

UT4: El servicio web

Servicios en Red- 2º Curso CFGM SMR

Índice.

- ① Introducción.
- ② Funcionamiento del servicio http.
- ③ Acceso seguro y utilización de certificados.
- ④ Parámetros de configuración del servicio http.
 - ① Servidor.
 - ② Cliente.
- ⑤ Configuración del servidor http.
 - ① Windows.
 - ② Linux.
- ⑥ Configuración del cliente http: navegadores.
 - ① Windows.
 - ② Linux.



1. Introducción.



HTTP HyperText Transfer Protocol

Protocolo de la capa de aplicación que facilita a los usuarios de forma sencilla e intuitiva el acceso a la información remota conectándose a una red TCP/IP.



1. Introducción.



WWW World Wide Web

El modelo cliente/servidor y el protocolo http son la base de la WWW o simplemente web. WWW es un servicio de distribución de información que permite acceder a millones de recursos electrónicos y aplicaciones distribuidos en servidores por todo internet y localizados por direcciones (URIs o URLs). Los recursos se conectan a través de hiperenlaces/hipervínculos/links lo que permite navegar de uno a otro fácilmente.

La www fue desarrollado por CERN en 1989 y actualmente su desarrollo está controlado por **W3C** (World Wide Web Consortium) una comunidad que desarrolla estándares web como XHTML, CSS y XML.



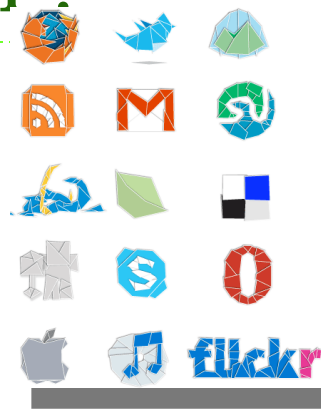
2. Funcionamiento del servicio HTTP.

- ▶ Componentes: el servicio que ofrece la Web se basa en el modelo cliente/servidor y está formado por los componentes:
 - ▶ Recursos.
 - ▶ Nombres y direcciones.
 - ▶ Clientes web (clientes HTTP o navegadores)
 - ▶ Servidores web (o servidores HTTP)
 - ▶ Proxies web (o proxies HTTP)
 - ▶ Protocolo HTTP.
 - ▶ Tecnologías web



2. Funcionamiento del servicio HTTP.

- ▶ **Recursos:** Documentos, vídeos imágenes, audio aplicaciones, buzones de correo, etc, ... accesibles a través de servidores web y conectados por hiperenlaces.
- ▶ **Nombres y direcciones** (URIs y URLs): sistema de nombres basado en cadenas de caracteres que identifican y localizan inequívocamente a los recursos en la Web.
- ▶ **Clientes web** (navegadores): permiten a los usuarios acceder a los recursos disponibles en servidores web.



2. Funcionamiento del servicio HTTP.

- ▶ **Servidores web:** atienden las peticiones de los clientes y les envían los recursos solicitados.
- ▶ **Proxies web:** programas intermediarios entre clientes y servidores web. Pueden actuar como cortafuegos y/o almacenar datos en cache para aumentar el rendimiento.
- ▶ **Protocolo HTTP:** conjunto de normas y reglas en base a las cuales “dialogan” los clientes, los servidores web y los proxies. Usa TCP como protocolo de transporte.



2. Funcionamiento del servicio HTTP.

► Recursos.

► Páginas web:

- **Documento hipermedia** o conjunto de información relacionada (texto, audio, imágenes, ...) que contiene links a otras web o recursos.

► Sitio web:

- **Conjunto de páginas web** relacionadas y accesibles a partir de un mismo nombre de dominio DNS. Todos los sitios web de internet constituyen la WWW.
- Normalmente ese conjunto está almacenado en un directorio específico del servidor web.
- En ese directorio se suele establecer una jerarquía de subdirectorios para organizar las distintas páginas web y el resto de elementos que lo componen.
- Index.html suele organizarlo y está en la raíz.



2. Funcionamiento del servicio HTTP.

► Nombres y direcciones (URLs)

► URL: Universal Resource Locator.

- Cadena de texto que se utiliza para identificar un recurso y además nos da información sobre como acceder a él, como localizarlo.

► Formato URL:

Parte de la URL	Descripción	Ejemplo
Servicio:	Indica el servicio o protocolo a usar: http, https, ftp, telnet, ...	http:
//	Separador	
Servidor	Indica la IP o el nombre del servidor	www.opensuse.org
Ruta	Indica el directorio o subdirectorios donde reside en recurso.	/es
Recurso	Recurso al que se quiere acceder.	/index.html



2. Funcionamiento del servicio HTTP.

- ▶ Servidores web o servidores http.
 - ▶ Son programas que **atienden peticiones HTTP**, procesan e interpretan código escrito en diferentes lenguajes y envían a los clientes los recursos solicitados.
 - ▶ Estos recursos pueden estar en el propio equipo o en otros.
 - ▶ Pueden enviar contenido estático (archivos en diferentes formatos) como dinámico (el resultado de ejecutar programas).
 - ▶ Por defecto, escuchan las peticiones HTTP en el puerto **TCP 80**.
 - ▶ Algunos servidores web:
 - ▶ Libres: Apache HTTP server, nginx (engine X), lighttpd (lighty).
 - ▶ Propietarios: IIS (Internet Information Server) de MS



2. Funcionamiento del servicio HTTP.

► Clientes web (navegadores).

- Programas con los que interactúa el usuario y que permiten, entre otras, introducir URLs para acceder a recursos de la red.
- Pueden actuar como clientes de diferentes protocolos pero su función principal es la de ejercer de clientes http.
- Mantienen una memoria caché para almacenar: historial, contraseñas, etc.
- Permiten opciones múltiples de configuración y personalización.
- Ejemplos: Internet explorer, Mozilla Firefox, Google Chrome, Safari, Opera, ...



2. Funcionamiento del servicio HTTP.

► Proxies web:

- En redes: Proxy = Intermediario.
- Proxy web = Intermediario entre servidor y cliente HTTP.
- Proxy directo (forward proxy):
 - Recibe la petición de un cliente web y la traslada al servidor.
 - La petición del cliente es hacia el servidor, no al proxy.
 - Usado para optimizar y controla accesos a Internet de los clientes de una empresa.
- Proxy inverso (reverse proxy):
 - Reciben la petición de un cliente y la reenvían al servidor.
 - La petición del cliente ahora es hacia el proxy (para los clientes es un servidor web).
 - Los clientes usan la URL del proxy.
 - Se usan para proporcionar acceso a servidores web que están detrás de cortafuegos y no son accesibles directamente.
 - Objetivo: balancear carga, aumentar la seguridad en los accesos, etc.



2. Funcionamiento del servicio HTTP.

► Protocolo HTTP.

- Define las reglas que utilizan los componentes software (clientes, servidores y proxies) para comunicarse.
- Determina los tipos de peticiones que los clientes pueden enviar, así como el formato y la estructura de las respuestas.
- También define una estructura de metadatos, en forma de cabeceras que se envían tanto en las peticiones como en las respuestas.
- Versiones:
 - `http/0.9` (obsoleta)
 - `http/1.0`
 - `http/1.1` (versión actual)
 - `http/1.2` (experimental)



2. Funcionamiento del servicio HTTP.

1. El usuario introduce una URL en la barra de direcciones del navegador (o clic sobre link).
2. El navegador descompone la URL en:
 1. Protocolo de acceso.
 2. El nombre DNS o dirección IP del servidor.
 3. El puerto, si es que viene especificado.
 4. El objeto requerido.
3. Si se ha especificado un nombre DNS, se buscará la IP asociada (UT4).
4. Establece una conexión TCP con el servidor Web (puerto 80 por defecto).
5. El navegador envía el mensaje **HTTP de petición**.
6. El servidor envía el mensaje **HTTP de respuesta** en función del estado del servidor y de la petición enviada.
7. Se cierra la conexión TCP.

