentes tipos de alojamiento web.

- ✓ **Alojamiento compartido gratuito:** normalmente dispone de poca funcionalidad y estabilidad.
- ✓ **Alojamiento compartido:** es el más frecuente. En este caso el sitio web está en un servidor que aloja muchos otros sitios web.
- ✓ **Servidor dedicado:** el proveedor se encarga de la administración del servidor.

A la hora de elegir un alojamiento web debes tener en cuenta las opiniones y experiencias de otros usuarios y usuarias, cuyos planes te pueden interesar. Si alguna vez tienes que buscar una empresa para realizar un alojamiento web para una página, busca en Internet distintas opciones y opiniones de diversas procedencias. Puede ser de mucha utilidad contar con la experiencia de los demás.

## Para saber más

Si quieres ampliar tus conocimientos sobre este tema, te invito a que visites el siguiente enlace donde se explica con mayor detenimiento los distintos tipos de alojamientos web que puedes encontrar en Internet.

[Artículo sobre Alojamiento Web.](#)

# 1.1.- Conceptos iniciales.

Es interesante que conozcas una serie de términos relacionados a la **World Wide Web (WWW)**, también conocida como la Web. Conocer estos términos te ayudará a desenvolverte con soltura cuando manejes textos y manuales relacionados con el tema de navegación web.

En primer lugar, fíjate en la definición de World Wide Web. **La WWW, o la web**, se compone de un sistema de distribución de información basado en hipervínculos que permiten su visualización en un ordenador conectado a la red Internet. Con un navegador, un usuario o usuaria visualiza sitios web compuestos de páginas web y navega a través de ellas usando hiperenlaces.



[Svilen.milev. \(CC BY-SA\)](#)

Lee con atención los siguientes términos asociados a la navegación web por Internet. Como podrás observar, cada uno de los términos descritos está relacionado con los demás.

- ✓ **Servidor web:** ordenador remoto que tiene instalados los programas necesarios para enviar los documentos requeridos por un ordenador cliente.
- ✓ **Cliente:** cualquier aplicación que sirve para solicitar páginas web, como por ejemplo, los navegadores.
- ✓ **Navegador web:** software instalado en el ordenador, que visualiza documentos en formato web (HTML) y permite el funcionamiento de los hiperenlaces. Los navegadores más utilizados son Internet Explorer, Opera, Mozilla Firefox, Chrome y Netscape.
- ✓ **URL:** acrónimo inglés del término Uniform Resource Location, que se traduce como “Localizador de recurso uniforme”. Es la dirección textual de un servidor web para la apertura de un archivo HTML.
- ✓ **IP:** acrónimo inglés del término Internet Protocol, que se traduce como “Protocolo de Internet” Es un código identificativo de un equipo informático en Internet formado por cuatro números, separados por puntos, cuyo valor no puede exceder de 255.
- ✓ **DNS:** acrónimo inglés de Domain Name System, que se traduce como “Sistema de nombres de dominio”. Es una base de datos que relaciona una dirección textual de un servidor con su dirección IP. Esta base de datos se encuentra en unos servidores DNS que deben ser especificados en nuestra conexión a Internet.
- ✓ **HTTP:** acrónimo inglés de Hypertext Transfer Protocol, que se traduce como “Protocolo de transferencia de hipertexto”. Al encabezar una URL con este término, el servidor reconocerá que se le está asociando un documento web.
- ✓ **HTML:** acrónimo inglés del término Hypertext Markup Language, que se traduce como “Lenguaje de marcado de hipertextos”. Los archivos creados en este formato tienen extensiones HTML ó HTM.

## Autoevaluación

¿Cómo se llama el ordenador que almacena la relación existente entre una URL y su dirección IP?

- Servidor Web.
- Servidor DNS.
- Cliente.
- Navegador Web.

No es correcta es concepto.

Muy bien. Así es como se llama.

Incorrecta, porque el cliente no tiene esa función.

El navegador ni siquiera es un ordenador. Repasa bien la teoría.

# Solución

1. Incorrecto
2. Opción correcta
3. Incorrecto
4. Incorrecto

## 1.2.- Funcionamiento de la web.

Una vez que ya manejas los conceptos iniciales de la web, ya estás en condiciones de entender el funcionamiento del sistema de visualización de documentos en la World Wide Web. Para hacer más sencillo de entender todo el proceso, se enumeran los pasos de cómo se actúa desde que el cliente solicita una página web en la barra de direcciones del navegador, hasta que ésta se visualiza en la ventana del navegador.

**Paso 1.** Desde tu ordenador ejecutamos el **programa navegador** de Internet y escribes la dirección de una página web. La dirección que se inserta se conoce como URL (por ejemplo, <http://www.google.es>).

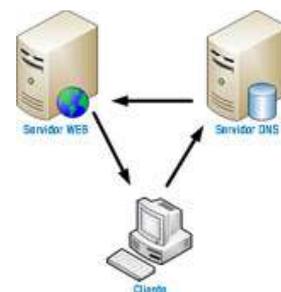
**Paso 2.** El servidor DNS que tengas configurado en tu conexión a Internet **localiza** la dirección IP que corresponde al **ordenador servidor** donde se encuentran los archivos web requeridos.

**Paso 3.** Como la URL que has tecleado comienza por **http://**, el ordenador servidor reconoce que estás solicitando algún documento web y nos envía un documento con extensión HTML ó HTM. Si en la dirección URL no se especifica un documento html concreto, el servidor buscará en la carpeta de la web y te enviará por defecto el archivo nombrado como **index.html**.

**Paso 4.** El programa navegador estudia el documento HTML recibido y **solicita al servidor los archivos** con los que esté enlazado, como imágenes, vídeos, animaciones, etc.

**Paso 5.** Finalmente, el programa navegador **compone la página** con los textos, enlaces, imágenes y animaciones. Es importante que al elaborar una página web compruebes si se visualiza correctamente en los distintos navegadores, ya que puede ocurrir que no se vea de igual manera en todos.

Con estos pasos que acabas de repasar, se ha conseguido abrir una página web. Estas se caracterizan por contener enlaces a otros archivos en el mismo servidor o en otros servidores. A estos enlaces se les conoce por hipervínculos o hiperenlaces. El funcionamiento de un hipervínculo consiste en ejecutar todo el proceso explicado anteriormente a partir de la dirección URL incluida en dicho enlace. Tan solo haciendo clic con el ratón en el hipervínculo, comienza todo el proceso de carga de un nuevo archivo.



Nuria Celis Nieto. (Uso educativo nc)

### Debes conocer

En el siguiente vídeo verás una animación sobre qué es y para qué sirve DNS.

**¿Qué es y para qué sirve DNS?**

<https://www.youtube.com/embed/dIGxJCqLJIY>

## 2.- El servidor web IIS.

### Caso práctico



[Stockbyte](#). (Uso educativo nc)

María, la Técnico Superior en Administración de Sistemas Informáticos en red, se encuentra reunida con Alberto, el encargado de mantenimiento de los equipos informáticos del IES Montes Claros. Están hablando de cómo solucionar el problema del alojamiento de una página web en el centro. -Mira, Alberto. La posibilidad que podemos barajar –sopesó María- sería tener un equipo en el centro que hiciera las veces de servidor de toda la red de ordenadores del instituto. -Eso me parece una buena idea, María –respondió Alberto.

Rápidamente a Alberto le surgió una duda.

-¿Pero tenemos todo lo necesario para poder hacerlo desde el instituto? -Yo, en este caso, me decantaría por la instalación de un sistema operativo de red. Nos permite la posibilidad de instalar un servidor web de manera rápida y sencilla. Atenderíamos las restricciones de acceso a distintas partes con esta solución –concretó María. -Muy bien –aceptó Alberto-. Estaría encantado de aprender a instalar y configurar ese sistema operativo y hacerlo funcionar como servidor web. -Una opción sería que nos decidiésemos por un sistema operativo propietario, de la familia de Microsoft –propuso María. -¿Podríamos hacer pruebas? Es para hacerme una idea de la complicación de la tarea –respondió Alberto prudentemente. -Por supuesto que sí. Vamos a ponernos manos a la obra -decidió María.

Como has podido comprobar, Alberto y María han tenido una conversación de lo más interesante acerca de que posibilidades existen para instalar un servidor web. La opción que están teniendo en cuenta es la de instalar un sistema operativo de la familia de Microsoft, como puede ser el Windows Server 2012 o Windows Server 2016.

En los siguientes capítulos aprenderás a instalar y configurar este sistema operativo para que realice las funciones que estamos describiendo.

Existe la posibilidad de descargar una versión gratuita de Windows Server desde la página oficial de Microsoft, válida para 60 días, ideal para hacer prácticas.

## 2.1.- Instalación de Windows Server 2008.

---

A estas alturas, ya sabes que un servidor provee servicios a otros ordenadores clientes. La arquitectura cliente-servidor consiste básicamente en un cliente que realiza peticiones al servidor, que le da respuestas. Esta misma idea se puede aplicar a programas que se ejecutan en un solo ordenador.

Ventajas de utilizar un esquema cliente-servidor:

- La capacidad de proceso está repartida entre los distintos equipos.
- Mejora la organización de la información, por tenerla centralizada.
- La separación de responsabilidades facilita y clarifica el diseño del sistema.

En este caso, vamos a ver como se instala un sistema operativo específico para servidores, donde se incluyen una serie de servicios en red que podrás instalar y configurar. Utilizaremos como ejemplo el sistema operativo **Windows Server 2012**. El proceso es similar para otras versiones como la 2016 ó 2019.

### Debes conocer

Las versiones de evaluación de Windows Server pueden descargarse en el "Evaluation Center":

<https://www.microsoft.com/es-es/evalcenter/>

Windows Server 2012:

<https://www.microsoft.com/es-es/evalcenter/evaluate-windows-server-2012-r2>

Windows Server 2016:

<https://www.microsoft.com/es-xl/evalcenter/evaluate-windows-server-2016>

Windows Server 2019:

<https://www.microsoft.com/es-es/evalcenter/evaluate-windows-server-2019>

Para instalar el sistema será necesario crear una máquina virtual en VirtualBox y utilizar la imagen descargada como ISO para que el sistema arranque desde ella por primera vez. En nuestro caso utilizaremos Windows Server 2012, pero el proceso es similar en otras versiones.

A continuación hay que seguir los pasos de la instalación del sistema, tal como se ha realizado para poder trabajar en unidades anteriores y en otros módulos de este ciclo de Servicios Microinformáticos y Redes.

Una vez instalado el sistema, hay que activar el servidor web IIS, como veremos en el siguiente apartado.

### Debes conocer

En el siguiente enlace tienes un vídeo donde se explica la instalación de un sistema operativo Windows Server 2012 en VirtualBox. Para la instalación de tu máquina virtual, es recomendable que sigas todos los pasos como están explicados en el vídeo, incluyendo las características de la máquina virtual como las opciones elegidas en la instalación del sistema operativo.

**Instalación de Microsoft Windows Server 2012 sobre VirtualBox.**

<https://www.youtube.com/embed/tVWJQ6egNcc>



## 2.2.- Instalación del servidor web IIS.

Antes de empezar la instalación del servidor Web en Windows Server 2012, veamos una pequeña introducción sobre el servidor IIS. Los servicios de Internet Information Server (o IIS), son los servicios de software que admiten la creación, configuración y administración de sitios Web, además de otras funciones de Internet. Es interesante que sepas que los servicios de Microsoft IIS incluyen el Protocolo de transferencia de archivos (FTP, o File Transfer Protocol, visto en la pasada unidad de trabajo 3), el Protocolo de Oficina de Correos (POP) o el Protocolo simple de transferencia de correo (SMTP), vistos en la pasada unidad de trabajo 4.

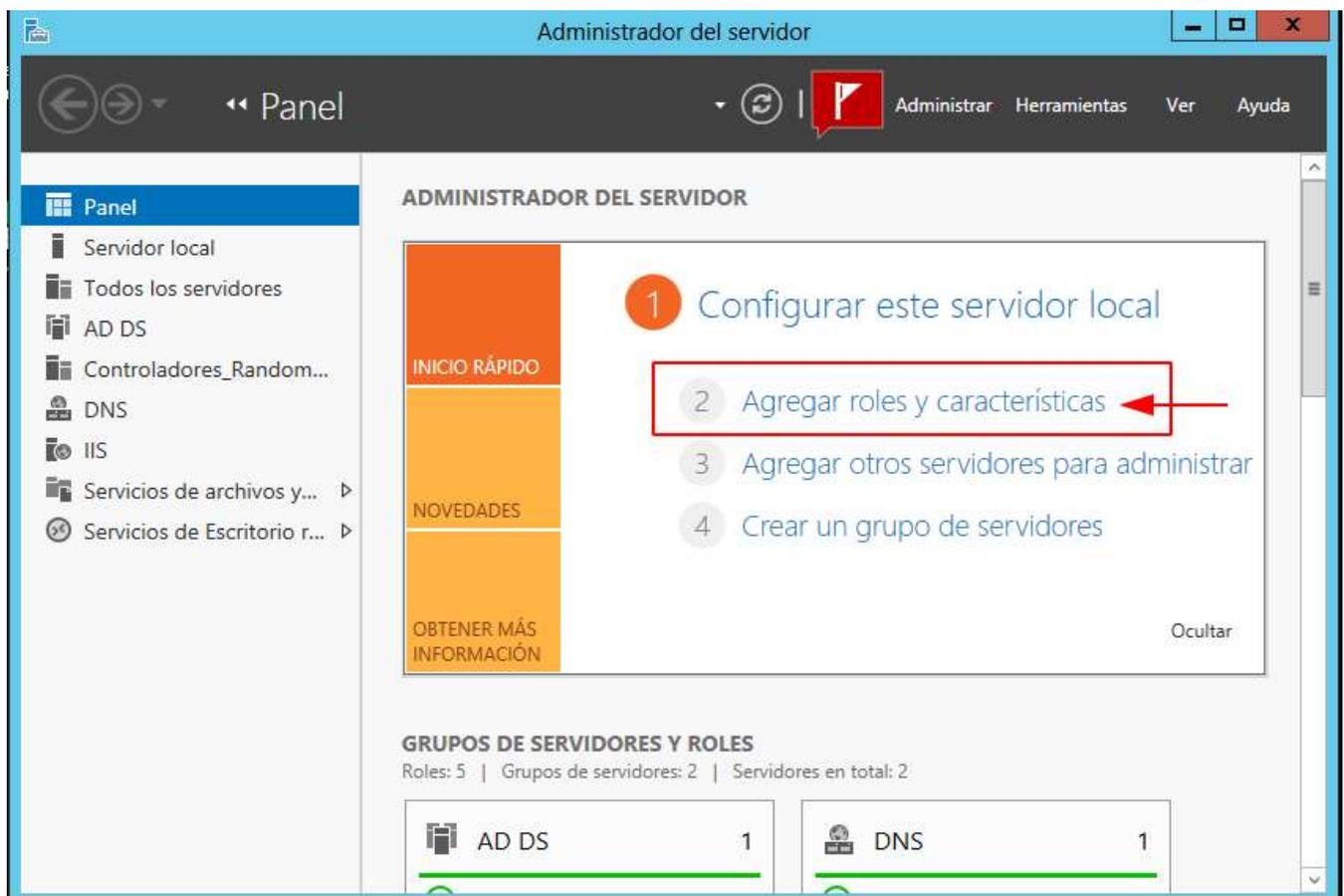
Además de estos protocolos, como características de IIS de Windows Server 2012, también encontrarás:

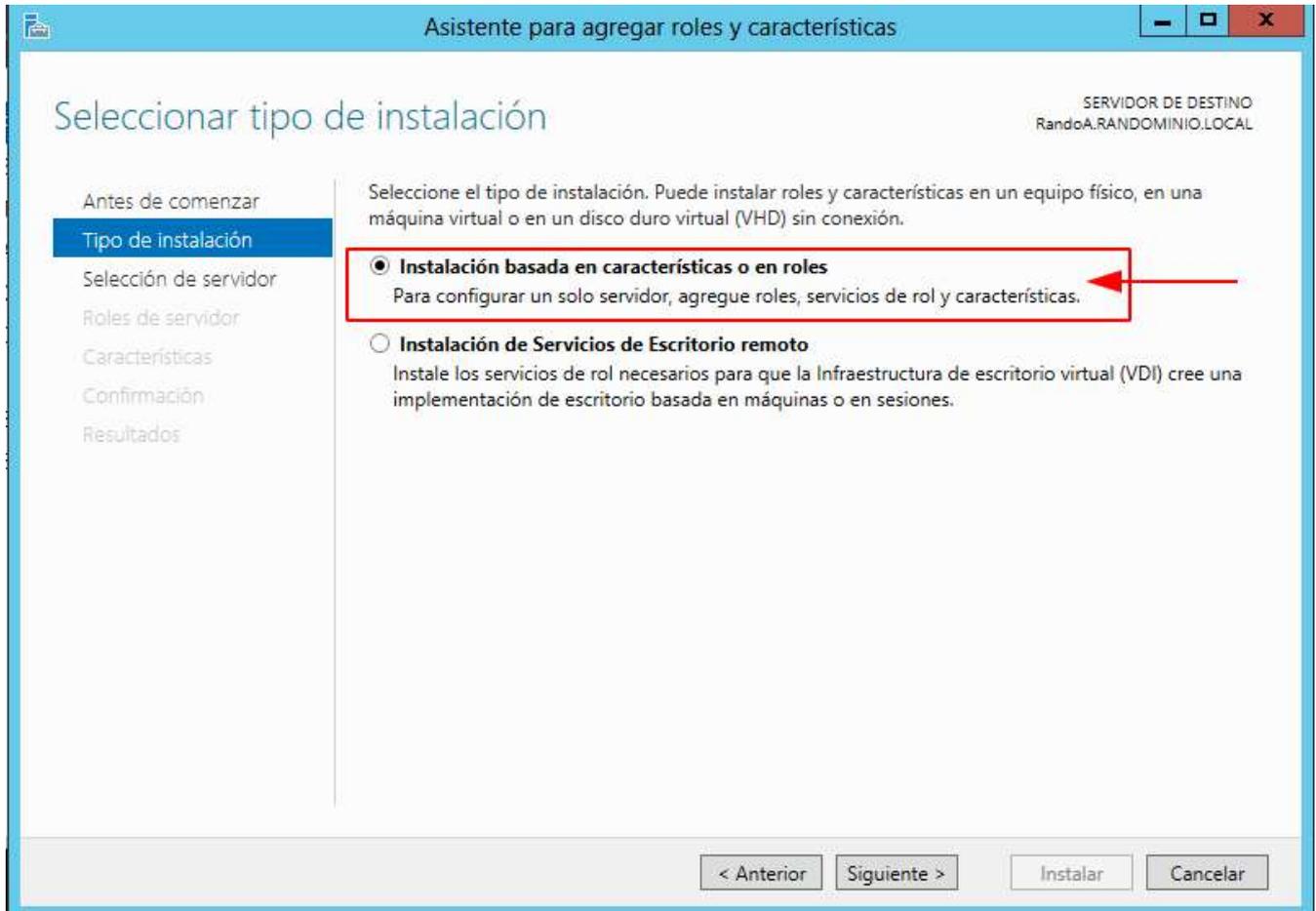
- La facilidad de la publicación de información en una Intranet o en Internet.
- Permite autenticación robusta y segura de los usuarios.
- Permite la creación de contenido dinámico en nuestras páginas Web, etc.

