

Introducción

Antes de desarrollar nuestro proyecto web es importante tener en cuenta el nombre que vamos a darle así como donde va a estar ubicado físicamente. Para dar nombre a nuestro proyecto elegiremos un **nombre de dominio** y con él determinaremos si queremos ubicar el proyecto en nuestro propio **Data Center** o queremos subcontratar este servicio a una empresa externa especializada.

Si nuestro proyecto es un proyecto web para Internet yo personalmente apostaría por la subcontratación del servicio ya que en nuestro Data Center necesitaríamos recursos como contratación de ancho de banda, compra de servidores, personal y SAls que nos garanticen el tráfico y el funcionamiento 365x24 de nuestro proyecto. Si se trata de un proyecto no crítico como por el ejemplo una Intranet este servicio podría ubicarse en nuestro CPD.

Hosting, housing, servidor virtual,

Nuestro servidor va a tener diferentes servicios: servicio dns, dhcp, http, ftp, ssh, correo, que tendremos que configurar si nuestro proveedor no lo realiza por nosotros.

Capítulo 1

Servicio DNS

1.1.- Introducción

El **sistema de nombres de dominios** (DNS, Domain Name System) convierte nombres de dominios en direcciones IP y viceversa. Cada ordenador que forma parte de Internet posee una dirección IP única. Una dirección IP en IPv4 consiste en 4 números entre 0 y 255 separados por puntos, como por ejemplo 217.160.242.178. Las direcciones son difíciles de recordar y es más fácil acordarse de un nombre como por ejemplo profesordeinformatica.com que es el nombre que corresponde a la anterior dirección IP.

Cuando nosotros navegamos por Internet estamos utilizando un servidor de DNS que nos hace esa traducción de nombres a direcciones. Cuando realizamos una configuración manual de nuestra tarjeta de red los datos que introducimos son la dirección IP de nuestro equipo, la máscara de subred, la puerta de enlace y el servidor de DNS. Cuando la configuración de estos datos se realizan a través de un servidor DHCP es el propio servidor el que nos proporciona estos datos.

En este capítulo vamos a aprender que son los dominios de Internet, como podemos registrar un dominio de Internet, vamos a aprender a realizar peticiones sobre un servidor DNS y vamos a configurar los clientes y servidores de DNS, tanto en Linux como en Windows.

1.2.- Dominios

La **ICANN** (Internet Corporation for Assigned Names and Numbers o Corporación para la Asignación de Nombres y Números en Internet) es una corporación sin ánimo de lucro encargada de la gestión de los dominios.

En Octubre de 1998 la **IANA** (Internet Assigned Numbers Authority) como **InterNIC** (Internet Network Information Center), la entidad que realizó esta tarea hasta entonces, se reorganizaron bajo el nombre de ICANN. El registro de dominios era gestionado por Network Solutions .inc y hasta que fue privatizado en noviembre de 1999.

Los **TLDs** (Top Level Domain) son los dominios de primer nivel. Son los que representan el texto después del último punto de un dominio. Por ejemplo el TLD de www.google.es es el **.es**. Los dominios los vamos a dividir en dominios genéricos y dominios de país.

Dominios genéricos:

Los dominios genéricos gTLD (generic Top Level Domain) a su vez se dividen en dominios históricos, dominios patrocinados y dominios no patrocinados.

Dominios históricos

Son los siete dominios que nacieron con Internet:

profesordeinformatica.com

.com : Organizaciones comerciales.
.edu : Fines educativos.
.gov : Utilizado por el Gobierno Federal de Estados Unidos.
.info : Páginas web informativas.
.int : Organizaciones con tratados internacionales.
.mil : Militar. Departamento de Defensa de Estados Unidos.
.net : Red (network). Organizaciones relacionadas con Internet.
.org : Organizaciones que no podían clasificarse en otros dominios.

Los .com, .net y .org no tenían ninguna restricción. En un principio fueron creados para uso específico pero hoy en día se utilizan de modo general. Normalmente las empresas buscan un .com libre, si este no lo está registran un .net y sino un .org.

Dominios patrocinados

.aero: Está reservado empresas de transporte aéreo. Fue el primer dominio patrocinado en 2002 para una industria. Está patrocinado por SITA (Société Internationale de Télécommunications Aéronautiques).

.coop: Reservado sociedades cooperativas y lo patrocina DotCooperation LLC (dotCoop o puntoCoop).

.museum: Reservado para museos, asociaciones de museos y profesionales relacionados con estos. Fue el primer dominio patrocinado de nivel superior el 20 octubre del 2001.

.travel: Reservado para agencias de viajes, líneas aéreas, oficinas de turismo y otras actividades relacionadas con esta industria. Fue aprobado el 8 de abril de 2005.