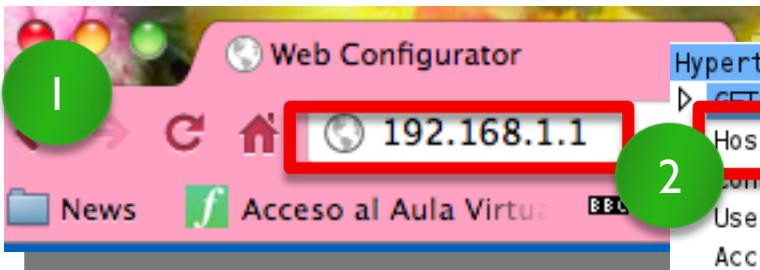


3. Acceso seguro y utilización de certificados.

▶ Autenticación:

- ▶ HTTP soporta el uso de mecanismos de autenticación para controlar el acceso que ofrece el servidor.
- ▶ Uso de las cabeceras `www-authenticate` y `authorization`.
- ▶ Los navegadores muestran al usuario un cuadro de diálogo para que se identifique.
- ▶ Los mecanismos de autenticación pueden ser:
 - ▶ Basic: envío de `user/password` codificados en `base64`. (Wireshark lo descifra 😊)
 - ▶ Digest: envío del `user` y una función hash (resumen) del `password`.
- ▶ Este tipo de autenticación no es segura y por ello se ha trasladado a las aplicaciones web.





Hypertext Transfer Protocol

GET / HTTP/1.1\r\n

Host: 192.168.1.1\r\n

Connection: keep-alive\r\n

User-Agent: Mozilla/5.0 (Mac

Accept: text/html,application

Accept-Encoding: gzip,deflat

Accept-Language: en-US,en;q=

Accept-Charset: ISO-8859-1,u

\r\n

3

HTTP/1.1 401 Unauthorized\r\n

WWW-Authenticate: Basic realm="P-660HW-D1"\r\n

Content-type: text/html\r\n

Transfer-Encoding: chunked\r\n

Server: RomPager/4.07 UPnP/1.0\r\n

Connection: close\r\n

EXT:\r\n

\r\n

HTTP chunked response

Line-based text data: text/html

4

The server 192.168.1.1:80 requires a username and password. P-660HW-D1.

User Name: 1234

Password:

Cancel

5

Hypertext Transfer Protocol

GET / HTTP/1.1\r\n

Host: 192.168.1.1\r\n

Connection: keep-alive\r\n

Authorization: Basic MTIzNDoxMjM0\r\n

Credentials: 1234:1234

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X

Accept: text/html,application/xhtml+xml,application

Accept-Encoding: gzip,deflate,sdch\r\n

Accept-Language: en-US,en;q=0.8\r\n

Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3\r\n

\r\n

[5/1] request URL: http://192.168.1.1/

6

[2 Reassembled TCP Segments (1001 bytes): #44(205), #46(

Hypertext Transfer Protocol

HTTP/1.1 200 OK\r\n

Content-Type: text/html\r\n

Date: Thu, 06 Jan 2000 16:52:37 GMT\r\n

Pragma: no-cache\r\n

Expires: Thu, 26 Oct 1995 00:00:00 GMT\r\n

Transfer-Encoding: chunked\r\n

Server: RomPager/4.07 UPnP/1.0\r\n

EXT:\r\n

\r\n

HTTP chunked response

3. Acceso seguro y utilización de certificados.

► Seguridad.

► HTTP no es un protocolo seguro

- El intercambio de información se realiza en texto plano.
- Los mecanismos de autenticación como Basic y Digest no son seguros.
- No hay mecanismos para garantizar que los equipos involucrados en la transferencia son quienes dicen ser. Susceptibilidad de ataques de suplantación de identidad (spoofing, man-in-the-middle).
- Existen ataques que se basan en el robo o falsificación de cookies y/o parámetros enviados en la URL o en el contenido de los mensajes, y que permiten al atacante “robar la identidad a un usuario suplantándolo en webs (bancos, webmails, redes sociales).



4. Parámetros de configuración del servicio http.

▶ Parámetros del servidor.

- ▶ **Configuración de red** correcta, **puerto 80** habilitado y nombre del sitio añadido en el servidor **DNS** correspondiente.
 - ▶ Crear un **directorio** donde se ubicará la **raíz** del sitio http.
 - ▶ Se puede establecer opcionalmente una **jerarquía de páginas** y subdirectorios a efectos de organización.
 - ▶ **Webmaster:** usuario encargado de gestionar el sitio.
 - ▶ Se deberán **alojar las páginas** en los directorios correspondientes antes de iniciar el servicio.
 - ▶ Asegurarse de que los enlaces interconectan las páginas adecuadamente: creación de un índice.
 - ▶ A continuación ya se podrán configurar parámetros adicionales: soporte HTTPS (generación de certificados), requerir autenticación, alojamiento virtual, etc.
-

4. Parámetros de configuración del servicio http.

► Parámetros del cliente.

- Una vez instalado, se pueden instalar elementos adicionales según las necesidades del usuario (plug-in/add-on/complemento/extensión).
- **Plug-in:** software adicional que permitirá al navegador ejecutar muchas de las aplicaciones disponibles en Internet (diferentes formatos de audio, video, etc).
- Configuración del **aspecto** del navegador, **web de inicio**, **historial** de navegación, gestión de **favoritos**, etc.
- Configuración de la **seguridad**: instalación de **certificados** al conectarnos a un sitio https.
- Configuración de un **proxy** si la red dispone de él.
- Configuración de **cookies**.



5. Configuración servidor http: Windows (IIS 7.0)

- ▶ Internet Information Server o IIS es un software que integra:
 - ▶ IIS7
 - ▶ ASP.NET
 - ▶ Windows Communication Foundation
 - ▶ MS Windows Sharepoint Services
- ▶ Es modular y permite ampliar su funcionalidad inicial añadiendo nuevos componentes/características.
- ▶ Versiones existentes:
 - ▶ IIS 6.0 Windows 2003 y Windows XP
 - ▶ IIS 7.0 Windows 2008 R2 y Windows 7



5. Configuración servidor http: Windows (IIS 7.0)

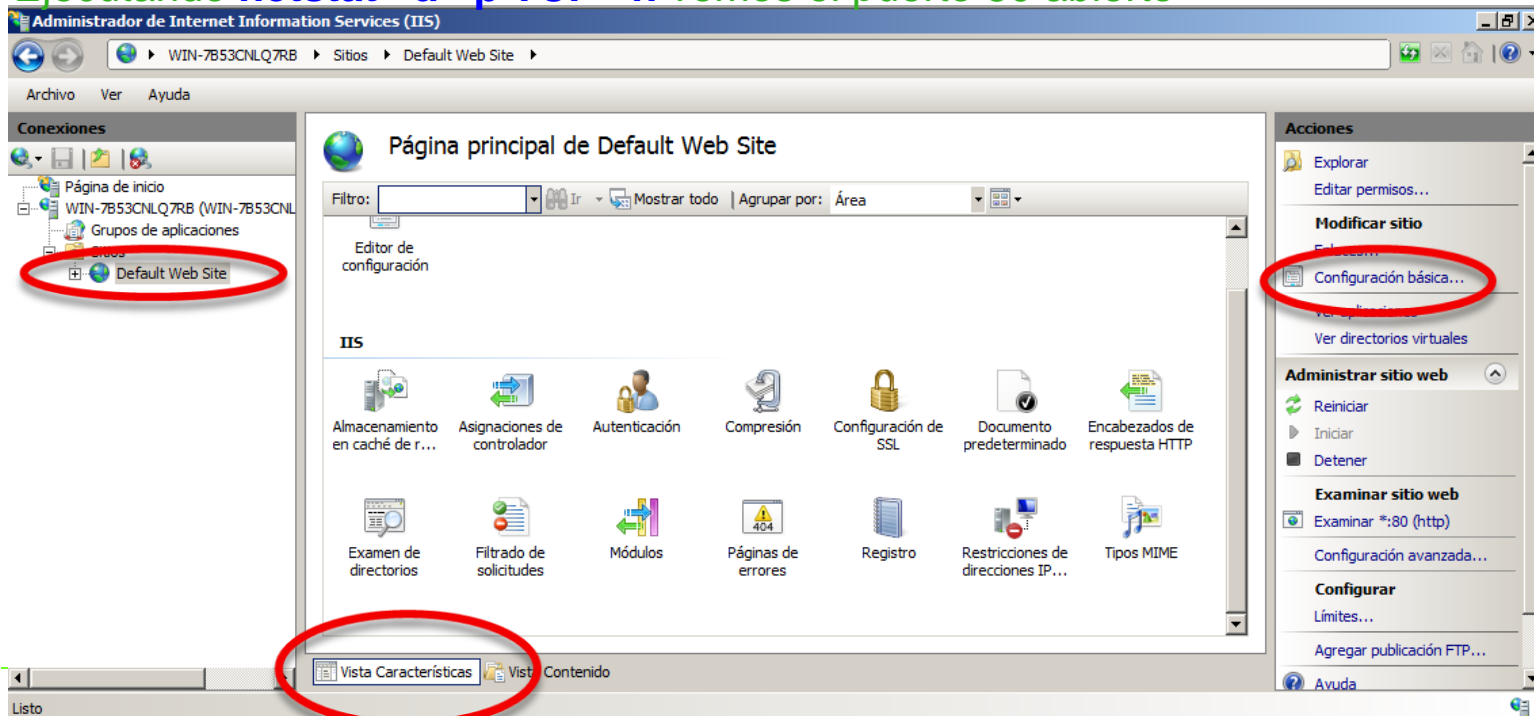
▶ Características de IIS7:

- ▶ Herramientas de administración mejoradas.
- ▶ Nueva herramienta de línea de comandos.
- ▶ Instalación modular basada en características:
 - ▶ IIS7 está compuesto por más de 40 módulos de características independientes.
 - ▶ De manera predeterminada sólo se instalan la mitad.
- ▶ Modelo de configuración distribuída.
- ▶ Diagnóstico y resolución de problemas.
- ▶ Arquitectura modular extensible: los usuarios pueden crear sus propios módulos.