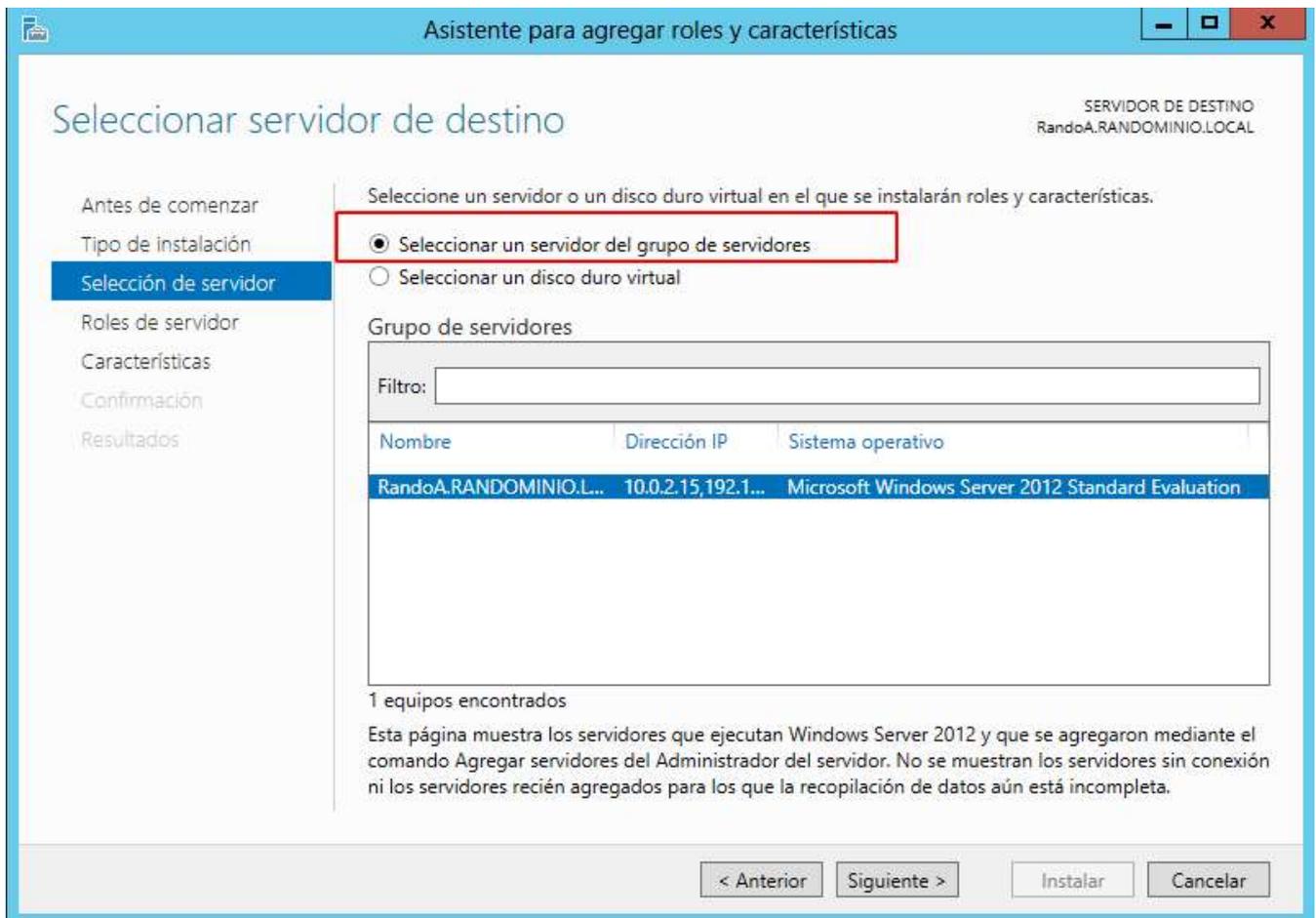
Sin duda, encontrarás el servidor IIS como una pieza clave para gestionar una red, pudiendo construir una Intranet con muchos servicios que habitualmente encontraremos en múltiples portales existentes en Internet.

Por último, es recomendable que instales el servidor IIS en un equipo Windows Server 2012 que NO sea controlador de dominio. Esto puede provocar un problema de seguridad, ya que los usuarios del dominio que no son administradores, podrían llegar a abrir una sesión local sobre el servidor, con el consiguiente problema de seguridad que ello supone. Si tú ya tienes una máquina virtual con este sistema operativo, y es controlador de dominio, puedes utilizarlos sin problemas para practicar, pero ten esto en cuenta en un supuesto real.

Partimos de la pantalla inicial de administración del servidor → Agregar roles o características:

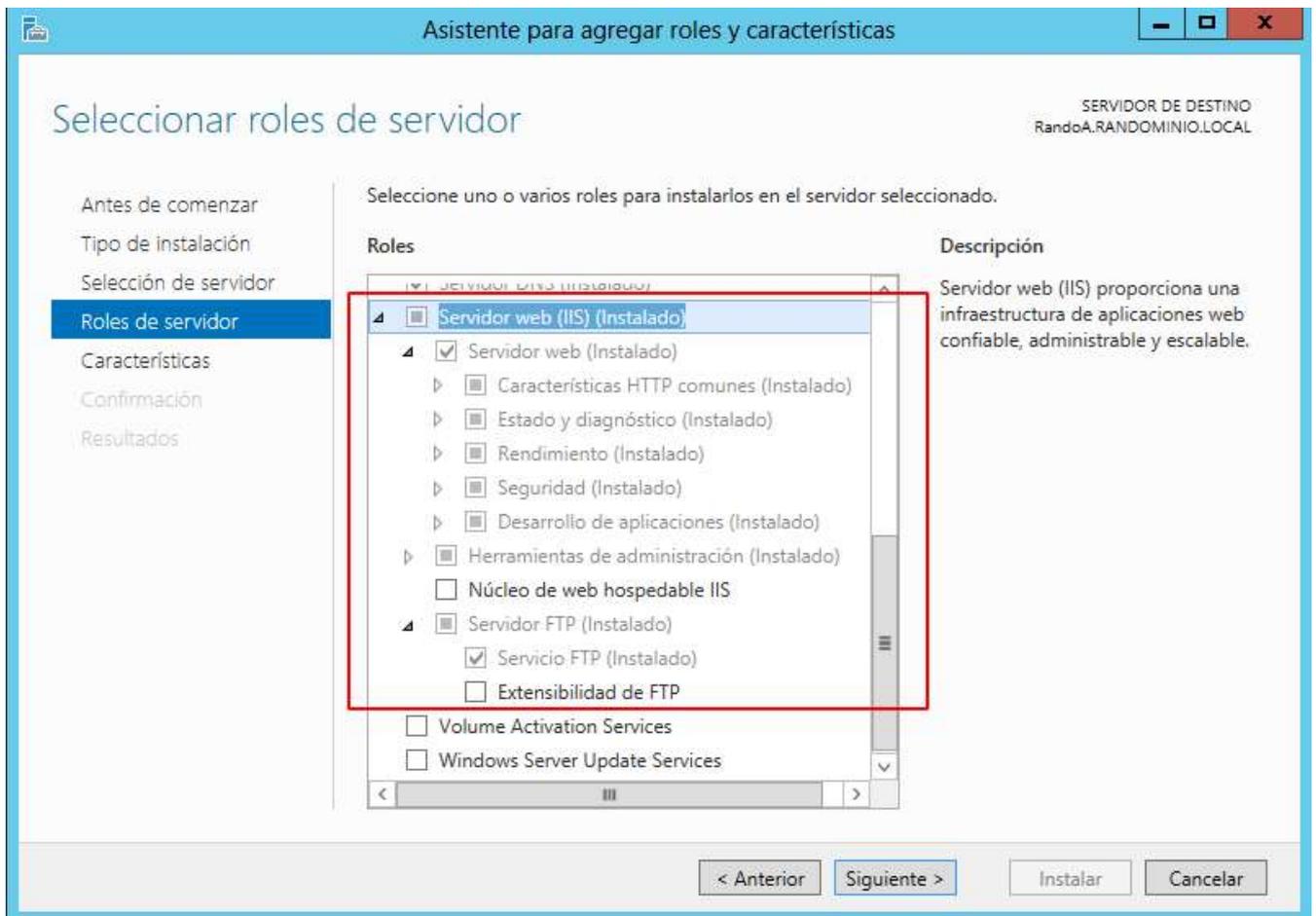






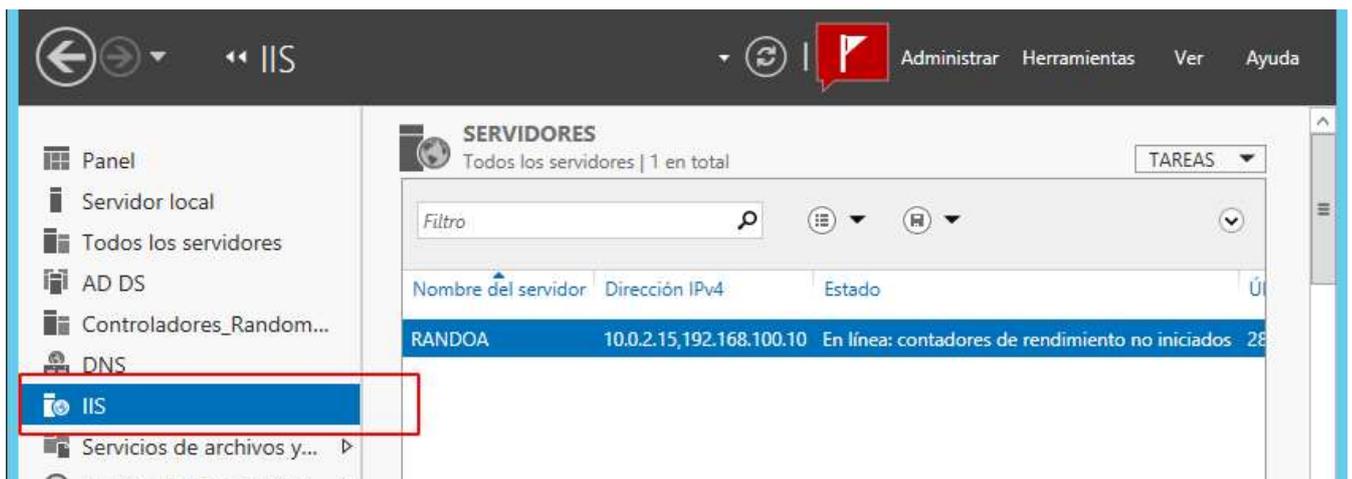
José A. Jiménez (CC0)

Con el rol de servidor web podemos activar diferentes opciones del servicio, así como del rol de servidor FTP, relacionado con el anterior.



José A. Jiménez (CC0)

Una vez instalado el rol de servidor web, podremos acceder a la ventana de administración del servidor, en la que nos aparecerá la nueva opción "IIS".



José A. Jiménez (CC0)

Para comprobar que el servidor web está instalado, abre una ventana del navegador en el servidor y escribe "localhost". Si el IIS aparece la pantalla de bienvenida significa que está bien instalado:



José A. Jiménez (CC0)

## Autoevaluación

Señala la afirmación correcta sobre el servidor IIS en Windows Server 2008:

- La instalación del servidor IIS se realiza a través del Panel de Control.
- No permite la creación de contenido dinámico en las páginas Web.
- Es obligatorio que el servidor sea un controlador de dominio.
- Incluye otros protocolos como FTP, POP o SMTP.

No es correcto. Se realiza desde el Administrador del Servidor.

Incorrecta, esa es una de sus características.

No es correcto, repasa la teoría.

Muy bien. Es una de sus características.

# Solución

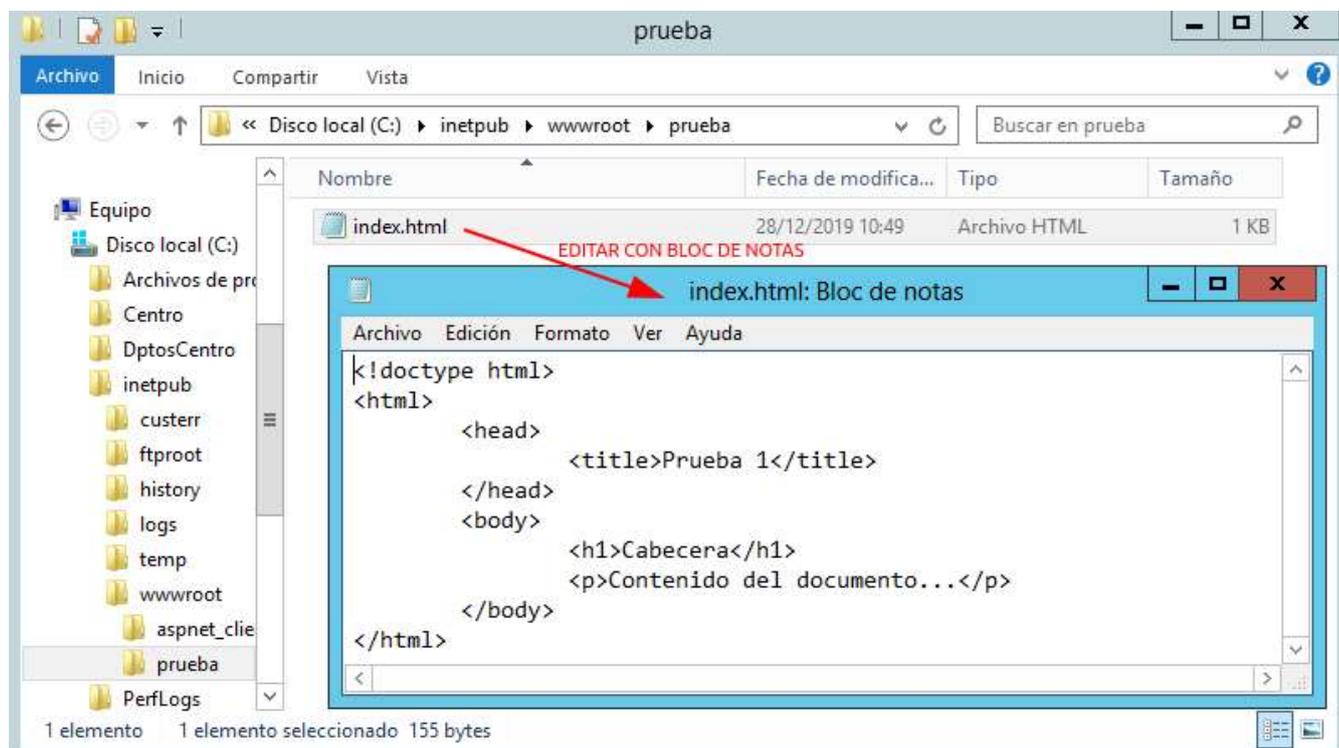
1. Incorrecto
2. Incorrecto
3. Incorrecto
4. Opción correcta

## 2.3.- Creación de distintos sitios web.

Una vez instalado IIS, tienes varias posibilidades para gestionar tu sitio web:

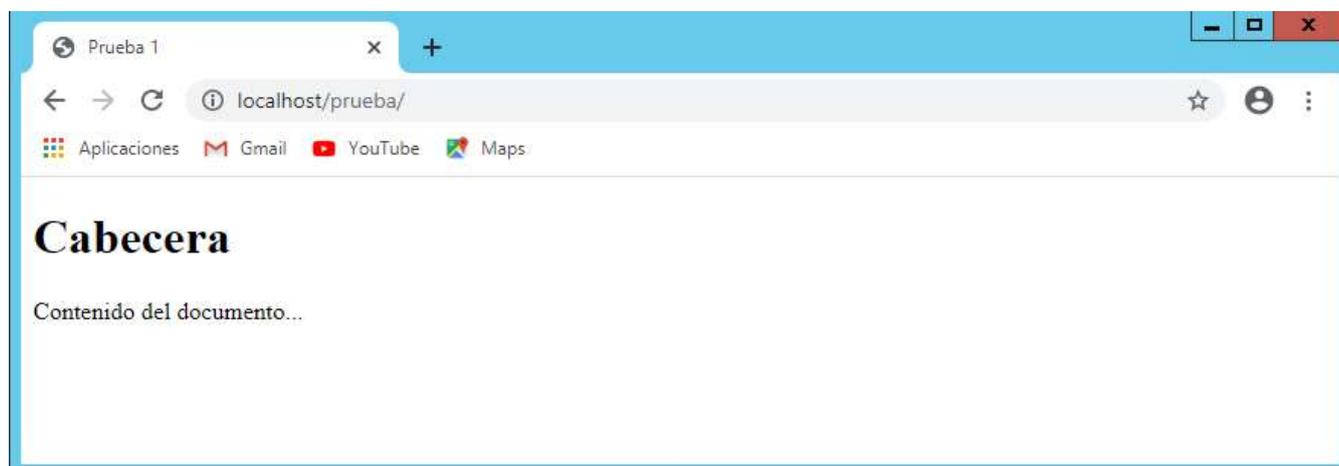
1. Puedes crear carpetas colgando del sitio Web predeterminado, que se ubica en “C:\inetpub\wwwroot”. Si creas carpetas dentro de “wwwroot”, serán accesibles desde el navegador.

Como ejemplo de este modo de trabajo, vamos a crear la carpeta “prueba” en esa ruta, y dentro de ella el archivo “index.html”, que utilizaremos para probar el servicio:



José A. Jiménez (CC0)

Si usamos el navegador para acceder a la carpeta “/prueba”, abrirá el archivo por defecto “index.html”. En el navegador será suficiente utilizar la URL <http://localhost/prueba> :



José A. Jiménez (CC0)

2. La segunda forma de organizar el sitio Web consiste en crear sitios Web independientes del que ofrece IIS por defecto. Se pueden crear distintos sitios Web sobre la misma dirección IP en puertos distintos.

Vamos a explicar todo el proceso de creación de un sitio Web a través de un ejemplo. Se trata de alojar la información del centro IES Montes Claros. Para ello, crearemos dos sitios Web. El primero de ellos recogerá la

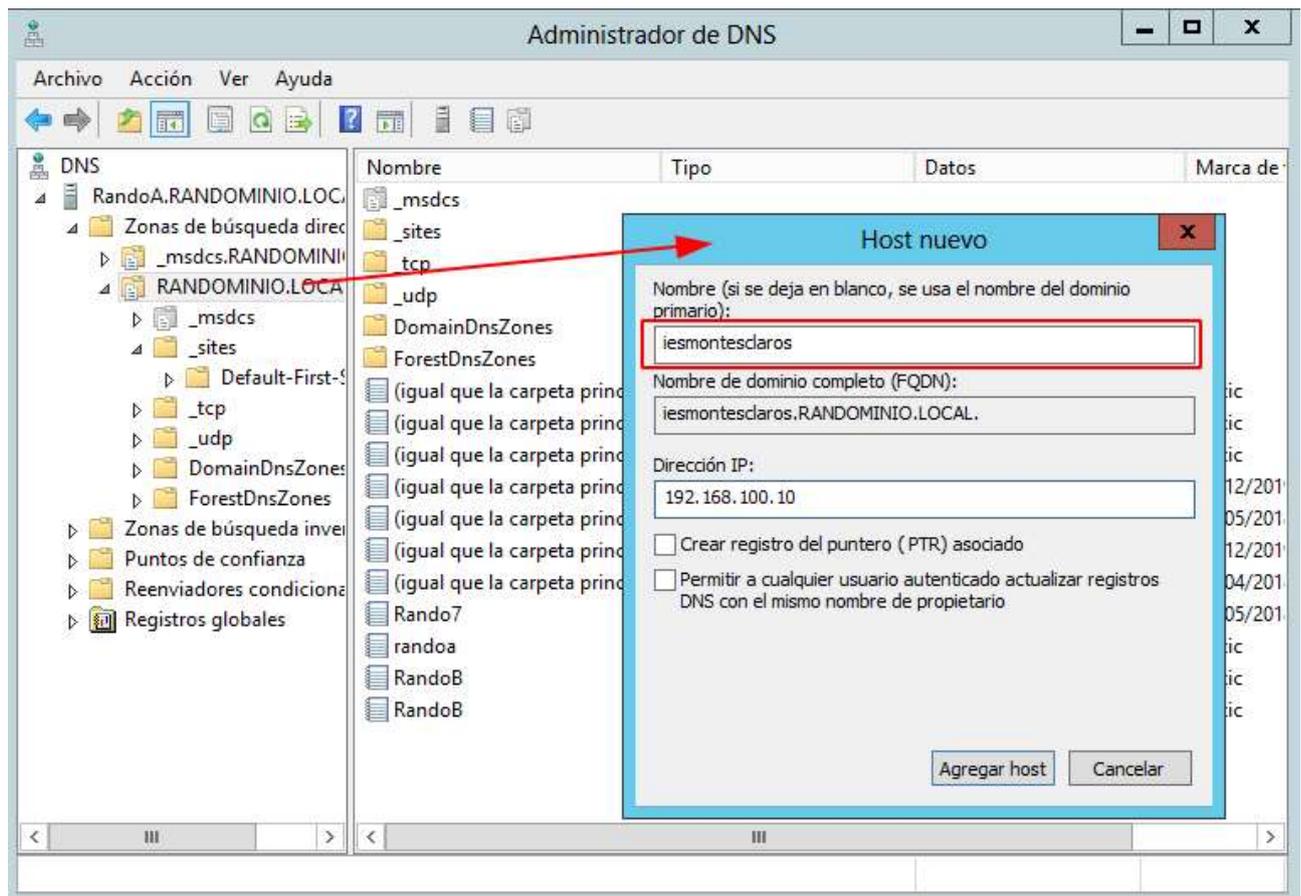
información del centro en general y el otro servirá para alojar la información propia de los departamentos del centro.

Para poder acceder a los nuevos hosts por su nombre, debemos integrarlos en la configuración de DNS. En el ejemplo seguido, estamos usando un servidor Windows que pertenece al dominio “randominio.local”, y este será el sufijo que tendrá nuestro servicio web. Si estuviéramos usando un servidor real ubicado en internet, correspondería a otro nombre de dominio y la URL de acceso sería la que correspondiera. En nuestro caso, como no estamos usando un servicio DNS ubicado en internet, haremos que el acceso sea local, con los siguientes nombres de servidor:

<http://iesmontesclaros.randominio.local>

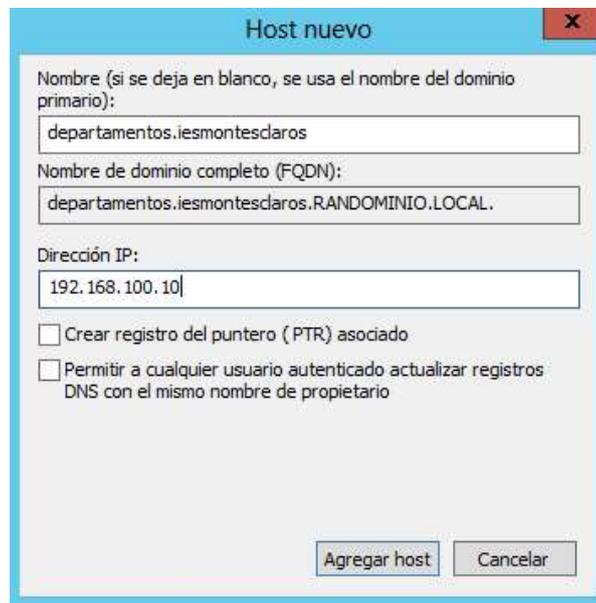
<http://departamentos.iesmontesclaros.randominio.local>

Para que funcione el acceso posterior desde el navegador, primero vamos a acceder a la configuración de DNS, añadiendo a las zonas de búsqueda directa los nuevos dominios “iesmontesclaros” y “departamentos”:



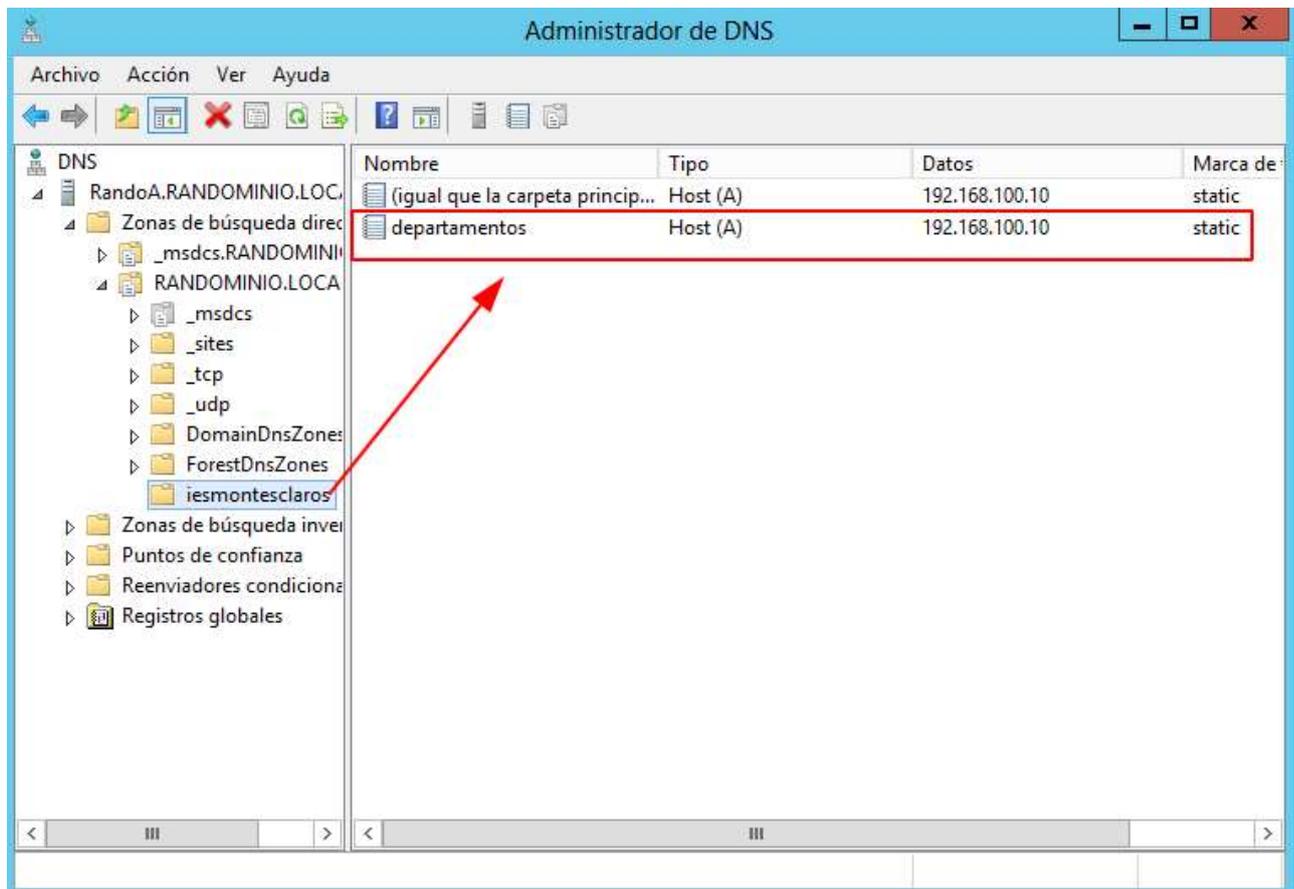
José A. Jiménez (CC0)

Repetimos la misma operación para añadir el nombre del subdominio “departamentos”. Con esta configuración DNS, nuestra máquina podrá reconocer los nuevos nombres.



José A. Jiménez (CC0)

El subdominio “departamentos” estará dentro de “iesmontesclaros”:



José A. Jiménez (CC0)

Podemos probar que estos nombres responden, haciendo un “ping” en la consola:

```
Administrador: Windows PowerShell
PS C:\Users\Administrador> ping departamentos.iesmontesclaros.randominio.local

Haciendo ping a departamentos.iesmontesclaros.randominio.local [192.168.100.10] con 32 bytes de
Respuesta desde 192.168.100.10: bytes=32 tiempo<1m TTL=128

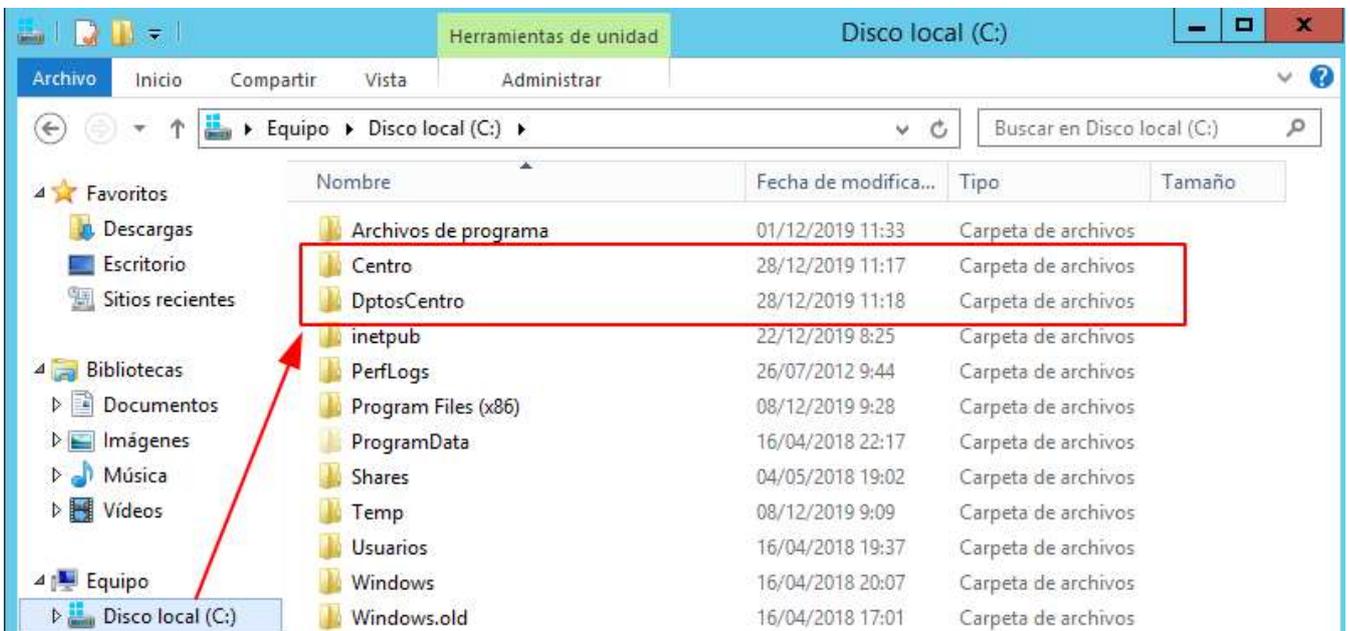
Estadísticas de ping para 192.168.100.10:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\Users\Administrador> ping iesmontesclaros.randominio.local

Haciendo ping a iesmontesclaros.randominio.local [192.168.100.10] con 32 bytes de datos:
Respuesta desde 192.168.100.10: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 192.168.100.10:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\Users\Administrador>
```

José A. Jiménez (CC0)

Ahora vamos a crear los contenidos de los dos servicios web. En primer lugar, hay que crear en la unidad “C:” del disco duro del servidor dos carpetas. Una de ellas con el nombre de “Centro” y otra denominada “DptosCentro”.

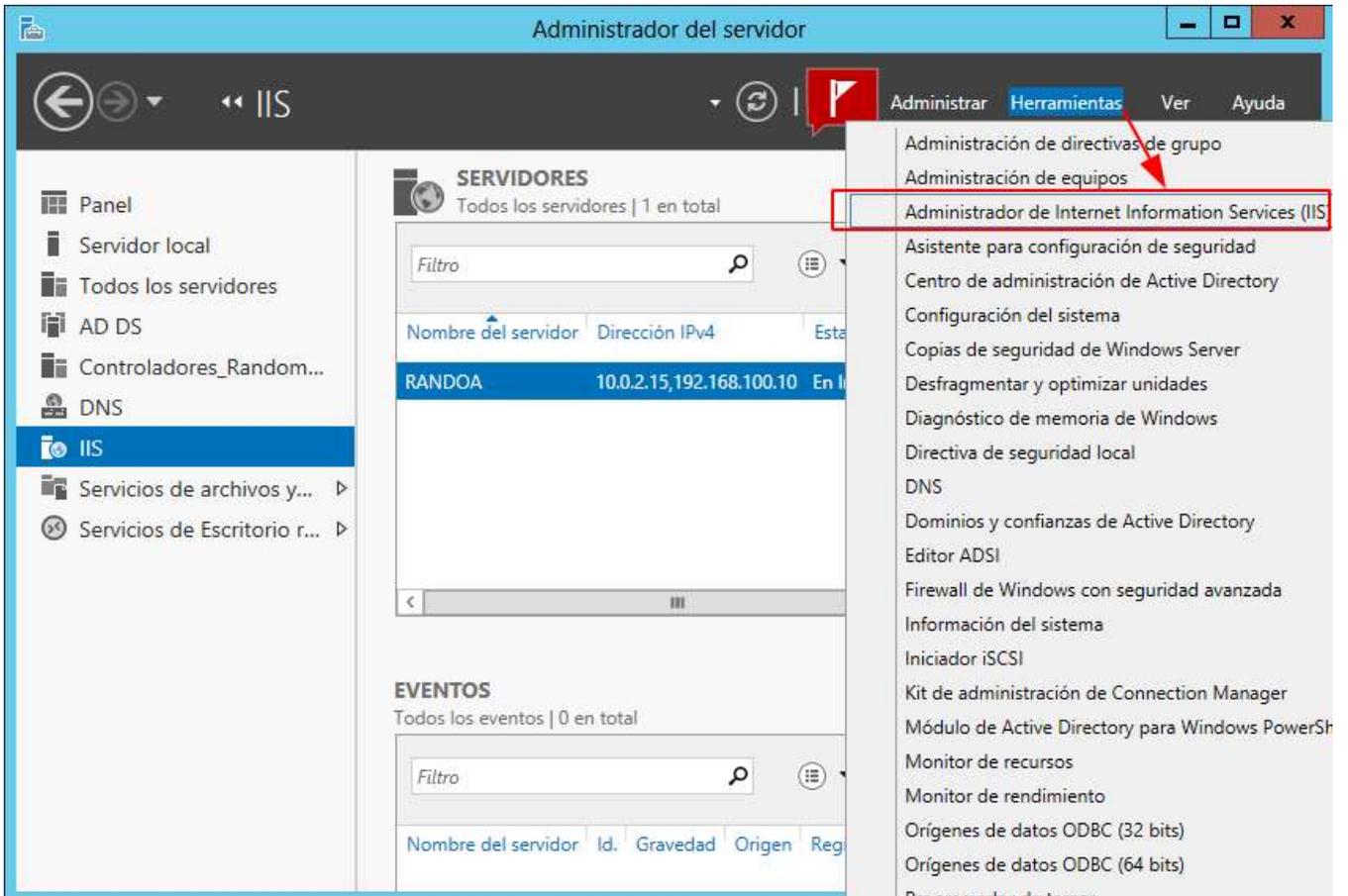


José A. Jiménez (CC0)