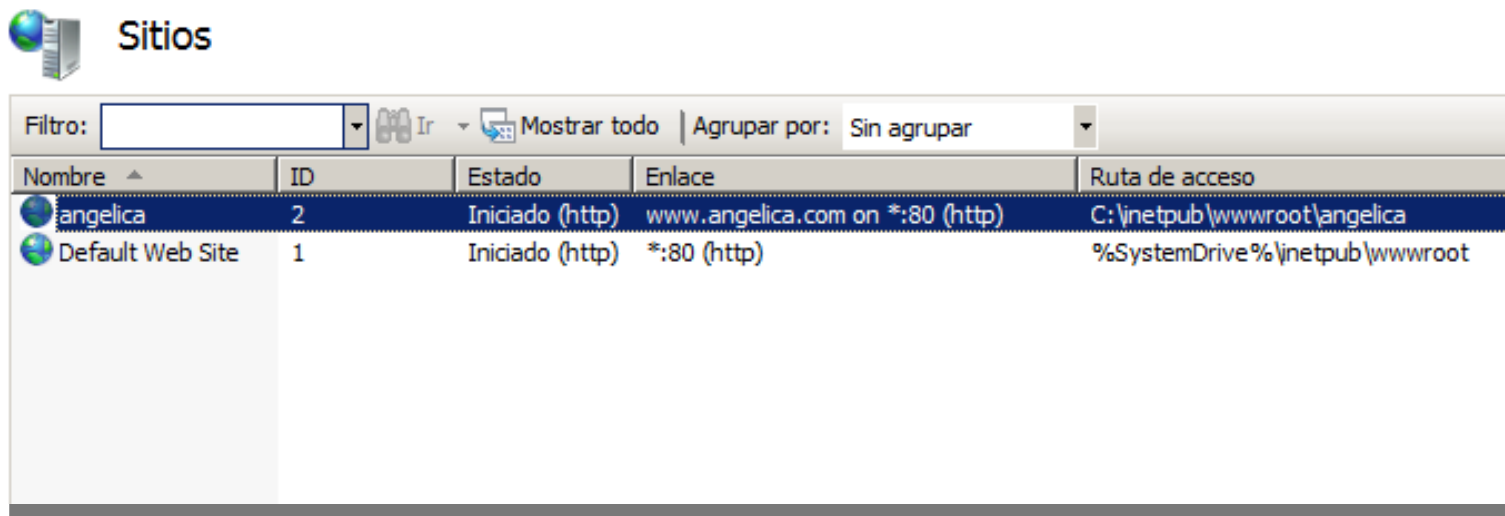
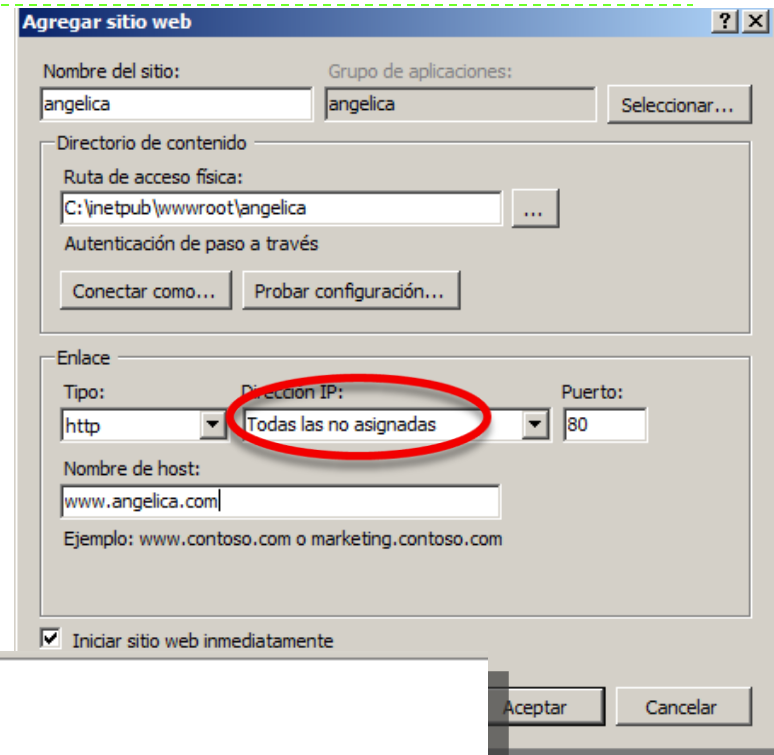
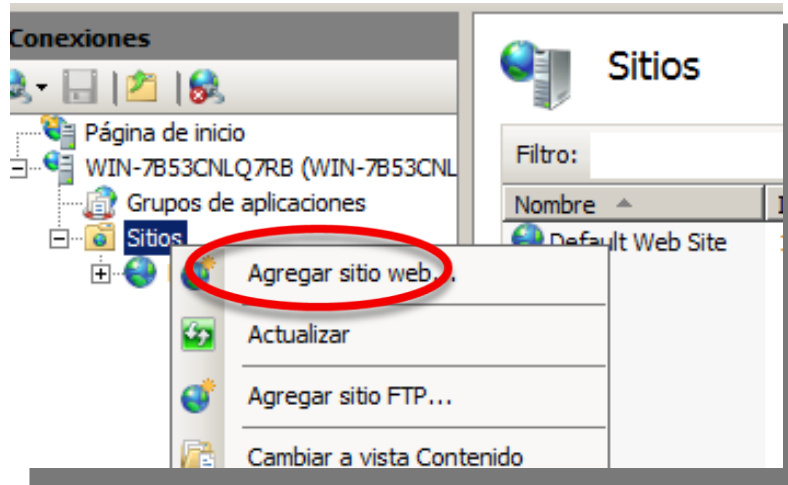
5. Configuración servidor http: Windows (IIS 7.0)

- Tras realizar la instalación, podremos acceder al Administrador de IIS.
- Por defecto se crea el sitio “Default Web Site”.
- En “configuración básica” veremos que el directorio donde está almacenado es c:\inetpub\wwwroot.
- Dentro de ese directorio se encuentra el archivo `iisstart.htm`. Si abrimos una navegador en el servidor y accedemos a la url <http://localhost> comprobaremos que el servidor está funcionando.
- Ejecutando `netstat -a -p TCP -n` vemos el puerto 80 abierto