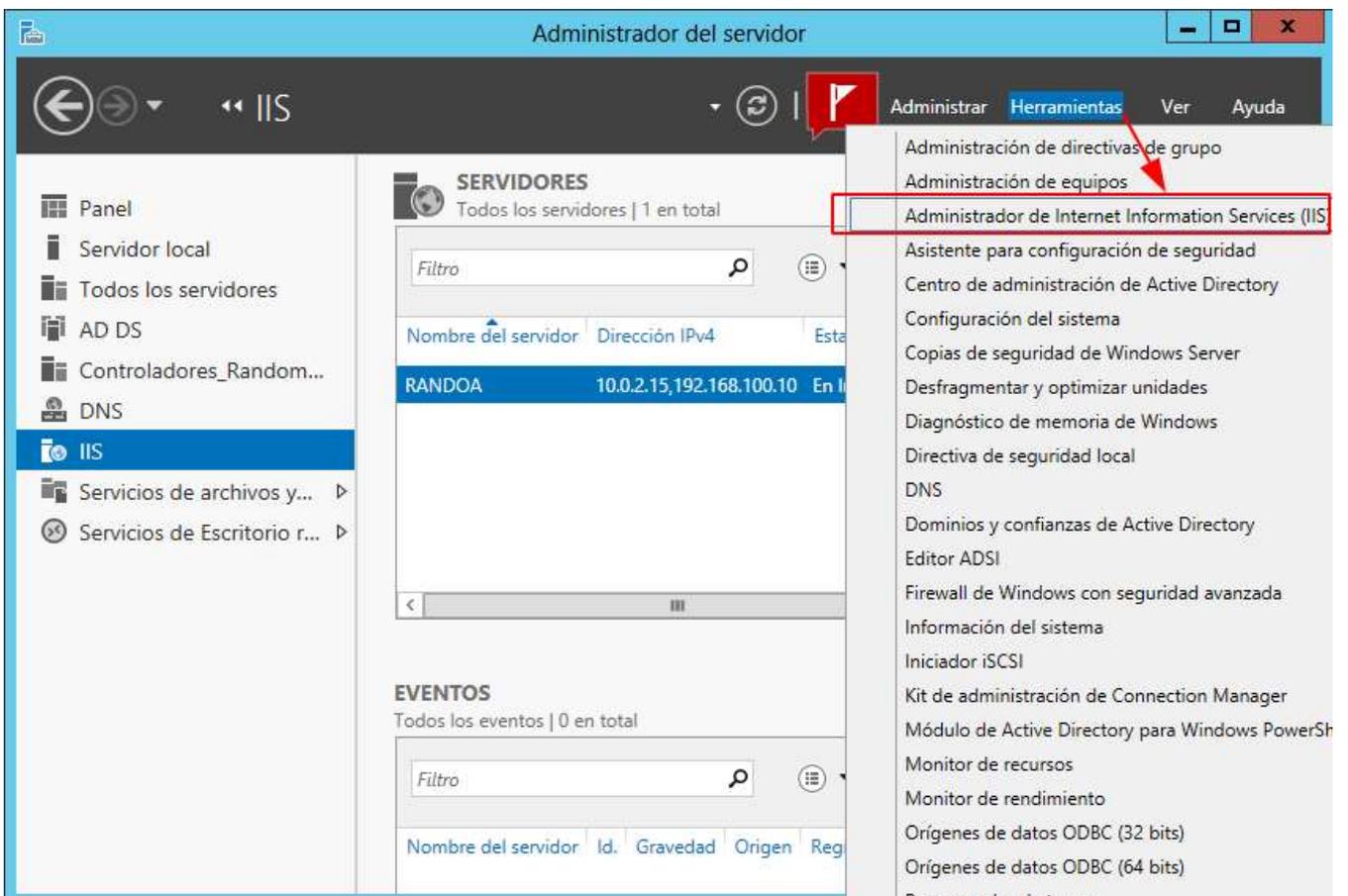
Ahora, debes acceder al panel de administración del servidor, desde la pantalla inicial de “Administrador del servidor” o bien desde el menú:

**Inicio -> Panel de Control -> Herramientas Administrativas -> Administración de Internet Information Services.**

Aparece una ventana como la siguiente:



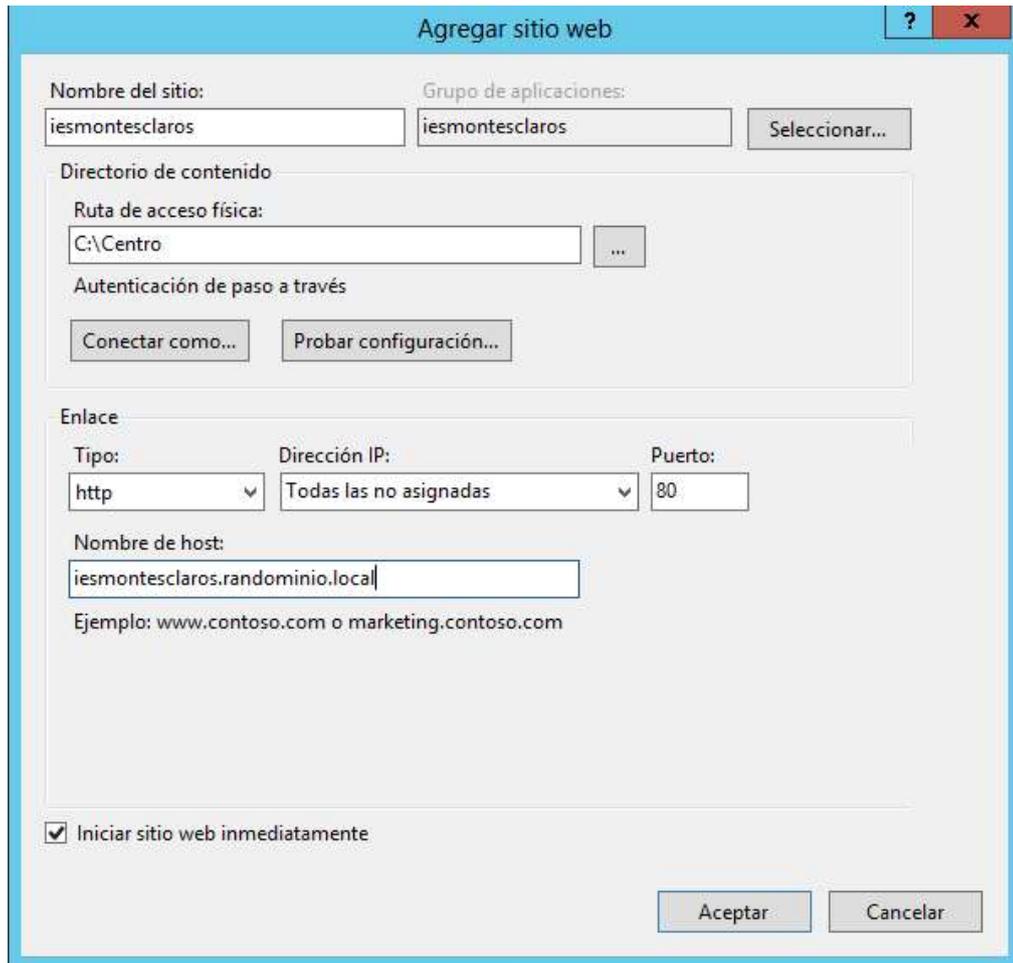
José A. Jiménez (CC0)



José A. Jiménez (CC0)

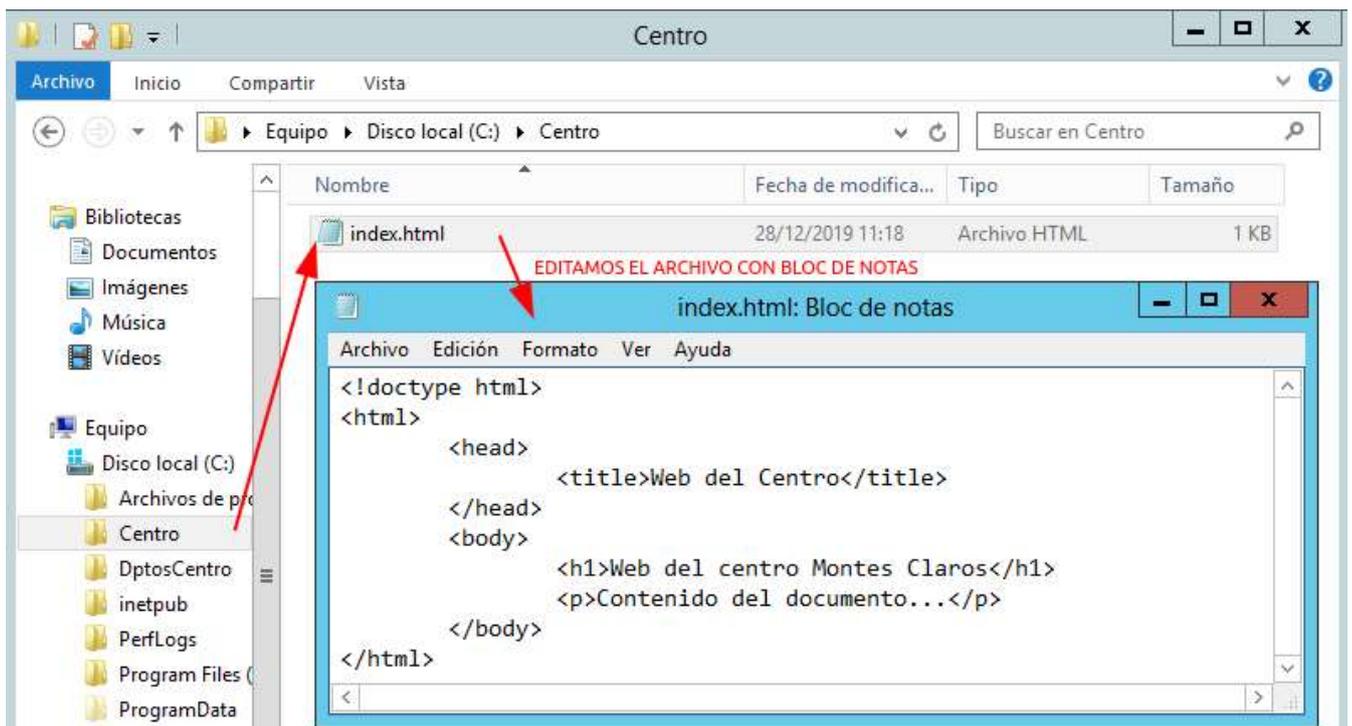
En el marco de la izquierda pulsamos sobre "Agregar sitio web". Añadiremos las dos carpetas creadas anteriormente para los sitios "iesmontesclaros" y el subdominio "departamentos". Para el segundo sitio

modificaremos el puerto, asignando el 81 en lugar del 80, que se usa por defecto:



José A. Jiménez (CC0)

Dentro de la carpeta "Centro" crearemos un archivo HTML básico, para comprobar que el servicio funciona bien:



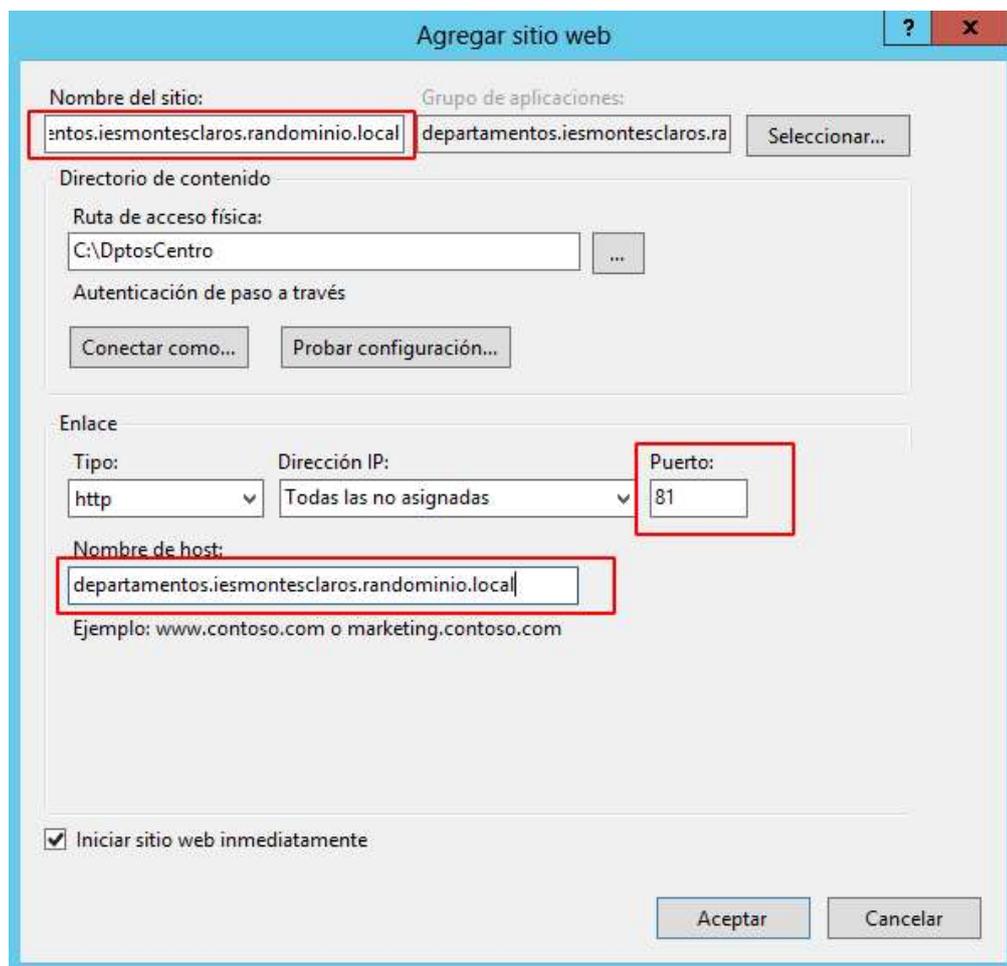
José A. Jiménez (CC0)

Si abrimos un navegador y accedemos a la URL de este servicio (<http://iesmontesclaros.randominio.local/>), tendremos lo siguiente:



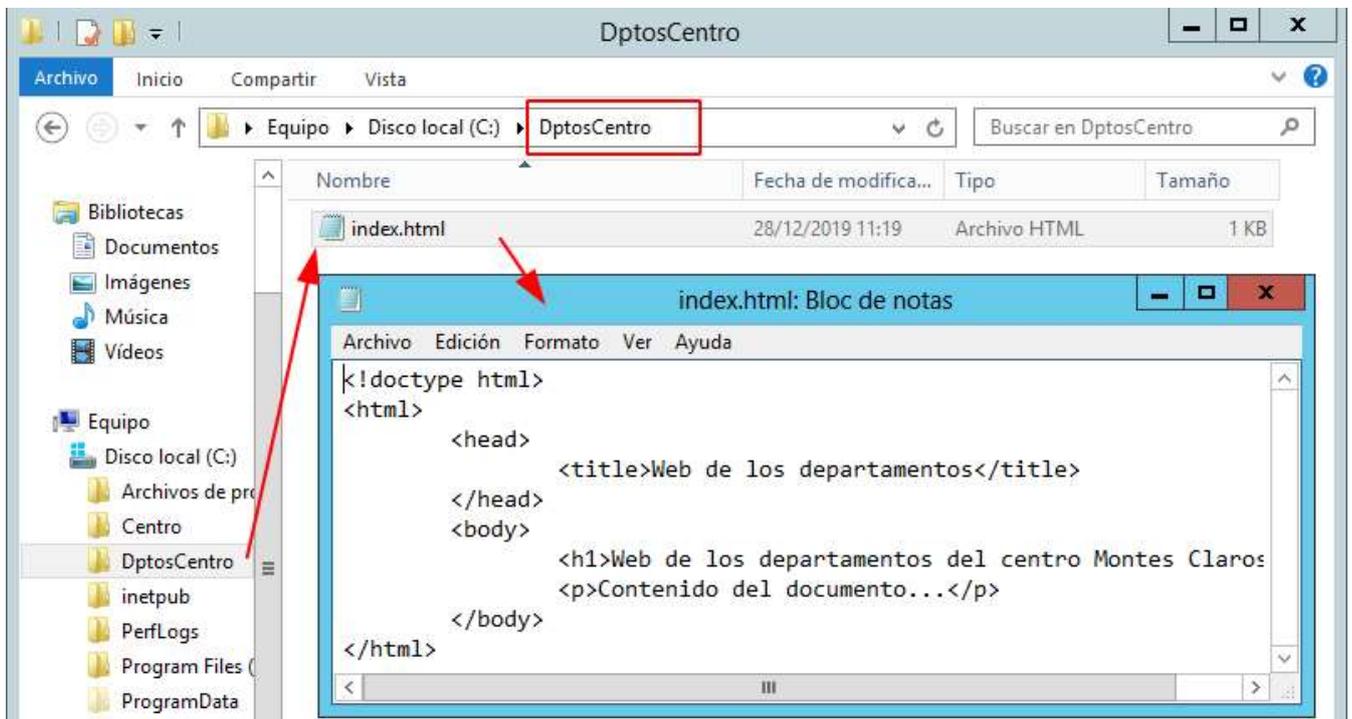
José A. Jiménez (CC0)

Ahora crearemos el servicio correspondiente al subdominio “departamentos”. En este caso, queremos acceder a él a través de un puerto específico, el 81:



José A. Jiménez (CC0)

Para probar el servicio, como en el caso anterior, crearemos un archivo HTML en la carpeta:



José A. Jiménez (CC0)

Si accedemos con un navegador a la URL <http://departamentos.iesmontesclaros.randominio.local:81>, es decir, especificando el puerto 81, tendremos lo siguiente:



José A. Jiménez (CC0)

Si no tuviéramos configurado el servicio DNS, también podríamos haber realizado todos los pasos anteriores y testearlo localmente, simplemente configurando los nombres de hosts en el servicio de nombres estático de Windows, **C:/Windows/System32/Drivers/etc/hosts**

En este fichero añadimos la siguiente información, suponiendo que la dirección IP de nuestro servidor es, por ejemplo, 192.168.100.10:

```
192.168.100.10 www.iesmontesclaros.com
192.168.100.10 departamentos.iesmontesclaros.com
```

## Autoevaluación

Señala la afirmación correcta sobre el servidor IIS en Microsoft Windows Server 2012.

- Se puede crear más de un sitio web con un servidor IIS.
- Podemos probar las páginas sólo con el nombre del sitio sin hacer ningún cambio de configuración.
- Si modificamos el fichero hosts, podremos hacer pruebas desde cualquier punto de la red.
- Si añadimos una entrada en el DNS, solo se pueden hacer pruebas locales.

Correcto.

Incorrecto. Es necesario configurar el servidor web para que presente las páginas

Incorrecto. El fichero hosts nos permite poner nombres a las IPs de ciertos hosts, pero para poder hacer pruebas es suficiente conocer la IP del servidor.

Incorrecto. El servicio DNS es independiente del servicio HTTP.

## Solución

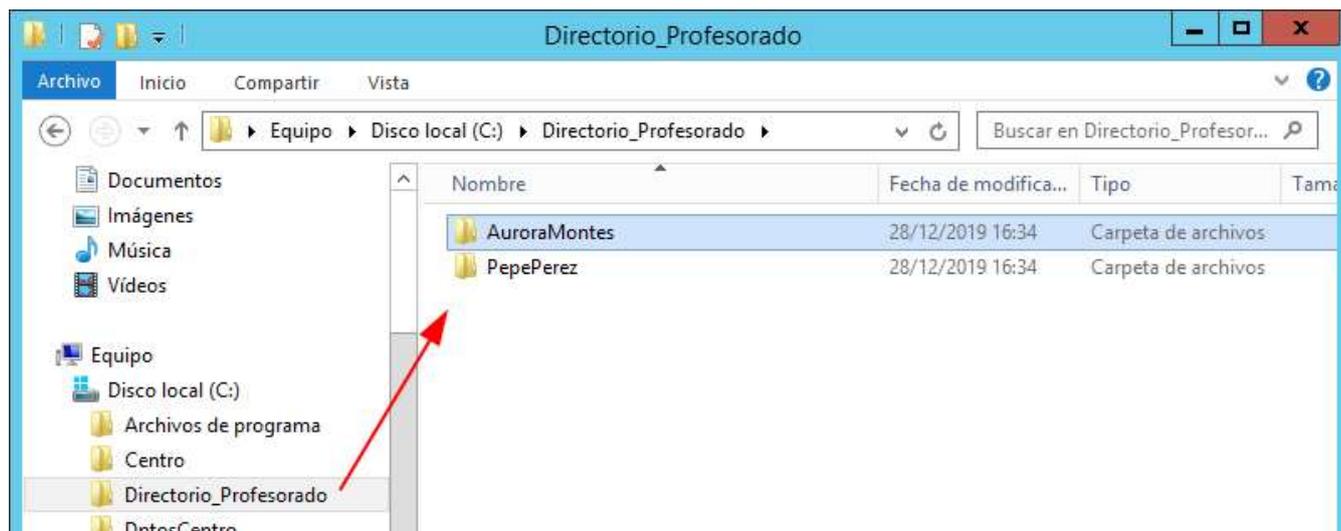
1. Opción correcta
2. Incorrecto
3. Incorrecto
4. Incorrecto

## 2.4.- Creación de directorios virtuales.

Un directorio virtual es aquel que aparece en los navegadores como si estuviera incluido en el directorio raíz del servidor web, aunque resida físicamente en otra parte.

Vamos a explicar la creación de directorios virtuales, aprovechando que tenemos un sitio web creado para los departamentos de un centro de enseñanza. Vas a definir un nuevo directorio virtual, al que le asignaras un nombre o alias. Pero los contenidos que incluyamos estarán en un directorio distinto al sitio web original. Desde el navegador parecerán que están dentro del sitio web elegido, aunque se aloja físicamente en otro lugar.

Crearemos, como aparece en la figura, una carpeta llamada “C:/Directorio\_Profesorado”. Aquí se ubicarán las páginas web de los profesores y profesoras interesadas:



José A. Jiménez (CC0)

Desde la dirección de la página, accederemos a las páginas web como si estuvieran ubicadas dentro de los departamentos del centro, pero esto no será así, puesto que convertiremos a este directorio en un directorio virtual que se llamará “Profesorado”.

Para acceder, por ejemplo, a la página de la profesora Aurora Montes, se accederá con la siguiente dirección: <http://departamentos.iesmontesclaros.com:81/Profesorado/AuroraConde>

Es importante que entiendas que, “Profesorado” no es una carpeta, sino un directorio virtual. Y se encuentra físicamente en “C:/Directorio\_Profesorado”, pese a que el sitio web de los departamentos está ubicado en “C:/DptosCentro”.

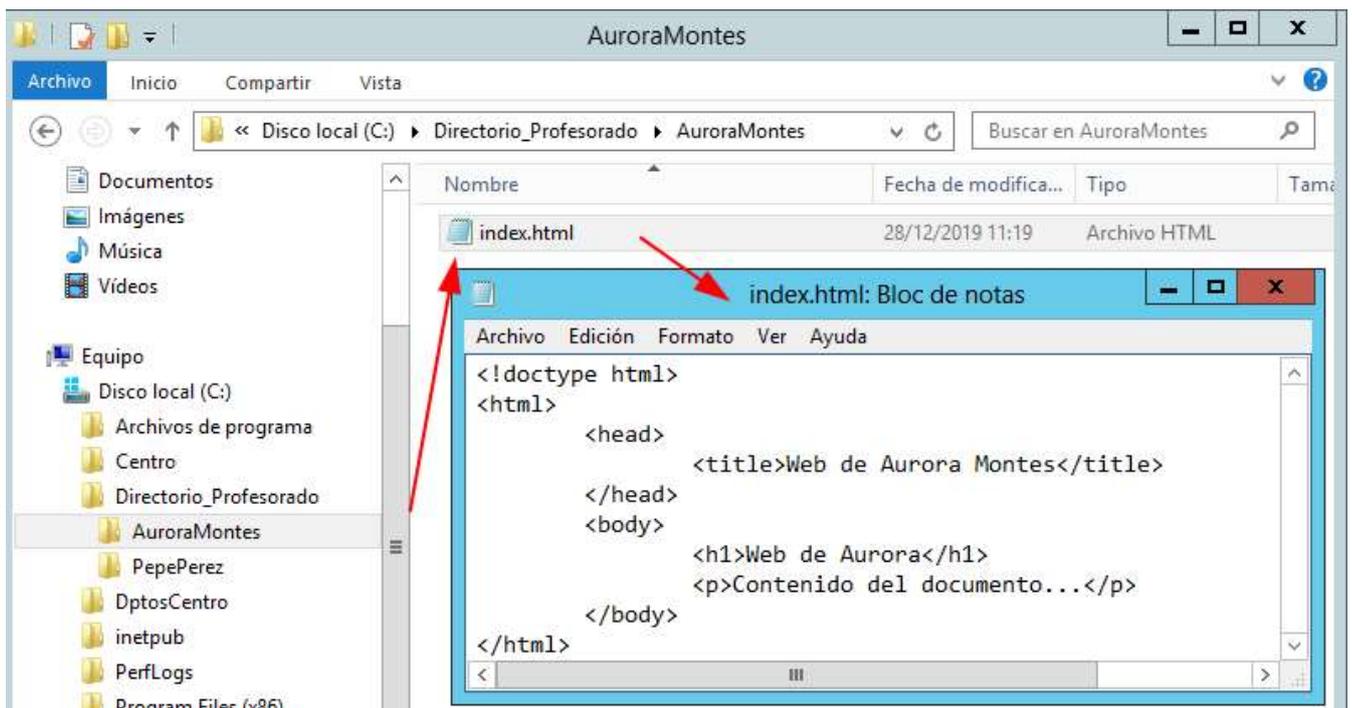
Una vez creada la carpeta “Directorio\_Profesorado” en “C:/”, vamos a asignarla. Abrimos **Inicio -> Herramientas Administrativas -> Administrador de Internet Information Services**. Encima del sitio departamentos, con el botón derecho, eliges “Nuevo Directorio Virtual”.

Rellenaremos los siguientes datos:



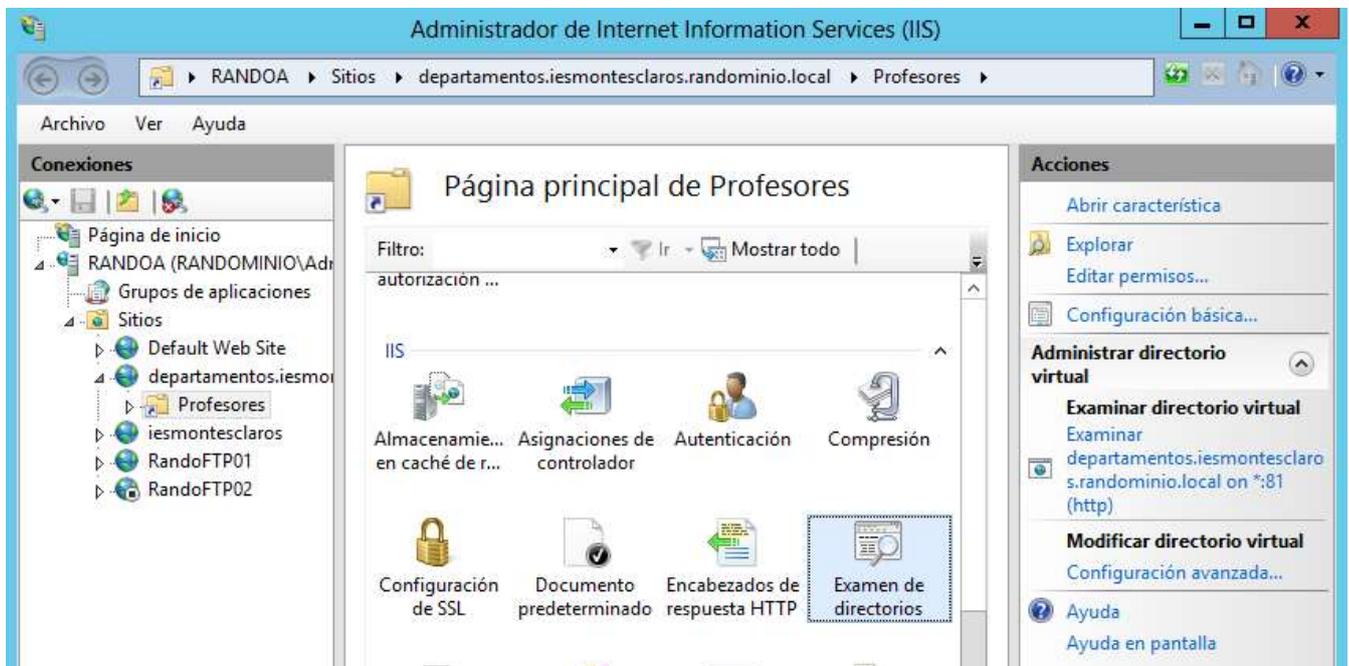
José A. Jiménez (CC0)

Para que la web de Profesores tenga contenido, creamos un archivo HTML en la carpeta de la profesora que acabamos de crear:



José A. Jiménez (CC0)

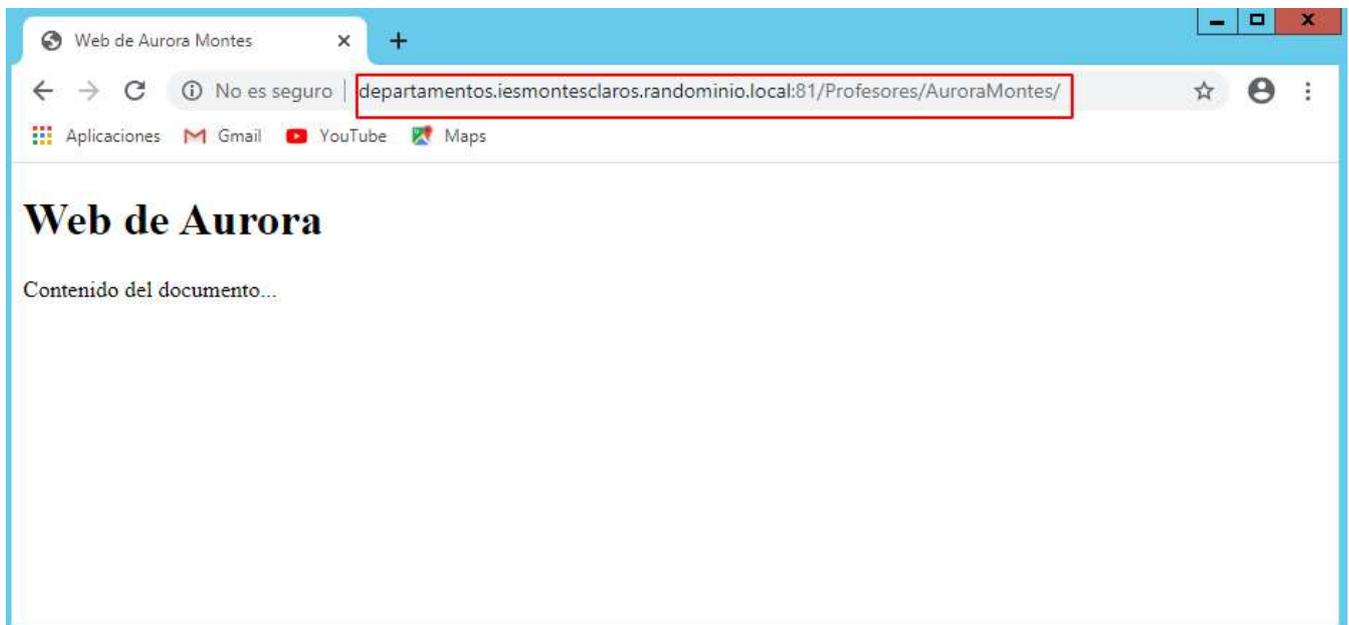
Y ahora habilitamos la “Exploración de directorios” para el servicio web:



José A. Jiménez (CC0)

Ahora tendremos habilitada la posibilidad de acceder a directorios virtuales que contienen diferentes servicios web, en este caso, las páginas de los profesores del centro.

URL: <http://departamentos.iesmontesclaros.randominio.local:81/Profesores/AuroraMontes/>



José A. Jiménez (CC0)

**Para saber más**

En el siguiente enlace tienes un artículo de Microsoft sobre la creación de directorios virtuales en el IIS de Windows Server.

[Creación y configuración de directorios virtuales.](#)

## 3.- Ampliación de las funciones del servidor web IIS.

### Caso práctico

Alberto, el encargado de mantenimiento de los equipos informáticos del IES Montes Claros, lleva unos días aprendiendo, de la mano de María, las funcionalidades básicas del servidor Web IIS. Pero Alberto, que es una persona muy curiosa, le sigue planteando más dudas.



[Stockbyte](#). (Uso educativo nc)

-La verdad, María, estoy aprendiendo mucho sobre este servidor Web.

-Muchas gracias, Alberto. Lo que has visto hasta ahora, es solo la punta del iceberg. Tiene un mundo lleno de posibilidades.

-Bueno, eso me lo puedo suponer. En un principio, lo que se me viene a la cabeza, sería el poder enviar información por la red de forma segura, por ejemplo.

-Sí, a ese tipo de cosas me refiero. Pero es que, además, se puede ir un poco más allá. Se podría instalar un servidor FTP para compartir datos con vuestros alumnos y alumnas a través de Internet –sugiere María.

-Eso sería también muy interesante. ¿Supondría mucha dificultad montar algo así en nuestro servidor Web? –preguntó Alberto.

-Yo si quieres te ayudo a instalar estas funciones adicionales –dijo María-, te enseñé como funcionan, y tú decides después que será lo mejor para las necesidades de vuestro centro.

-Eso sería estupendo, la verdad –responde con entusiasmo Alberto.

-Pues nada, nos ponemos a trabajar ahora mismo, Alberto.