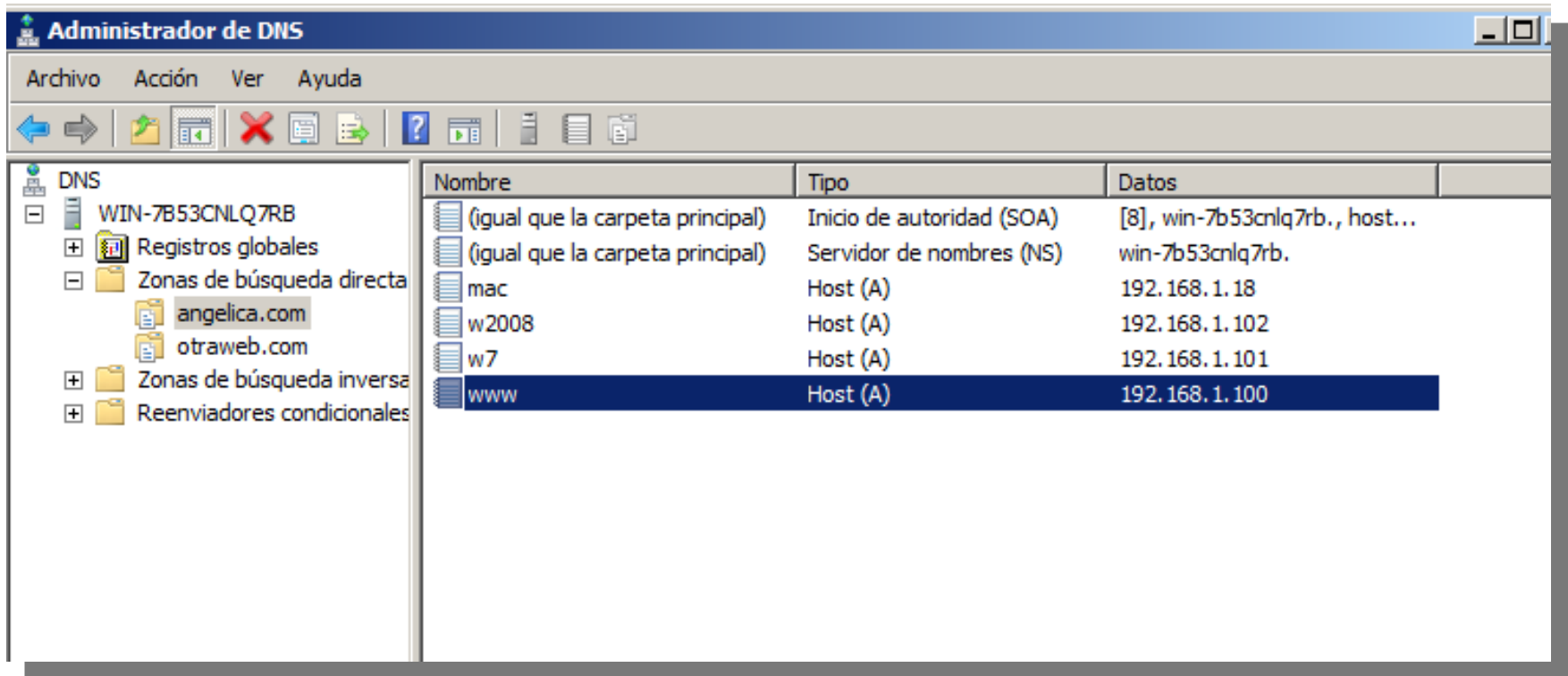
5. Configuración servidor http: Windows (IIS 7.0)

Cómo añadir un sitio nuevo:



5. Configuración servidor http: Windows (IIS 7.0)

Hay que añadir una entrada en el **servidor DNS**.



The screenshot shows the 'Administrador de DNS' (DNS Administrator) window. The left pane displays the tree structure of the DNS server 'WIN-7B53CNLQ7RB', including 'Registros globales' and 'Zonas de búsqueda directa' with sub-zones 'angelica.com' and 'otraweb.com'. The right pane shows a list of DNS records with the following columns: Nombre, Tipo, and Datos.

Nombre	Tipo	Datos
(igual que la carpeta principal)	Inicio de autoridad (SOA)	[8], win-7b53cnlq7rb., host...
(igual que la carpeta principal)	Servidor de nombres (NS)	win-7b53cnlq7rb.
mac	Host (A)	192.168.1.18
w2008	Host (A)	192.168.1.102
w7	Host (A)	192.168.1.101
www	Host (A)	192.168.1.100

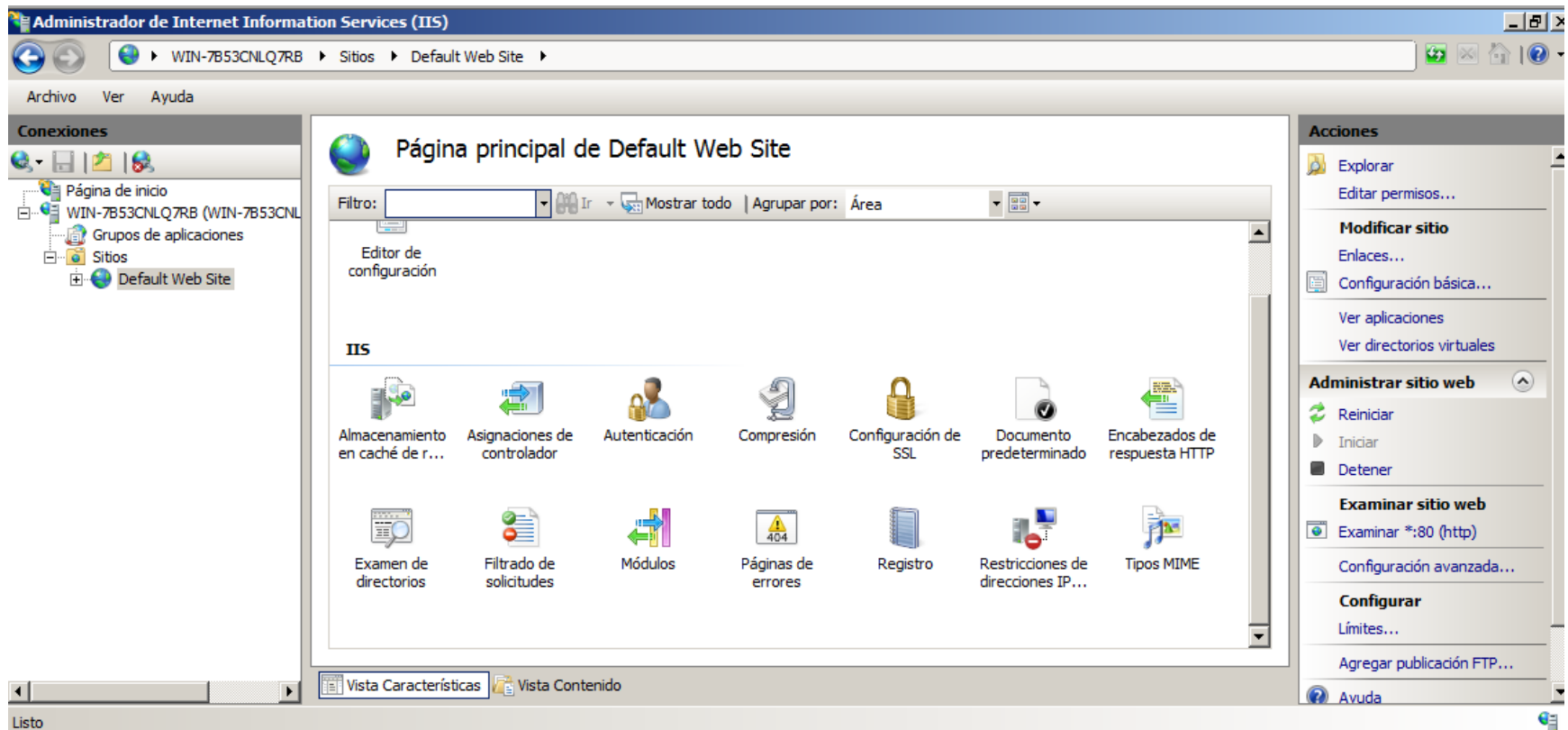
Actividad.

- ▶ Instalar un servidor IIS en una máquina Windows 2008.
- ▶ Crear un archivo llamado red.html con el siguiente código html y guárdalo en c:\inetpub\wwwroot.

```
<html>  
<body>  
<h1> Este es el servidor Web de TUNOMBRE. </h1>  
</body>  
</html>
```
- ▶ ¿Cuál es la URL para acceder a dicho archivo?
- ▶ Escríbela en un navegador y comprueba que se muestra el página web.
- ▶ Crea ahora una carpeta dentro de c:\inetpub\wwwroot llamada tunombre.
- ▶ Copia ahora red.html en c:\inetpub\wwwroot\tunombre y bórralo de la ubicación anterior.
- ▶ ¿Qué URL debes especificar para que se muestre red.html? Comprueba que funciona.
- ▶ Ahora vete a configuración básica y cambia el directorio raíz por c:\inetpub\wwwroot\tunombre. ¿Qué URL debes especificar ahora para que se muestre red.html? Comprueba que funciona.
- ▶ Vuelve a dejar la máquina con la configuración original. Deja el archivo red bajo el directorio c:\inetpub\wwwroot.

5. Configuración servidor http: Windows (IIS 7.0)

Ahora veremos algunas de las características disponibles para un sitio:





Documento
predeterminado

Especifica el nombre de la página web que se muestra por defecto cuando en la URL no se especifica ningún archivo concreto.



Documento predeterminado

Utilice esta característica para especificar los archivos predeterminados que se devolverán cuando un cliente no solicite un nombre de archivo específico. Establezca los documentos predeterminados en orden de prioridad.