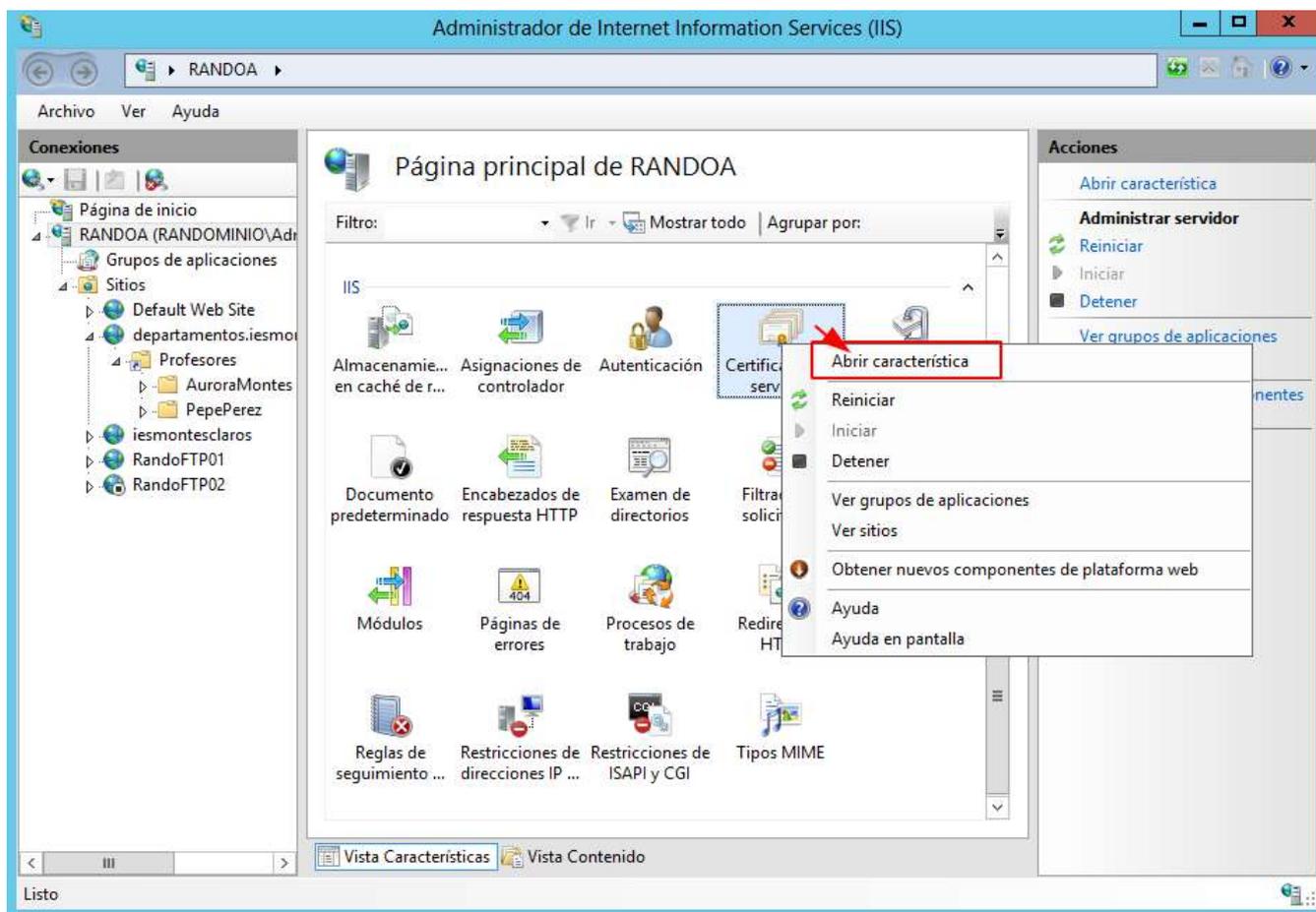
Acabas de leer que, Alberto, es una persona de lo más curiosa. El mundo de la informática le apasiona, y las carencias que tiene sobre ciertos temas, las suple pidiendo ayuda y mostrando interés cuando le explican cosas nuevas. En realidad, así es como tiene que ser una persona que le interese este mundo. No es necesario saber todo de todos los campos de la Informática, pero sí es necesario que te intereses y tengas ganas de aprender de cualquier área.

Como recordarás de los apartados anteriores, las funcionalidades básicas del servidor Web IIS están muy bien si controlas todo lo que has visto. Pero no creas que todo termina ahí. Este servidor Web tiene más características. Como la instalación de conexiones seguras utilizando el protocolo HTTPS o la utilización de un servidor FTP. Espero que todos estos temas te resulten interesantes, ya que son aplicaciones del servidor Web que, sin ser básicas, son muy utilizadas e importantes.

## 3.1.- Servidor de certificados y de carpetas seguras.

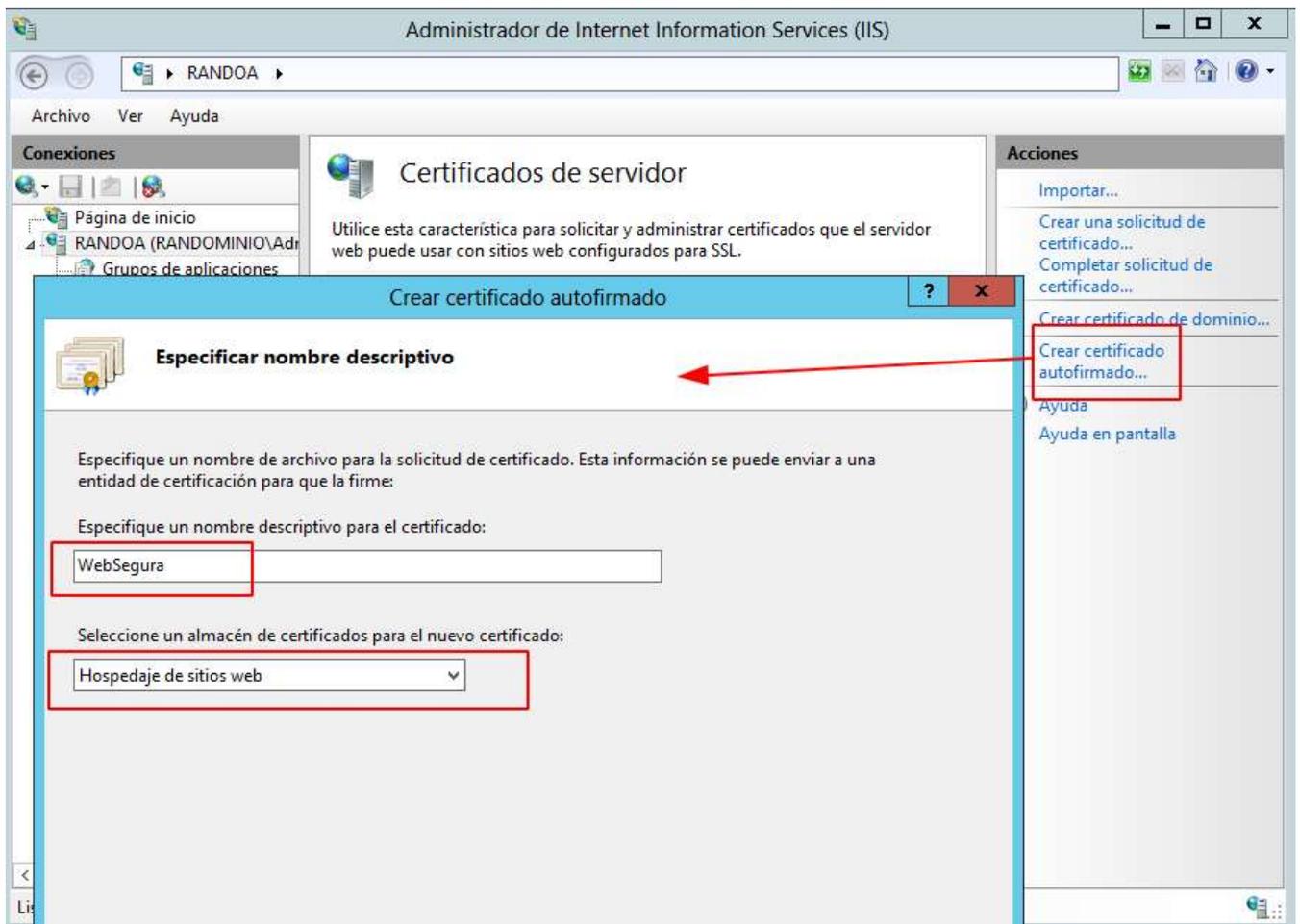
En primer lugar, tienes que saber que un certificado permite confiar en la entidad que ha emitido dicho certificado. Cuando vas a confiar en una entidad emisora de un certificado, significa que esa entidad tiene las directivas adecuadas para evaluar las solicitudes de certificados. Aprenderás a emitir un certificado para los equipos y usuarios del servidor que tenemos instalado. Con este certificado, podremos utilizar el protocolo HTTPS en los sitios web de nuestro servidor. La validez de dichos certificados solo engloba al dominio gestionado por el equipo "SERVIDOR".

Para comenzar la instalación vas a **Inicio-> Herramientas administrativas-> Administrador de Internet Information Services** y pinchas en el icono **Certificados de Servidor**.



José A. Jiménez (CC0)

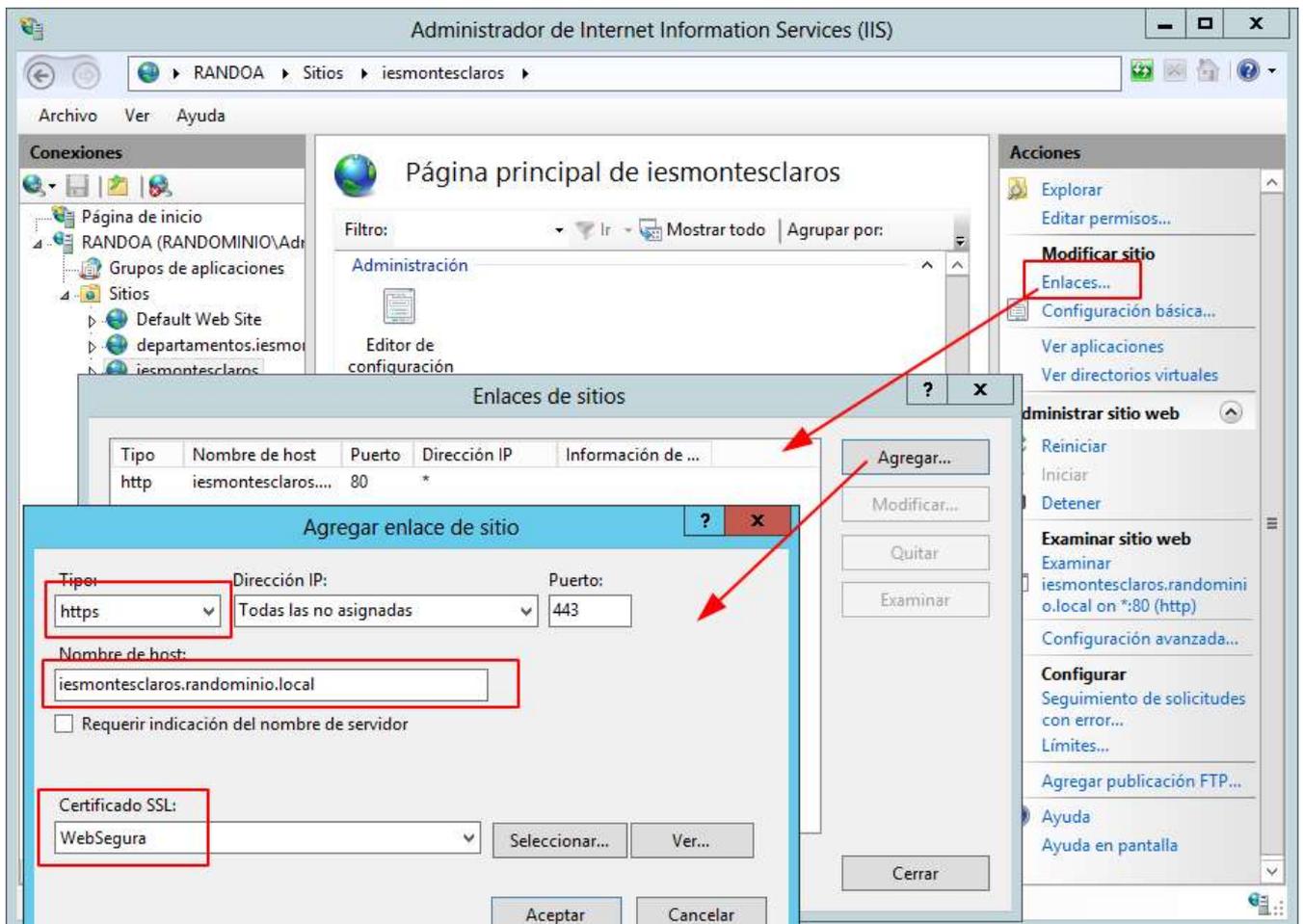
El siguiente paso será crear un certificado autofirmado. Para ello, pinchas en el enlace "Crear certificado autofirmado". Ponemos un nombre al certificado, por ejemplo, WebSegura.



José A. Jiménez (CC0)

Ahora podemos dirigirnos al sitio web que queremos que sea accesible a través de HTTPS. En nuestro caso, seleccionamos el sitio web principal de la web "iesmontesclaros" En la parte derecha de la ventana, tienes una opción llamada "Enlaces..."

Pulsamos "Agregar" y seleccionamos el protocolo HTTPS, la IP de la página, y el puerto 443. Y así ya tienes el certificado creado.



José A. Jiménez (CC0)

Ahora podríamos acceder a través de la nueva URL: <https://iesmontesclaros.randominio.local>

## Autoevaluación

Señala la afirmación correcta sobre el servidor IIS.

- No permite la creación de sitios seguros.
- Se puede utilizar el protocolo HTTPS para llegar a páginas seguras.
- Si se utiliza una página segura, ya no es posible utilizar HTTP para una página con acceso normal.
- Para utilizar el protocolo HTTPS no es necesario utilizar un certificado.

No es correcto. Repasa el capítulo.

Muy bien. Esta es una de las características del servidor IIS.

Incorrecta, puedes tener los dos sitios funcionando a la vez.

No es correcto, puesto que sí es necesario utilizar un certificado.

# Solución

1. Incorrecto
2. Opción correcta
3. Incorrecto
4. Incorrecto

## 4.- Servidor web Apache.

### Caso práctico



[Stockbyte](#). (Uso educativo no)

Como recordarás, habíamos dejado a Alberto, el responsable de mantenimiento del centro IES Montes Claros, reunido con María, la técnica superior en Administración de Sistemas Informáticos en Red. Estaban barajando distintas posibilidades sobre el servidor Web que fuesen las más convenientes para el centro.

-Me he dado cuenta, María, de lo potente que puede llegar a ser un sistema operativo de red- valoró Alberto.-Si, desde luego. Ofrece muchos servicios que se pueden aprovechar para prácticamente todos los ámbitos de trabajo –respondió María.-Sí, estoy de acuerdo. El único problema que nos podemos encontrar es si el centro podrá adquirir la licencia de este sistema operativo –añadió Alberto.-No hay problema, Alberto. Tenemos más opciones –le tranquilizó María- para instalar servidores Web. Existen servidores gratuitos muy potentes que se podrían instalar. Así te ahorras el tener que comprar la licencia.-¿También existen este tipo de productos de software libre? Me dejabas asombrado –dijo Alberto.-Podemos hacer pruebas con un servidor web gratuito sobre una distribución de Linux, que también será gratuita. Luego, con las dos versiones del mismo servicio, podrás hablar con Julián, el director, y darle los argumentos que consideres oportunos –dijo convincentemente María.-Me encantaría aprender. ¿Me ayudarías tú, por favor?-Faltaría más, solo necesitamos...-formuló María.

Como se desprende de la conversación entre Alberto y María, es importante conocer distintas versiones de un mismo producto. Eso te permitirá, en el futuro, elegir lo más adecuado en cada momento y circunstancia. Es posible que, cuando estés ejerciendo de técnico o técnica de Sistemas Microinformáticos y Redes, te pidan tu opinión sobre qué es lo más adecuado utilizar en un determinado contexto. Aquí entraría en juego los conocimientos que tú tengas del problema y la capacidad para valorar las distintas soluciones.

En los siguientes apartados correspondientes a este capítulo, aprenderás a instalar y mantener un servidor Web Apache sobre una distribución de Linux, como puede ser Ubuntu. Como introducción, cabe decir que el servidor Web Apache es una aplicación muy utilizada en el ámbito de Linux. Es el servidor más implantado entre los distintos servidores que ofertan servicios en Internet.

Se pueden destacar de este programa las siguientes características:

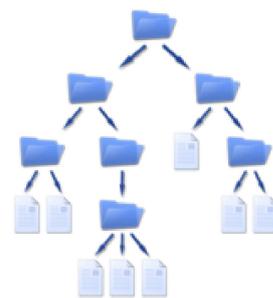
- ✓ Es modular.
- ✓ Permite crear servidores virtuales.
- ✓ Permite crear servidores seguros HTTPS.
- ✓ Permite crear sitios privados.
- ✓ Permite crear sitios de usuario.

Una vez que conoces las características genéricas de este servidor Web, te pondrás manos a la obra. Procedemos a la organización del sitio Web, la instalación del servidor Web y su posterior configuración.

## 4.1.- Organización del sitio web.

Debes aprender cómo es la clásica organización en los sistemas Linux, puesto que será la que utilizaremos para gestionar nuestro servidor Apache. En el siguiente listado verás donde se almacenan cada una de las páginas que van a depender de nuestro sitio Web.

1. **La página web de la intranet:** se almacena en la carpeta raíz del servidor Web.
2. **Las páginas de los usuarios:** se almacenan en la carpeta "home" de cada usuario.
3. **Las páginas de distintos departamentos** (o cualquier otra clasificación): lo más práctico es crear nuevos usuarios con el nombre del departamento y almacenar la página en la carpeta "home" del nuevo usuario.



FrancoGG. (CC BY-SA)

Has de saber que, una vez que esté instalado el servidor Web Apache, la carpeta raíz por defecto del servidor es `/var/www`. Todos los documentos que se encuentren dentro de la carpeta raíz del servidor web serán accesibles vía web. Para acceder a la página almacenada en la carpeta raíz del servidor desde un navegador, se accede directamente con la dirección IP del servidor o utilizando el nombre del servidor si tenemos el DNS funcionando.

A modo de resumen, se puede afirmar sobre un servidor Web Apache que:

- ✓ La carpeta raíz del servidor web es: `/var/www`
- ✓ El acceso a la web principal es: `http://ip-del.servidor` o `http://nombre-del-servidor`

Seguimos con la organización de los ficheros y carpetas del sitio Web. Cada usuario del sistema dispone en la carpeta "home" de una carpeta llamada "**public\_html**". Si no existe, se puede crear con los permisos 644 para el grupo y el resto de usuarios. En esta carpeta, el usuario tiene derechos para crear y copiar dentro de ella su página web.

El acceso a dicha página será: `http://ip-del-servidor/~nombre-usuario/`.

El carácter `~` se obtiene pulsando `Alt Gr + 4`. Así se indica a Apache que la página se encuentra en el directorio "home" del usuario.

Ejemplo: Tenemos creado el usuario "ana" y ha creado la carpeta `/home/ana/public_html`. Si copia su página Web aquí, desde cualquier PC de la red podrá acceder a la dirección `http://ip-del-servidor/~ana/`. Para que la página aparezca automáticamente, es necesario crear un archivo llamado **index.html**.

Si queremos proporcionar espacio web para otras clasificaciones, por ejemplo, departamentos del centro, lo más sencillo es crear un usuario por departamento. Se procederá de igual modo que como hemos visto para los usuarios personales creados para los profesores y profesoras.

### Autoevaluación

Señala la afirmación que es correcta sobre la organización del sitio web con Apache.

- La carpeta por defecto para el servidor web Apache es `/etc/www`.
- Para acceder a la página principal, solo se puede realizar a través del nombre del servidor.
- Las carpetas de los usuarios están dentro de la carpeta por defecto del servidor web Apache.
- Las carpetas de los usuarios se encuentran dentro de la carpeta "home", en la carpeta "public\_html".