Nombre	Tipo de ent...
Default.htm	Heredada
Default.asp	Heredada
index.htm	Heredada
index.html	Heredada
iisstart.htm	Heredada

Acciones

[Agregar...](#)

[Deshabilitar](#)

[Revertir a primaria](#)



[Ayuda](#)

[Ayuda en pantalla](#)



Examen de
directorios

En el caso de que no se encuentre
ningún archivo por defecto,
podremos dar la opción de mostrar
el contenido de la carpeta
especificada en la URL.





Examen de directorios


Utilice esta característica para especificar la información que se muestra en un listado de directorios.

- ☒ Hora
- ☒ Tamaño
- ☒ Extensión
- ☒ Fecha
 - ☐ Fecha larga

Acciones

-  Aplicar
-  Cancelar

[Deshabilitar](#)

-  Ayuda
- [Ayuda en pantalla](#)



Páginas de errores

Modificar página de errores personalizados

Código de estado:

403

Ejemplo: 404 o 404.2

Acción de respuesta

☐ Insertar contenido del archivo estático en respuesta de error

Ruta de acceso del archivo:

☐ Prueba a devolver el archivo de error en el lenguaje del cliente

☒ Ejecutar una dirección URL en este sitio

Dirección URL (relativa a la raíz del sitio):

/noencontrada.htm

Ejemplo: /ErrorPages/404.aspx

☐ Responder con una redirección 301

Dirección URL absoluta:

Acciones

Agregar...

Modificar...

Cambiar código de estado

☒ Quitar

Modificar configuración de característica...

☒ Ayuda

Ayuda en pantalla

Se debe crear ese archivo bajo el directorio raíz.



Páginas de errores

Utilice esta característica para configurar respuestas de errores de HTTP. Las respuestas de errores pueden ser páginas de errores personalizados o mensajes de error detallados que contienen información para la solución del problema.

Agrupar por

Código de

403

404

405

406

412

500

501

502

Modificar configuración de páginas de errores

Respuestas de errores

Cuando el servidor encuentra un error, devuelve:

☒ Páginas de errores personalizados

☐ Errores detallados

☐ Errores detallados para solicitudes locales y páginas de errores personalizados para solicitudes remotas

Página predeterminada

Ruta de acceso:

Tipo de ruta de acceso:

Archivo

Aceptar

Cancelar

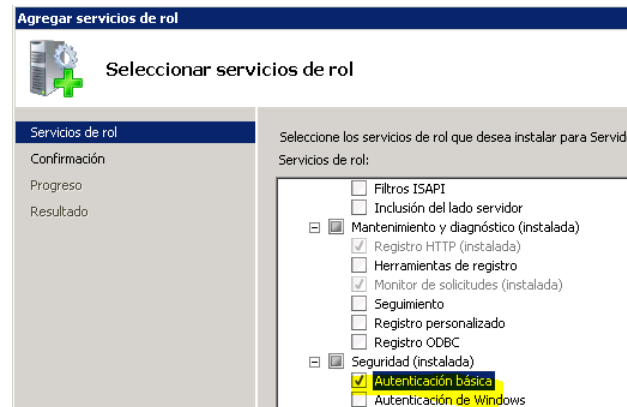
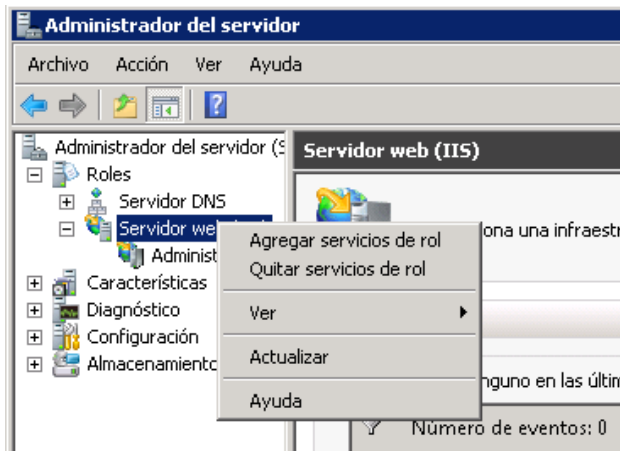
Actividad.

- ▶ Elimina iisstart.htm de la lista de documentos predeterminados.
- ▶ Accede a través de un navegador a la url <http://localhost>.
- ▶ ¿Qué está sucediendo? ¿Qué error se muestra?
- ▶ Añade ahora red.htm a la lista de documentos predeterminados y vuelve a conectarte a la url <http://localhost>. Explica que sucede ahora.
- ▶ Vuelve a eliminar red.htm de la lista de documentos predeterminados y configura una página de error personalizada asociada al error que se está mostrando. (Pista: Debes elegir la opción “Ejecutar una dirección URL en este sitio, e introducir /personalizada.htm habiendo creado previamente dicha página).

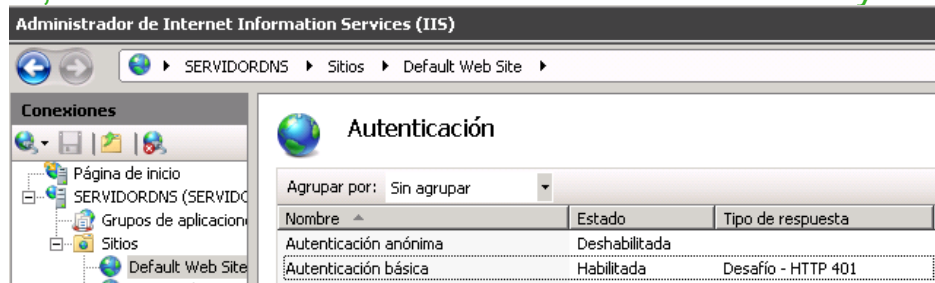




Debemos añadir al rol IIS el servicio de autenticación anónimo.



Tras añadirlo, debemos habilitar la autenticación básica y desactivar la autenticación anónima.



Tras habilitar la autenticación, hay que añadir el usuario correspondiente en el sistema.



Cuentas de usuario

Agregar o quitar cuentas de usuario

Administración

Realizar cambios en la cuenta de zipi

- el nombre de la cuenta
- la contraseña
- la imagen
- el tipo de cuenta
- la cuenta
- crear otra cuenta





Registro

Utilice esta característica para configurar el modo en que IIS registra las solicitudes en el servidor web.

Un archivo de registro por:

Sitio

Archivo de registro

Formato:

W3C

Campos seleccionados

Directorio:

%SystemDrive%\inetpub\logs\LogFiles

Examinar...

Codificación:

UTF-8

Conversión de archivos de registro

Seleccionar el método que utiliza IIS para crear un nuevo archivo de registro

Programación:

Diariamente

Tamaño máximo de archivo (en Bytes):

No crear nuevos archivos de registro

Usar la hora local para nomendatura y conversión de archivos

Acciones

Aplicar

Cancelar

Deshabilitar

Ver archivos de registro

Ayuda

Ayuda en pantalla

LogFiles

Equipo > Disco local (C:) > inetpub > logs > LogFiles

Buscar LogFiles

Organizar > Incluir en biblioteca > Compartir con > Nueva carpeta

Nombre	Fecha de modificación	Tipo	Tamaño
W3SVC1	30/10/2011 19:18	Carpeta de archivos	
W3SVC2	03/11/2011 8:58	Carpeta de archivos	
W3SVC3	30/10/2011 21:58	Carpeta de archivos	

3. Acceso seguro y utilización de certificados.

- ▶ Veremos:

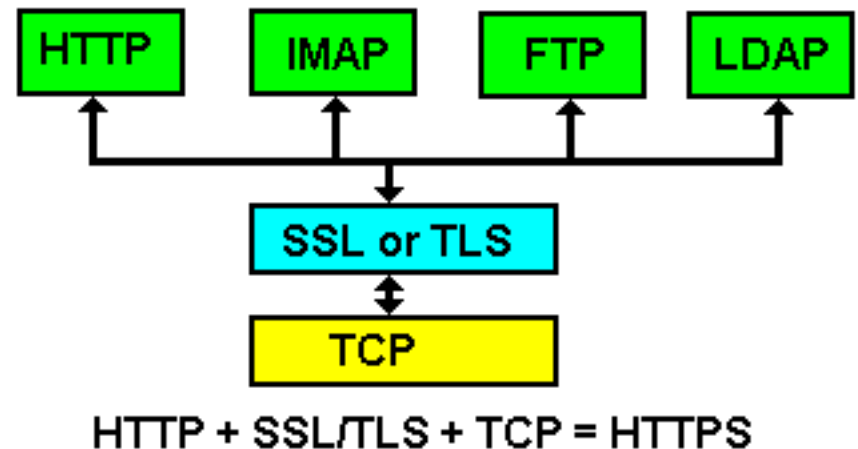
- ▶ SSL/TLS.
- ▶ HTTPS.
- ▶ Certificados autofirmados.
- ▶ Configuración de https y creación de protocolos autofirmados.



SSL/TLS

► **Protocolo SSL: Introducción.**

- ✓ SSL Secure Sockets Layer
- ✓ Protocolo creado en 1992 por Netscape para el intercambio de información de manera segura entre cliente y servidor.
- ✓ La IETF creó su propia versión a la que llamó TLS.
- ✓ Por eso vemos que se hace referencia a SSL/TLS.
- ✓ SSL es un protocolo que opera entre las capas de aplicación y de transporte.
- ✓ Puede recibir datos de varias aplicaciones: http, ftp, ldap, ...
- ✓ Opera sobre TCP en el nivel de transporte.

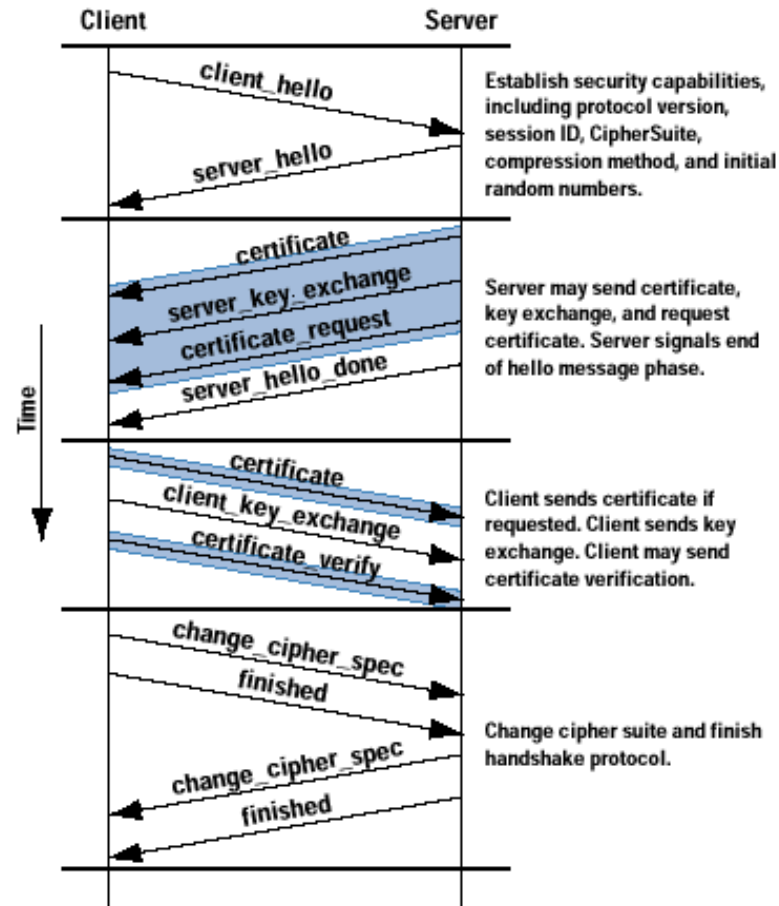


SSL/TLS

► **Protocolo SSL: Establecimiento de conexión.**

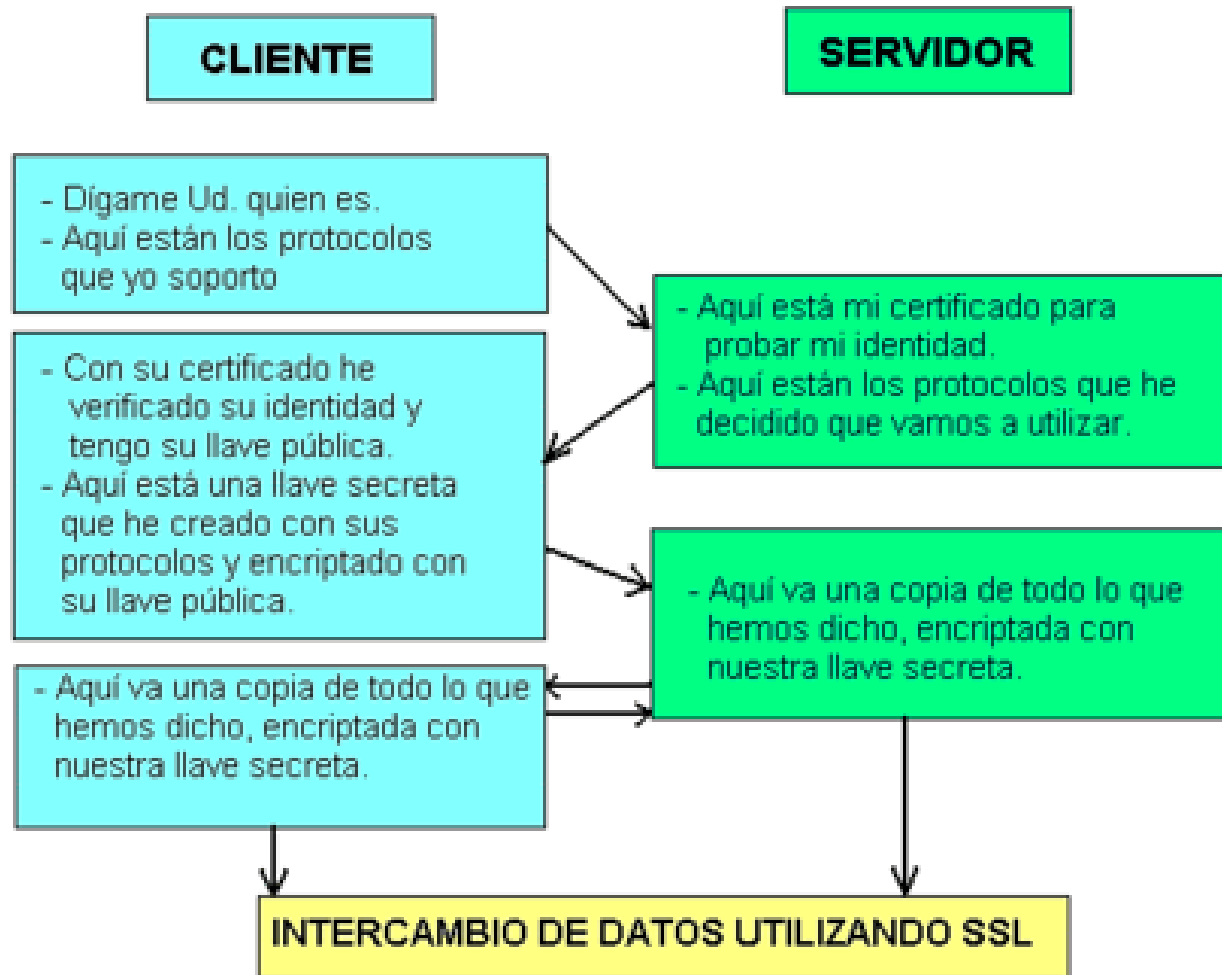
1. Cliente y servidor se ponen de acuerdo sobre la versión del protocolo SSL/TLS a utilizar, algoritmos de cifrado, métodos de compresión, el ID de la sesión, y un par de números aleatorios, uno de cada.
 1. El cliente enviará un mensaje SSL “Client Hello” con su información.
 2. El servidor responderá con un mensaje SSL “Server Hello” con su información a partir de lo recibido del cliente.
2. Servidor envía su certificado digital, su clave pública, etc. .
 1. El servidor enviará su certificado digital (X.509) en el mensaje Certificate.
 2. Tras enviar toda esta información, hasta tres mensajes podría enviar, indicará al cliente que ya ha terminado enviando el mensaje “Server_Hello_Done”.
3. Intercambio de certificados y envío de la clave pre-master.
 1. Tras recibir el “Server_Hello_Done” el cliente verificará que el certificado es válido y que los parámetros son aceptables.
 2. El cliente enviará entonces cifrado con la clave pública una secuencia de 49 bytes, pre-master, para cifrar los mensajes. Sólo el servidor la podrá descifrar con su clave privada. Ambos obtendrán a partir de dicha clave y del algoritmo elegido en la anterior fase, la clave master a usar.
 3. Además enviará su certificado, clave pública, etc, si se hubiera requerido.
4. Se envían los mensajes change_cipher_spec que indicarán que a partir de ese momento se usarán las claves acordadas.