No es esta la opción que buscas.

Esta opción no es correcta.

No es correcto, busca la verdadera.

Muy bien. Esta es la respuesta correcta.

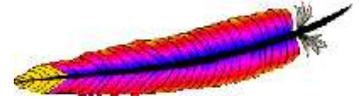
## Solución

1. Incorrecto
2. Incorrecto
3. Incorrecto
4. Opción correcta

## 4.2.- Instalación y configuración del servidor web Apache en Linux.

---

Para la instalación del servidor Web Apache necesitas tener una distribución Linux, que actuará como servidor de una red. (Puede ser un sistema operativo virtualizado o real). Para poder realizar la instalación del servidor Web Apache, necesitas tener conexión a Internet.



[PotatoSalad](#), (GNU/GPL)

Si cumples estos requisitos, para realizar la instalación, escribes en un terminal de Linux la siguiente instrucción (necesitas que tu máquina esté conectada a Internet):

```
$ sudo apt-get install apache2
```

Se necesita la cláusula **sudo** porque para realizar la instalación debe pedimos la contraseña del administrador. Te hará alguna pregunta sobre el consentimiento de la instalación de ciertos paquetes, y tú habrás de contestar que sí. Pasado un tiempo, el servidor quedará instalado.

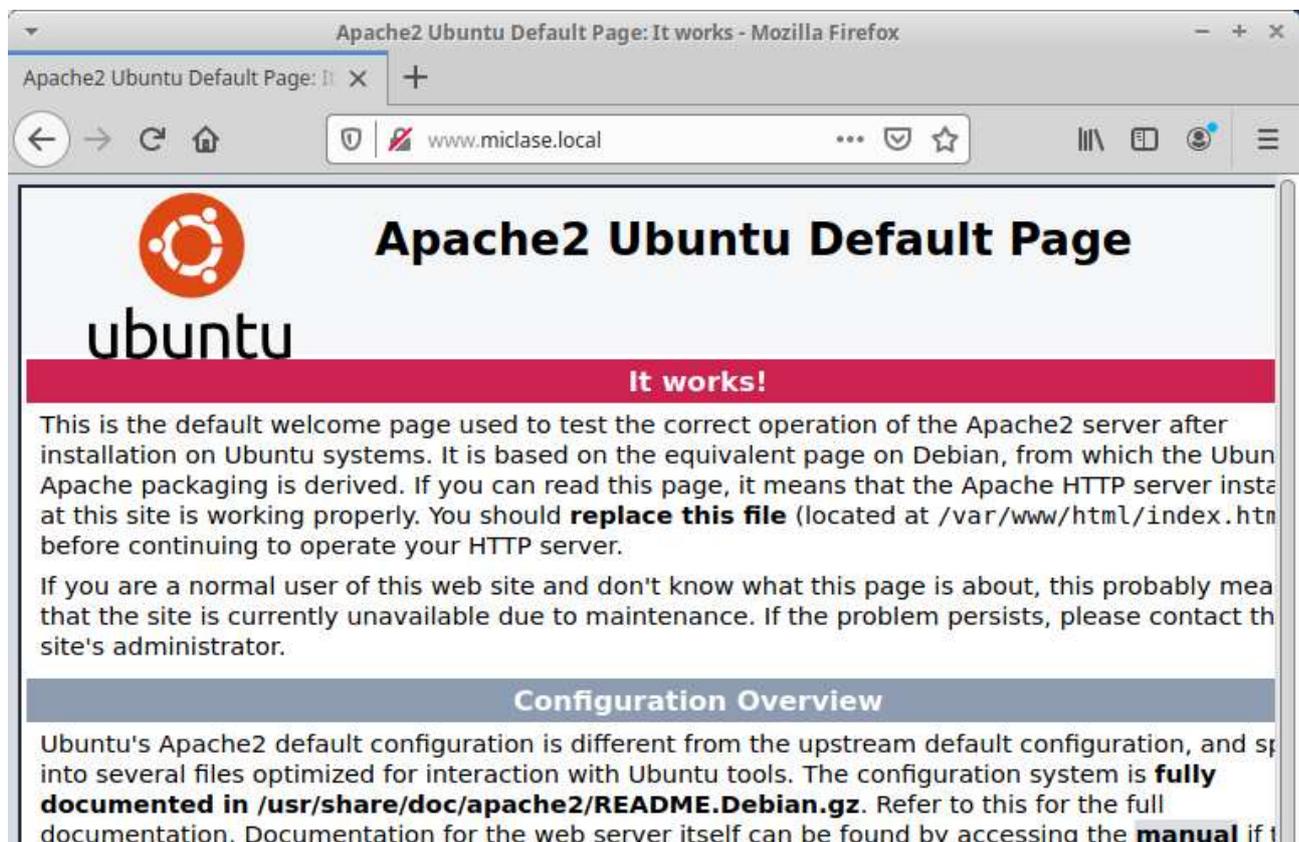
Pasamos ahora a la configuración del servidor. Los archivos de configuración se encuentran en la carpeta `/etc/apache2`. El archivo principal de configuración es `/etc/apache2/apache2.conf`. Es muy recomendable que antes de modificar nada en ese archivo, realices una copia de seguridad del mismo. Si se encuentra algún error en este fichero, el servidor Web no arrancará.

Se pueden configurar muchos parámetros. Nosotros editamos el archivo `.....apache2.conf` y añadimos el siguiente parámetro: **ServerName miclase.local**

Para que los ordenadores de la red local sepan que `miclase.local` es nuestro servidor web, se debe crear una entrada "www" hacia su dirección IP en el servidor DNS. Si añadimos la entrada al servidor DNS, cualquier equipo de la red local podrá acceder a la dirección completa, y no sería necesario recordar su dirección IP.

Otro modo sería editando el archivo `/etc/hosts` y agregando la línea "**192.168.100.254 www.myclase.local**" (suponiendo la IP del servidor 192.168.100.254). Esto solo nos resuelve el problema de que entienda la dirección completa en el propio servidor, aunque desde el resto de los clientes de la red tendrías que recordar la dirección IP del servidor web.

Si al realizar cualquiera de estas pruebas en el navegador, obtienes como respuesta una pantalla como la siguiente, significa que la instalación ha sido correcta y que el servidor Web funciona bien.



José A. Jiménez (CC0)

Recuerda que cuando hagas un cambio en la configuración del servidor web Apache, es necesario que **reinicies** el servidor Web. Debes conocer que, al igual que todos los servicios en Debian, dispones de un script de arranque y parada en la carpeta /etc/init.d/. Básicamente, son unos pequeños programas que automatizan la tarea de reiniciar el servidor o de pararlo.

- ✓ Para arrancar o reiniciar el servidor: `$ sudo systemctl restart apache2` (o `start`)
- ✓ Para parar el servidor: `$ sudo systemctl stop apache2`

## Para saber más

Existen muchas fuentes de información y tutoriales fáciles de localizar en internet. Por ejemplo, el siguiente tutorial explica cómo instalar un servidor Apache basado en Ubuntu 18.04

<https://www.digitalocean.com/community/tutorials/como-instalar-el-servidor-web-apache-en-ubuntu-18-04-es>

## 4.3.- Carpetas de usuario en Apache.

---

Una vez que has instalado y configurado adecuadamente el servidor Web Apache, puedes crear una página index.html cualquiera y cambiarla por el index.html que se encuentra en “/var/www”. Este es el lugar donde está ubicado el sitio web general.

Además de esta página, ya se comentó que cada usuario puede tener su espacio web. Este espacio se almacena en la carpeta **public\_html** dentro de su carpeta personal. Puede suceder que esta carpeta no exista, con lo que el propio usuario la puede crear para almacenar en ella su sitio web. Para poder ver el contenido de esta carpeta, tienes que habilitar la funcionalidad del Apache **UserDir**.

En versiones anteriores de Apache, habilitar la funcionalidad userdir consistía en agregar una línea al archivo de configuración del servicio apache. Pero en versiones posteriores, la parte de UserDir se convirtió en un módulo más, por lo que se hace necesario cargar ese módulo cuando el servidor Apache se inicia, para tener disponibles los directorios de usuario.

En el directorio /etc/apache2/ encontrarás, entre otras cosas, los siguientes ficheros y directorios:

- ✓ **Apache2.conf**: archivo de configuración de apache.
- ✓ **Mods-enabled**: contiene los módulos que se activan cuando se inicia Apache.
- ✓ **Mods- available**: contiene los módulos que se instalaron en el momento de instalar Apache.

Abrimos una ventana de Terminal y entramos en la carpeta **mods-available** con la siguiente instrucción:

```
$ cd /etc/apache2/mods-available
```

Dentro de la carpeta ejecutamos el comando: **ls userdir\***. Nos debe mostrar dos archivos: userdir.conf y userdir.mod. Estos módulos están disponibles pero no activados. Son los que necesitamos para poder utilizar las carpetas personales de los usuarios. Para activarlos tenemos dos opciones:

1) Con el comando "a2enmod"

```
$ sudo a2enmod userdir
```

2) Manualmente: debemos entrar en el directorio mods-enabled y crear enlaces simbólicos hacia estos dos archivos.

Para meternos en el directorio mods-enabled escribimos el comando “**cd ../mods-enabled**”. Una vez dentro del directorio, ejecutamos el siguiente comando:

```
$ sudo ln -s /etc/apache2/mods-available/userdir* . (Ojo, no olvides poner el punto final).
```

El comando anterior crea un enlace simbólico de los archivos userdir que se encuentran en mods-available. El **punto** final significa que esos enlaces los tiene que **crear en la carpeta actual**, que se supone es mods-enabled.

El siguiente paso será **reiniciar** el servidor apache con la siguiente instrucción:

```
$ sudo systemctl restart apache2
```

Si todo sale bien, veremos el servicio en estado “enabled”:

```
$ sudo systemctl status apache2
```

```
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
   Active: active (running) since Sat 2019-12-28 18:53:51 CET; 45s ago
   Docs: https://httpd.apache.org/docs/2.4/
   Process: 5710 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCE
   Main PID: 5727 (apache2)
   Tasks: 55 (limit: 2331)
   Memory: 4.9M
   CGroup: /system.slice/apache2.service
           └─5727 /usr/sbin/apache2 -k start
             └─5728 /usr/sbin/apache2 -k start
               └─5729 /usr/sbin/apache2 -k start
```

Ahora podremos ir a la carpeta "home" de algún usuario, por ejemplo /home/profesor, donde crearemos una carpeta "public\_html" que albergará los archivos html del usuario.

```
profesor@servidordns:~$ pwd
/home/profesor
```

```
profesor@servidordns:~$ mkdir public_html
```

Creamos un archivo "index.html" de la siguiente forma:

```
profesor@servidordns:~$ pwd
/home/profesor
```

```
profesor@servidordns:~$ cd public_html
```

```
profesor@servidordns:~/public_html$ pwd
/home/profesor/public_html
```

```
profesor@servidordns:~/public_html$ vi index.html
```

(aquí editaríamos el contenido del archivo. El comando "vi" es el editor de texto clásico de Linux, pero también puedes usar otros editores, como emacs o nano)

```
profesor@servidordns:~/public_html$ cat index.html
<!doctype html>
<html>
<head>
<title>Web Profesor</title>
</head>
<body>
<h1>Web del profesor</h1>
<p>Contenido de la web</p>
</body>
</html>
```

Y ahora podremos acceder a él desde un navegador con la URL

<http://ip-del-servidor/~nombre-usuario>

En este caso (suponiendo que ya tenemos configurado DNS con el dominio miclase.local:

<http://www.miclase.local/~profesor/>



## Web del profesor

Contenido de la web

José A. Jiménez (CC0)

## Autoevaluación

¿Cómo se llama el archivo de configuración de Apache?

- Apache2.conf.
- Es el fichero Hosts.
- Son los ficheros que se encuentran en el directorio /mods-enabled.
- Ninguna de las otras.

Muy bien. Esa es la respuesta correcta.

No es esta la opción que buscas, puesto que es incorrecta.

No es ésta la opción correcta.

No es correcto, busca la opción verdadera.

## Solución

1. Opción correcta
2. Incorrecto
3. Incorrecto
4. Incorrecto

## 4.4.- Uso de archivos .htaccess para limitar acceso a usuarios

---

Los archivos .htaccess permiten modificar la configuración de un servicio instalado en un directorio. Suelen usarse en carpetas de usuarios que no tienen permisos totales, permitiendo que ejerzan cierto control sobre su parcela de contenidos del servicio web.

Para ello, se debe crear un archivo denominado ".htaccess" en el directorio que debe verse afectado. Las modificaciones afectarán a todos los subdirectorios y no requieren reiniciar el servidor.

Para que los archivos .htaccess sean tenidos en cuenta, se requiere la directiva "AllowOverride".

El contenido del archivo .htaccess (recordemos que afecta a la carpeta en la que se ubica y a sus subdirectorios) tiene el mismo efecto que si se añaden las líneas a la sección <Directory> del archivo /etc/apache2/apache2.conf.

Algunos de los usos más comunes de los archivos .htaccess son:

- Crear direcciones amigables
- Crear redirecciones
- Restringir el acceso a directorios del servidor
- Bloquear el acceso desde ciertas direcciones IP
- Manejar errores del servidor
- Evitar el hotlink o enlace a imágenes
- Cambiar un dominio con o sin www

Para ver la utilidad de un archivo .htaccess, crearemos una carpeta web de acceso privado, /var/www/html/privado, protegido por contraseña.

Seguiremos estos pasos:

1. Creamos el directorio privado y un archivo HTML contenido en ella

```
$ sudo mkdir /var/www/html/privado
$ cd /var/www/html/privado
$ cat index.html
<!doctype html>
<html>
<head>
<title>Web Privada</title>
</head>
<body>
<h1>Web privada</h1>
<p>Contenido de la web</p>
</body>
</html>
```

2. Habilitamos la directiva **AllowOverride** para la carpeta en la que instalaremos el servicio:

```
$ cat /etc/apache2/apache2.conf
(...)
<Directory /var/www/html/privado>
Options Indexes FollowSymLinks
AllowOverride FileInfo AuthConfig Limit Indexes
Require all granted
</Directory>
(...)
```

3. Creamos un archivo .htaccess en la carpeta afectada, /var/www/html/privado:

```
profesor@servidordns:/var/www/html/privado$ sudo vi .htaccess
profesor@servidordns:/var/www/html/privado$ cat .htaccess
AuthType Basic
```

```
AuthName "Este directorio está protegido"  
AuthUserFile .htpasswd  
Require valid-user
```

4. Creamos la contraseña encriptada para el usuario "profesor":

```
profesor@servidordns:/var/www/html/privado$ htpasswd -n profesor  
New password:  
Re-type new password:  
profesor:$apr1$aFQLRZk5$o2K/e3dILetlRL11FTmWX/
```

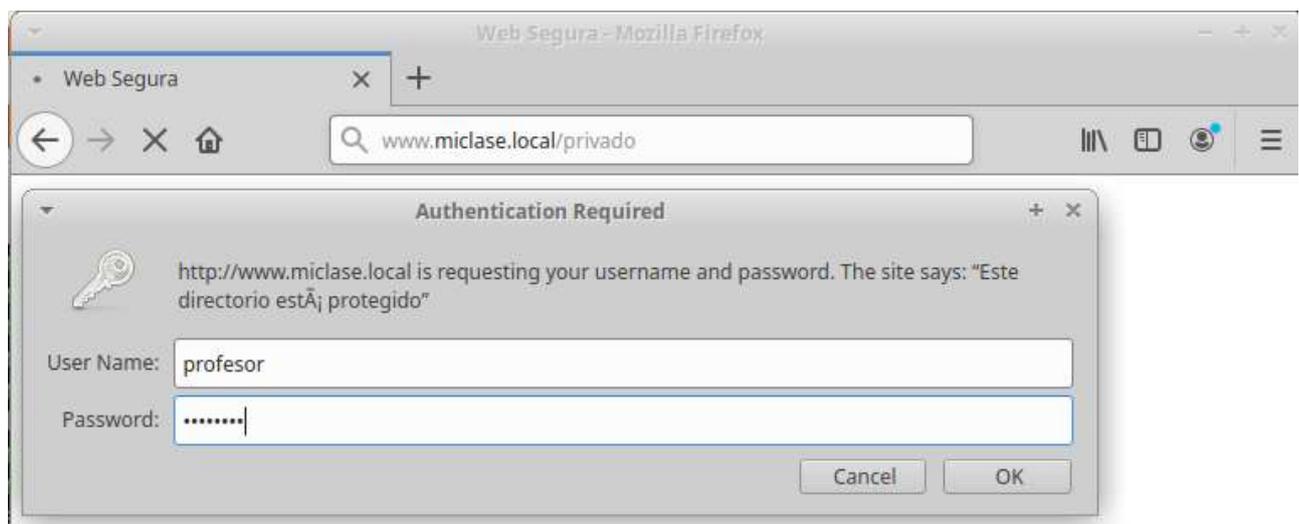
5. Generamos el archivo de contraseña, con la obtenida en el paso anterior, en /etc/apache2:

```
profesor@servidordns:/etc/apache2$ sudo vi .htpasswd  
profesor@servidordns:/etc/apache2$ cat .htpasswd  
profesor:$apr1$aFQLRZk5$o2K/e3dILetlRL11FTmWX/
```

6. Reiniciamos el servicio web:

```
$ sudo systemctl restart apache2
```

7. Al acceder a la URL afectada (<http://www.miclase.local/privado>) nos encontraremos con una petición de credenciales:



José A. Jiménez (CC0)

Y si ahora introducimos la contraseña del usuario autorizado (en este caso, "profesor"), accederemos a la web privada.



# Web privada

Contenido de la web

José A. Jiménez ([CC0](#))

## 4.5.- Uso de archivos .htaccess para redireccionamiento

---

Las redirecciones son muy importantes en el mundo del marketing digital, porque los servicios de Google penalizan las páginas con contenido duplicado o enlaces rotos. Si una empresa migra su portal web a una nueva dirección, las URLs de acceso antiguas deben redirigirse a las nuevas ubicaciones.

Para que funcionen las directivas de redirección hay que activar el módulo "rewrite" en el panel de configuración principal de Webmin o desde un terminal, escribiendo:

```
$ a2enmod rewrite
$ sudo systemctl restart apache2
```

Para probar la redirección creamos la carpeta /var/www/html/redirecciones.

En primer lugar añadimos la directiva "AllowOverride" en el archivo de configuración /etc/apache2/apache2.conf

```
(...)
<Directory /var/www/html/redirecciones>
Options Indexes FollowSymLinks
AllowOverride FileInfo AuthConfig Limit Indexes
Require all granted
</Directory>
(...)
```

### Redirección de directorios dentro de un servidor

En primer lugar, partimos del directorio del servicio /var/www/html/redirecciones :

```
profesor@servidordns:/var/www/html/redirecciones$ sudo mkdir nuevaweb
profesor@servidordns:/var/www/html/redirecciones$ sudo mkdir viejaweb
profesor@servidordns:/var/www/html/redirecciones$ ls -al
(...)
drwxr-xr-x 2 root root 4096 dic 28 22:03 nuevaweb
drwxr-xr-x 2 root root 4096 dic 28 22:03 viejaweb
```

Los archivos html en las carpetas serán estos:

- en **nuevaweb/index.html**:

```
<!doctype html>
<html>RewriteEngine On
Redirect 301 /redirecciones/viejaweb/ /redirecciones/nuevaweb
<head>
<meta charset="utf-8">
<title>Nueva Web</title>
</head>
<body>
<p><strong>Bienvenido a la nueva web</strong></p>
<p> Esto es un texto de prueba de la nueva web </p>
</body>
</html>
```

- en **viejaweb/index.html**:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Vieja Web</title>
</head>
<body>
<p><strong>Bienvenido a la vieja web</strong></p>
<p> Esto es un texto de prueba de la vieja web </p>
```

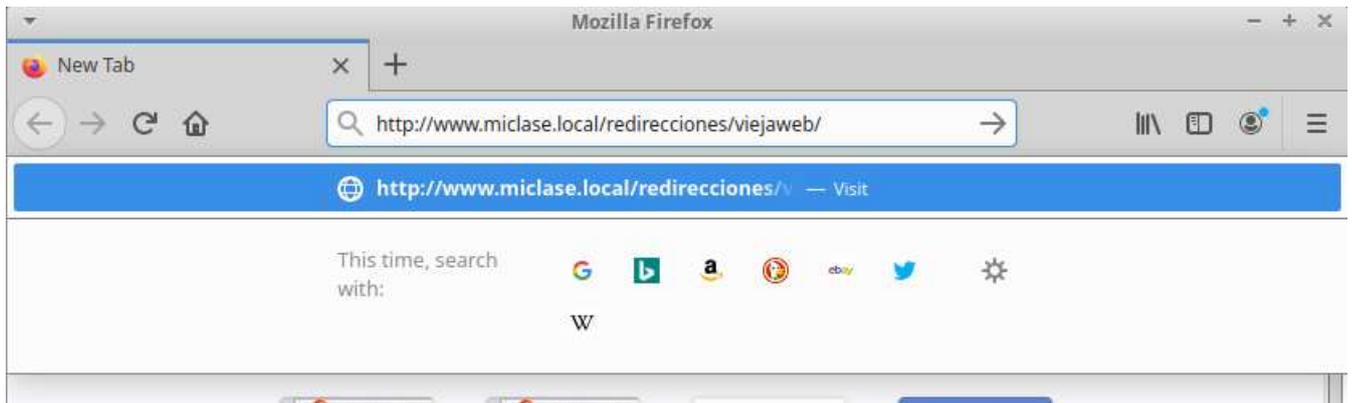
```
</body>
</html>
```

Para conseguir la redirección automática de las carpetas, crearemos un archivo `.htaccess` en la carpeta principal del servicio, `/var/www/html/redirecciones`:

```
$ cat .htaccess
RewriteEngine On
Redirect 301 /redirecciones/viejaweb/ /redirecciones/nuevaweb
```

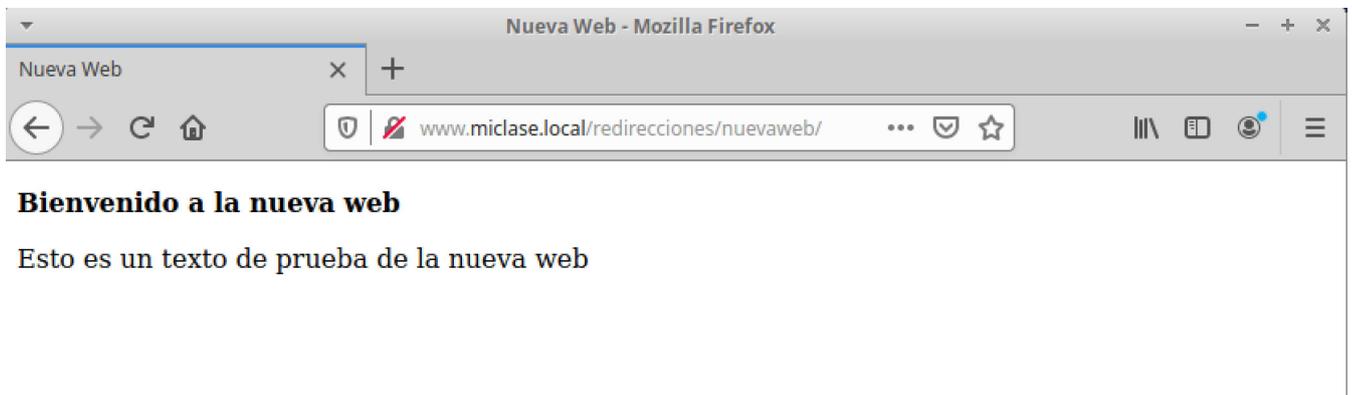
Y tras reiniciar el servicio, cada vez que intentemos acceder con el navegador a la carpeta “viejaweb”, nos llevará a “nuevaweb”. Abrimos el navegador, tecleamos la URL:

<http://www.miclase.local/redirecciones/viejaweb/>



José A. Jiménez (CC0)

Al pulsar ENTER, el navegador carga la página de destino (redirigida):



José A. Jiménez (CC0)

## 4.6. Creación de Virtual Hosts

---

En un solo equipo podemos tener varios sitios o portales web atendiendo a diferentes nombres de dominio o diferentes puertos. Veamos un sencillo ejemplo de creación de un "virtualhost" para que nuestro servidor web Apache responda por el puerto 81, devolviendo una página web diferente a la que devolvería si se accede por el puerto 80 (el puerto por defecto).

El procedimiento consiste en habilitar el puerto 81 y definir el nuevo host virtual, al que llamaremos "webport81", y que será accesible desde un navegador en la url "www.miclase.local:81". Seguiremos los siguientes pasos:

### 1. Habilitar el puerto 81

```
profesor@servidordns:/etc/apache2$ ls
apache2.conf  envvars  mods-enabled  sites-enabled
conf-available  magic  ports.conf
conf-enabled  mods-available  sites-available
```

```
profesor@servidordns:/etc/apache2$ sudo vi ports.conf
```

```
profesor@servidordns:/etc/apache2$ more ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf
```

```
Listen 80
Listen 81
(...)
```

### 2. Crear el host virtual (crear un archivo en /etc/apache2/sites-available, denominado "webport81.conf", indicando el nuevo puerto y la carpeta donde se alojará este servicio):

```
profesor@servidordns:/etc/apache2/sites-available$ more webport81.conf
<VirtualHost *:81>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
ServerName www.miclase.local

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html2
```

### 3. Crear el contenido web en la carpeta (creamos la carpeta /var/www/html81 y escribimos un archivo "index.html").

```
profesor@servidordns:/var/www$ sudo mkdir html2
profesor@servidordns:/var/www$ cd html2/
```

```
profesor@servidordns:/var/www/html2$ sudo vi index.html
profesor@servidordns:/var/www/html2$ cat index.html
```

```
<!doctype html>
<html>
<head>
<title>Webport 81</title>
</head>
```

```
<body>
<h1>www.miclase.local:81</h1>
<p>Esta es la web del virtualhost por el puerto 81</p>
</body>
</html>
```

#### 4. Habilitar el host virtual

```
profesor@servidordns:/etc/apache2$ ls sites-available
000-default.conf default-ssl.conf webport81.conf
profesor@servidordns:/etc/apache2$ ls sites-enabled
000-default.conf
```

```
profesor@servidordns:/etc/apache2/sites-enabled$ sudo a2ensite webport81.conf
```

```
profesor@servidordns:/etc/apache2$ ls sites-enabled
000-default.conf webport81.conf
```

#### 5. Recargar / reiniciar el servicio

```
profesor@servidordns:/etc/apache2/sites-enabled$ sudo systemctl reload apache2
```

Y desde un navegador abrimos la url [www.miclase.local:81](http://www.miclase.local:81). Veremos este resultado:



José A. Jiménez (CC0)

## Debes conocer

Una de las tareas más importantes del Administrador de Sistemas es el diagnóstico de incidencias basándose en los registros de errores. Los registros del servicio Apache se encuentran en `/var/log/apache2` y son los siguientes:

```
profesor@servidordns:/var/log/apache2$ ls -al
total 48
drwxr-x--- 2 root adm 4096 dic 28 18:42 .
drwxrwxr-x 14 root syslog 4096 dic 28 21:43 ..
-rw-r----- 1 root adm 15046 dic 28 23:24 access.log
-rw-r----- 1 root adm 16599 dic 28 23:24 error.log
```

## 4.7.- Acceso a carpetas seguras.

Ya sabes, porque lo has visto al instalar el servidor Windows Server 2012, que una página web segura es un sitio que utiliza el protocolo HTTPS en lugar de utilizar el protocolo HTTP.

Estos protocolos son idénticos, con la excepción de que la transferencia de información entre el cliente (navegador web) y el servidor (servidor web) viaja a través de Internet de manera codificada, utilizando robustos algoritmos de cifrado de datos proporcionados por el paquete OpenSSL.

Ya sabes que, la carpeta que usa Apache por defecto para alojar sus páginas es “var/www”. Bien, pues dentro de esta carpeta, vamos a crear una llamada websegura, donde alojaremos el servicio securizado. Para poder activarla seguiremos estos pasos:

- Habilitar el módulo SSL:

```
sudo a2enmod ssl
```

- Habilitar el módulo SSL por defecto:

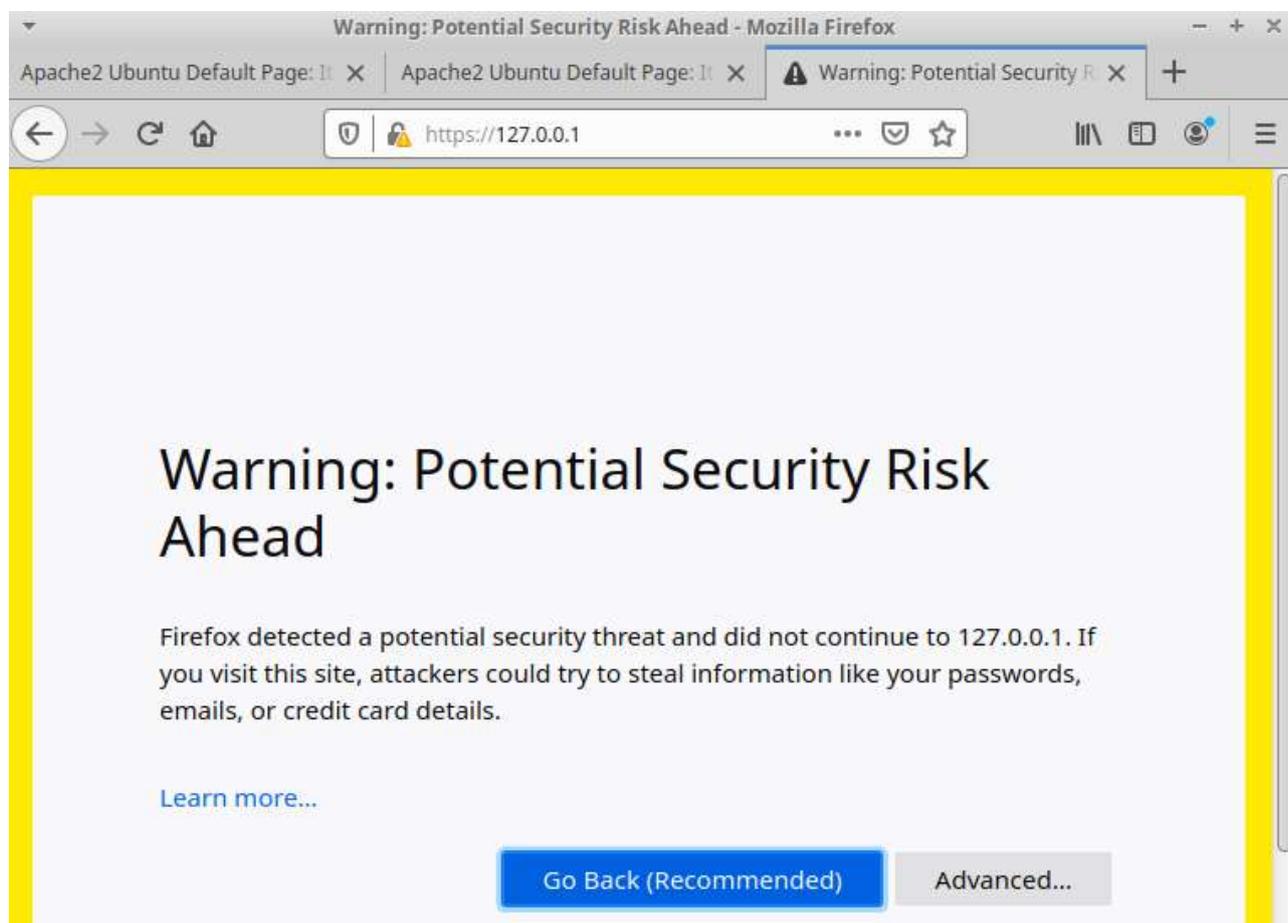
```
sudo a2ensite default-ssl
```

- Reiniciar el servidor:

```
sudo /etc/init.d/apache2 restart
```

Con estos pasos, ya queda habilitado el acceso al sitio web de manera segura. Se puede probar poniendo en el navegador la dirección <https://www.miclase.local>

Nos aparecerá una advertencia de seguridad porque el certificado no es oficial, pero pulsamos en “Advanced” y lo aceptamos:



Ahora, vamos a crear nuestro propio certificado. Hay que seguir una serie de pasos:

- Vamos al directorio apache2:

```
$ cd /etc/apache2
```

- Generas la llave:

```
$ sudo openssl genrsa -des3 -out server.key 1024
```

- Creamos el certificado con la llave:

```
$ sudo openssl req -new -key server.key -out server.csr
```

- Creamos la clave, contestando una serie de preguntas y recordando la contraseña:

```
$ sudo openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

- Cambias de carpeta los archivos:

```
$ sudo cp server.crt /etc/ssl/certs/  
$ sudo cp server.key /etc/ssl/private/
```

- Abrimos el archivo de configuración default-ssl:

```
$ cd /etc/apache2/sites-available
```

- Lo editamos:

```
$ sudo vi default-ssl
```

- Una vez abierto el archivo, reemplazamos y habilitamos la siguiente configuración, modificando las líneas que corresponda:

```
<VirtualHost websegura.miclase.local:443>  
DocumentRoot /var/www/websegura  
<Directory /var/www/websegura>  
SSLOptions +FakeBasicAuth +ExportCertData  
SSLCertificateFile /etc/ssl/certs/server.crt  
SSLCertificateKeyFile /etc/ssl/private/server.key
```

- Habilitamos el archivo :

```
$ sudo a2ensite default-ssl
```

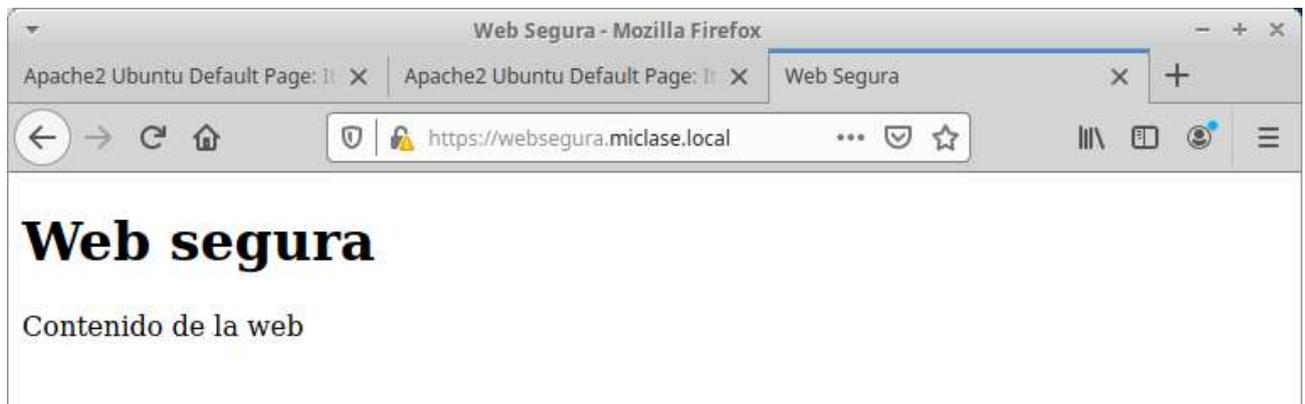
- Creamos la carpeta /var/www/websegura y añadimos un archivo index.html
- Damos de alta la dirección websegura.miclase.local en el DNS, añadiendo esta línea en el archivo /etc/bind/db.miclase.local:

```
websegura IN CNAME servidordns
```

- Reiniciamos apache y el servicio quedará activo:

```
$ sudo /etc/init.d/apache2 restart
```

Ahora solo te queda probar el acceso a <https://websegura.miclase.local>



José A. Jiménez ([CC0](#))

## Para saber más

En el siguiente enlace se propone que leas un artículo sobre el servidor web Apache. Resulta interesante por la enumeración que ofrece de sus ventajas o de los módulos que se pueden instalar, cosa que tú ya has hecho, por ejemplo, con el módulo ssl.

[Servidor web Apache.](#)