SSL/TLS

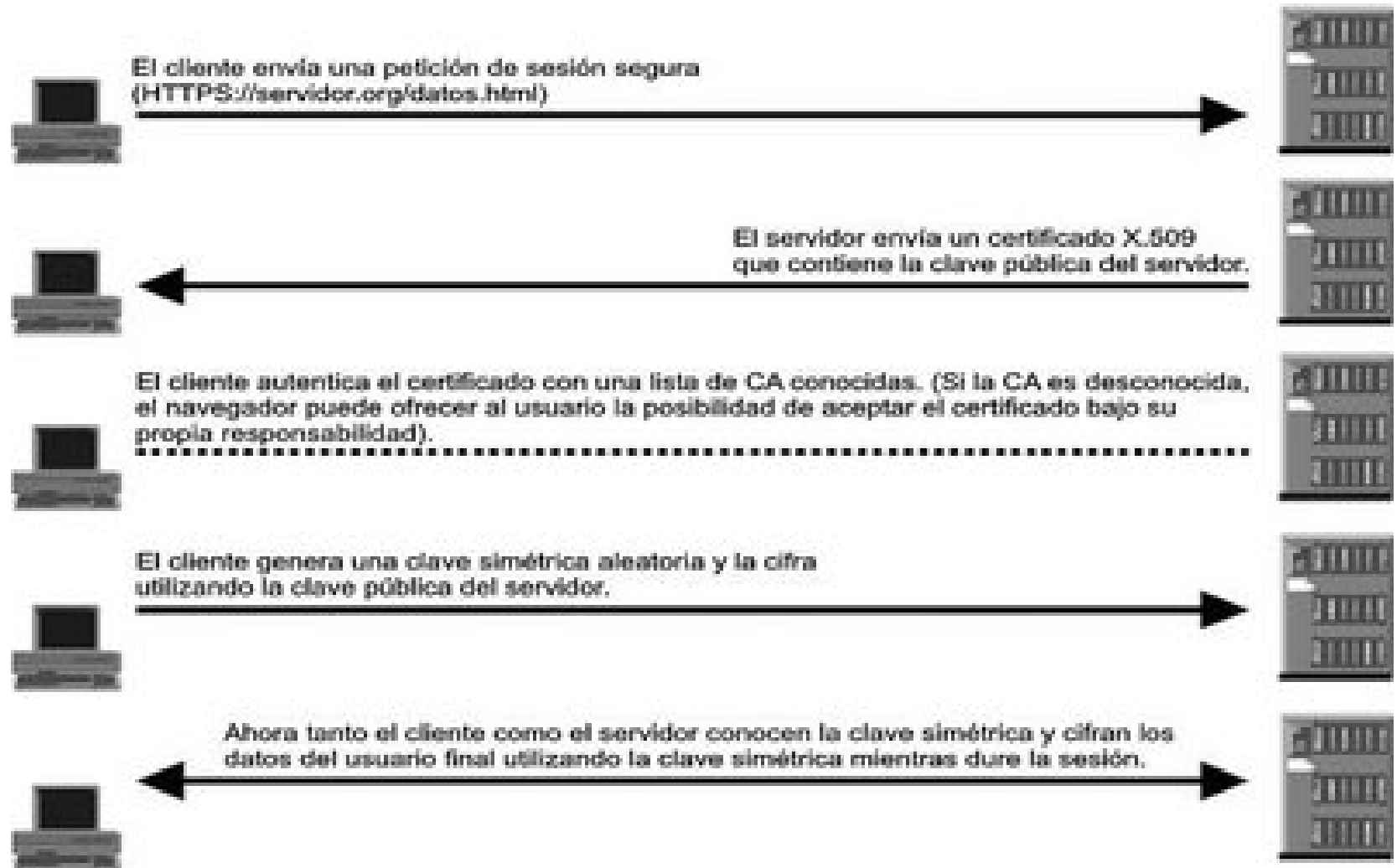


Note: Shaded transfers are optional or situation-dependent messages that are not always sent

HTTPS.



SSL/TLS



HTTPS.

- ✓ Con http la información viaja por la red en **texto claro**.
- ✓ Pero en ocasiones cliente y servidor puede requerir intercambiar información confidencial.
- ✓ Para paliar este problema surge https.
- ✓ https se apoya en una conexión establecida en **SSL/TLS**.
- ✓ La información http viajará encapsulada en el protocolo seguro SSL.
- ✓ El puerto para este tipo de conexiones será el **443** en lugar del 80.
- ✓ El cliente accederá a este servicio usando https en la URL en vez de http.
- ✓ Cuando accedemos a un sitio web con https, necesitamos conocer previamente que el servidor es realmente quien dice. Para demostrarlo ***deberá poseer un certificado que debe estar firmado por una CA*** que sea de nuestra confianza.
- ✓ La obtención de un certificado acreditado por una CA para nuestra web suele conllevar gastos.



Certificados autofirmados.

- ✓ Cuando accedemos a un sitio web con https, necesitamos conocer previamente que el servidor es realmente quien dice. Para demostrarlo deberá poseer un certificado que debe estar firmado por una CA que sea de nuestra confianza.
- ✓ La obtención de un certificado acreditado por una CA para nuestra web suele conllevar gastos.
- ✓ Alternativa: utilizar certificados autofirmados, que se pueden generar con herramientas software como OpenSSL.



Ejercicio

► Ejercicio:

- Investigar como se puede crear un certificado autofirmado en IIS7.
- Crearlo y comprobar su funcionamiento.
- Obtener una traza con Wireshark y comentar la información obtenida.



5. Sitios web virtuales.

▶ Alojamiento virtual de sitios web.

- ▶ Consiste en simular que existen varios hosts con sus respectivos sitios web sobre un solo servidor web en un mismo servidor web, es decir, alojar varios sitios web en un mismo servidor.
 - ▶ También se usan los términos hosts virtuales, servidores virtuales y sitios virtuales para referirse a este tipo de configuración.
 - ▶ Permite reducir el número de máquinas físicas necesarias para alojar millones de sitios web que existen en Internet y al mismo tiempo aprovechar mejor los recursos (uso de CPU, memoria, ...) de los equipos.
 - ▶ La mayoría de los sitios web actuales (Apache, IIS, ...) soportan estas funcionalidades.
 - ▶ Se pueden diferenciar tres tipos de alojamiento virtual:
 - ▶ Alojamiento virtual basado en IPs.
 - ▶ Alojamiento virtual basado en nombres.
 - ▶ Alojamiento virtual basado en puertos.
-



5. Sitios web virtuales.

▶ Alojamiento virtual basado en IPs.

- ▶ El servidor tendrá varias direcciones IP.
- ▶ Dependiendo de la dirección IP utilizada por el cliente, se mostrará un sitio web u otro.
- ▶ Es decir, es como si existieran varios servidores web, uno en cada dirección IP.
- ▶ Para realizar esto la máquina debe tener:
 - ▶ O bien varias tarjetas de red, una con cada dirección IP.
 - ▶ O bien, una tarjeta de red con varias direcciones IP (alias o interfaces virtuales).
- ▶ Debéis tener en cuenta, que en el servidor DNS habrá que añadir una entrada por cada servidor virtual.



5. Sitios web virtuales.

▶ Alojamiento virtual basado en nombres.

- ▶ El servidor permite alojar varios nombres de dominio sobre la misma dirección IP.
- ▶ Cada servidor virtual atiende las peticiones de un nombre de dominio.
- ▶ Esta es la forma de alojamiento más utilizada ya que se ahorra no solo recursos de la máquina sino que también direcciones IP.



5. Sitios web virtuales.

- ▶ Alojamiento virtual basado en puertos.
 - ▶ Cada servidor virtual atiende peticiones en una dirección IP y/o dominio:puerto diferentes.
 - ▶ Consiste en cambiar el alojamiento basado en IP y/o en nombres con el uso de varios puertos a la escucha.
 - ▶ Esta es la forma de alojamiento más utilizada ya que se ahorra no solo recursos de la máquina sino que también direcciones IP.



5. Sitios web virtuales.

► Ejemplos:

- **Basado en IP:** Tenemos una máquina que tiene dos IPs 10.0.0.1 y 10.0.0.2.

Queremos usarla para servir dos sitios web, uno se llama www.asir1.com y otro www.asir2.com.

Con el sistema basado en IP, www.asir1.com se serviría por ejemplo en la IP 10.0.0.1 y el otro www.asir2.com se serviría en otra IP, por ejemplo 10.0.0.2.

En el DNS se añadirá los registros:

www.asir1.com. IN A 10.0.0.1

www.asir2.com. IN A 10.0.0.2

- **Basado en nombre:** Tenemos una máquina con una sola IP.

Queremos usarla para servir dos sitios web. Cuando se acceda con el nombre www.asir1.com se servirá un sitio y cuando se acceda con www.asir2.com el otro.

- **Basado en puerto:** Tenemos una máquina con una sola IP.

Queremos usarla para servir dos sitios web en la máquina www.asir.com.
Dependiendo del puerto al que se conecten los clientes, se servirá una web u otra.

5. Sitios web virtuales.

► Ejemplos:

- **Basado en IP:** Tenemos una máquina que tiene dos IPs 10.0.0.1 y 10.0.0.2.

Queremos usarla para servir dos sitios web, uno se llama www.asir1.com y otro www.asir2.com.

Con el sistema basado en IP, www.asir1.com se serviría por ejemplo en la IP 10.0.0.1 y el otro www.asir2.com se serviría en otra IP, por ejemplo 10.0.0.2.

En el DNS se añadirá los registros:

www.asir1.com. IN A 10.0.0.1

www.asir2.com. IN A 10.0.0.2

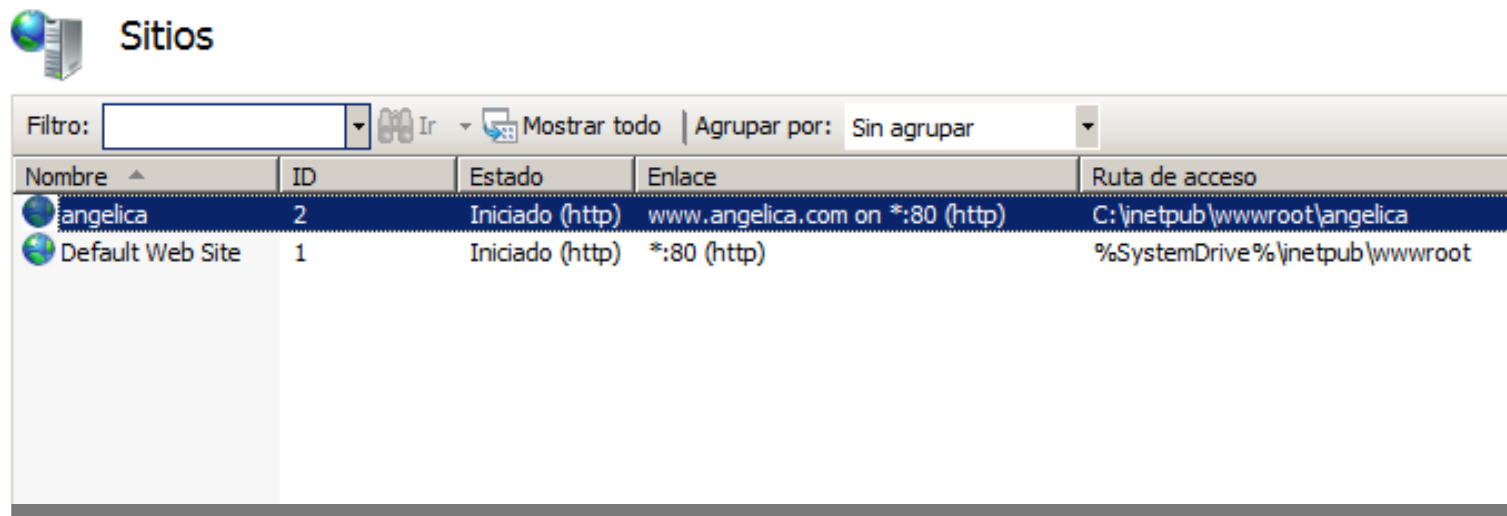
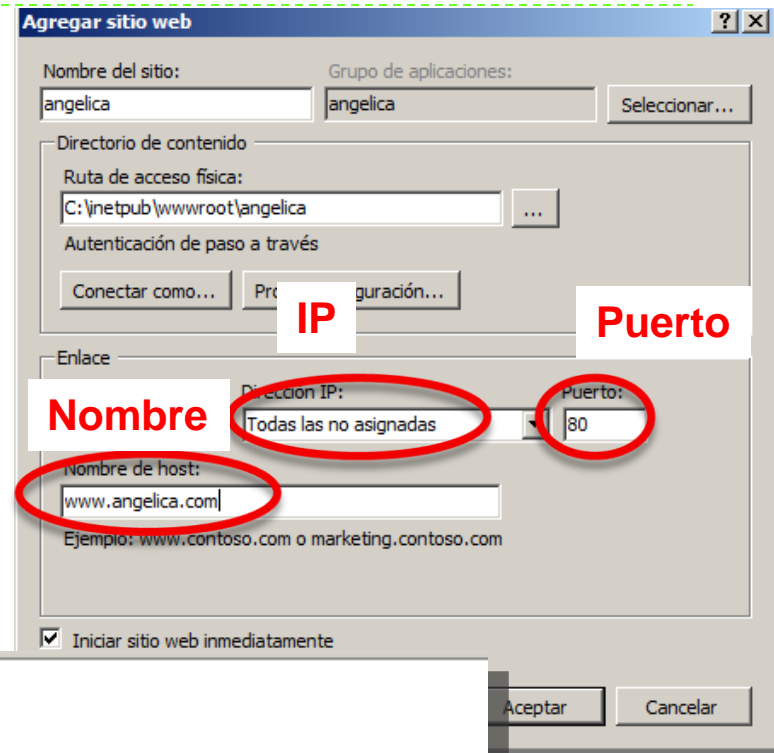
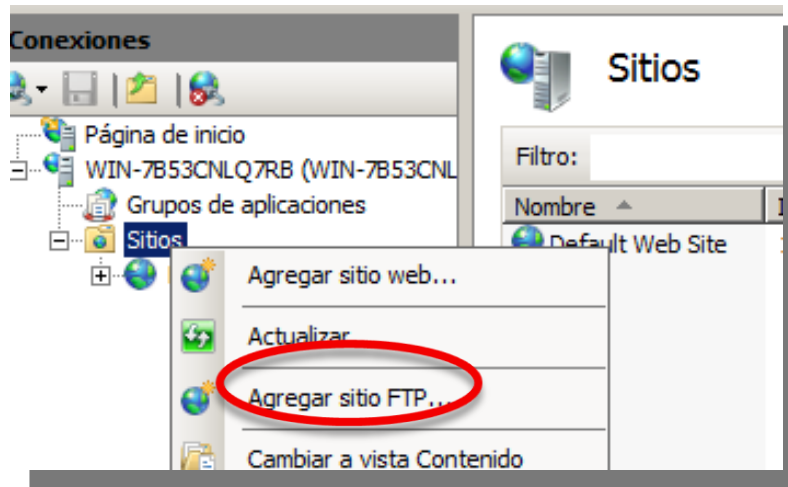
Cuando el navegador acceda a la url <http://www.asir1.com> usará la 10.0.0.1.

Cuando el navegador acceda a la url <http://www.asir2.com> usará la 10.0.0.2.



5. Configuración servidor virtual en Windows 2008.

Cómo añadir un sitio nuevo:



Actividad.

- ▶ Configurar el servidor DNS para que resuelva los nombres www.asir1.net y www.asir2.net.
 - ▶ Detén el servidor virtual por defecto.
 - ▶ Crear y habilitar el sitio www.asir1.net según las siguientes especificaciones:
 - ▶ Directorio raíz: c:\intepub\asir1.
 - ▶ Servirá index.html si no se indica ningún fichero en la URL.
 - ▶ Se mostrará un listado del directorio raíz si no se solicita ningún fichero.
 - ▶ Cuando se produzca un error 403 mostrará el mensaje “Página no encontrada www.asir1.net”.
 - ▶ Crear y habilitar el sitio www.asir2.net según las siguientes especificaciones:
 - ▶ Directorio raíz: c:\intepub\asir2.
 - ▶ Servirá indice.html si no se indica ningún fichero en la URL.
 - ▶ No se mostrará un listado del directorio raíz si no se solicita ningún fichero.
 - ▶ Cuando se produzca un error 403 mostrará el mensaje “Página no encontrada www.asir2.net”.
-



Actividad.

- ▶ Modifica los sitios anteriores para servir www.asir1.net en la IP 10.0.0.1 y www.asir2.net en 10.0.0.2 pero dentro de la misma máquina.
- ▶ Modifica los sitios para servir www.asir1.net en el puerto 80 y www.asir2.net en el puerto 8080.