.mobi: dispositivos móviles y páginas que ofrezcan servicios para estos. Fue creado en 2005 y está patrocinado por compañías como Google, Microsoft, Nokia, Samsung, Ericsson, Vodafone o Telefónica móviles.

.jobs: Reservado para sitios web relacionados con el empleo. Fue aprobado el 8 de abril de 2005.

.cat: Reservado para la lengua y cultura catalana. Fue aprobado el 15 de septiembre de 2005. Gestionado por la fundación puntCat.

.eus: Reservado para la lengua y cultura vasca. Fue autorizado el 14 de junio de 2013. Gestionado por la asociación puntuEus y estará en funcionamiento en 2014.

.gal: Reservado para la lengua y cultura gallega. Fue autorizado el 14 de junio de 2013. Gestionado por la asociación Puntogal y estará en funcionamiento en 2014.

El 20 de junio de 2011 la ICANN aprobó la liberación de los dominios de Internet. De esta forma los nombres podrán terminar con el nombre de la ciudad o una marca comercial.

Actualmente existen casi 2.000 dominios candidatos a ser aceptados clasificados en criterios como denominaciones genéricas solicitadas por compañías privadas como Amazon (.book, .music, .news), marcas comerciales (.seat, .xbox, .godaddy, .mango, .bmw) o denominaciones geográficas (.barcelona, .bcn, ..catalonia, .madrid, .paris, .roma, etc.).

Registrar un gTLD no es barato ya que hay que realizar un pago de 185.000 dólares, 135.00 €, en concepto de costes del proceso, que no te garantizan nada, ya que hay que solicitar una serie de auditorías y una vez aprobado se pagan 20.000 dólares anuales para su mantenimiento. Hay varias empresas que están pujando por un mismo dominio.

Listado de dominios candidatos:

<https://gtldresult.icann.org/application-result/applicationstatus/viewstatus>

Dominios no patrocinados

Son nuevos dominios que no están patrocinados.

.pro: Reservado para profesionales acreditados. En un principio ofrecía tres dominios de segundo nivel para abogados, médicos y contadores públicos (.law.pro, .med.pro, .cpa.pro). Fue creado en 2003 y en 2008 se ampliaron los términos para profesionales o titulaciones universitarias reconocidas.

.biz: Reservado para negocios, derivado de business. Muchas páginas de Bizkaia hacen de este dominio como si fuera de su provincia.

.name: Reservado para personas.

Dominios de país o geográficos

Los dominios de Estado o ccTLD (country code Top Level Domain) son dominios de primer nivel correspondientes a países o zonas geográficas.

Google da mucha importancia a los dominios propios de cada país cuando se realizan búsquedas desde dicho país, es decir, si estamos haciendo una búsqueda en España el algoritmo de búsqueda de Google nos posicionará mejor si tenemos un dominio .es a si tenemos un dominio .fr.

Ejemplos de estos dominios son:

.es: España. Gestionado por red.es perteneciente Ministerio de Industria, Energía y Turismo. A partir de 2005 se crearon dominios de tercer nivel como .com.es, .org.es, .nom.es. Hasta noviembre de 2005 para registrar un .es era obligatorio tener una marca registrada o un nombre de empresa y a partir de esta fecha desaparecen las restricciones, lo que hace que en número de registro de dominios se multiplicase a partir de esta fecha.

.co: Colombia, **.uk:** Reino Unido, **.fr:** Francia, **.po:** Portugal.

.tk: Archipiélago neozelandés de Tokelau. Conocidos por ofertar dominios gratuitos en los que

incluyen publicidad y redirección de correo electrónico.

.eu: Reservados para páginas web la Unión Europea. Es administrado por la entidad sin ánimo de lucro EURid. Ha tenido mucho éxito en Portugal, Rumanía y Galicia ya que en gallego, rumano y portugués **eu** significa yo. En Euskadi también ha sido utilizado asociándolo a su lengua el euskera.

Registro de dominios

Hasta 1999 la compra de dominios se realizaban a través de **Internic o Network Solutions** que era una empresa privada encargada de la venta de los mismos. El precio de un dominio era de **70\$** por 2 años.

En 1999 se **liberalizó** la venta de dominios históricos lo que hizo que el precio de los dominios bajasen considerablemente. Las **empresas acreditadas** por la ICANN para la venta de dominios la podemos encontrar en <http://www.icann.org/registrar-reports/accredited-list.html>. Actualmente los dominios históricos se pueden comprar por un precio inferior a **10\$**.

Para la compra de dominios de país como un .es se realizaba a través de la web de EsNic nic.es. En 2005 se liberalizó la venta y las condiciones de registro.

Las empresas líderes mundiales en la venta de dominios son Godaddy, 1and1, Tucows, Network Solutions o Register. Entre las empresas españolas que se dedican al registro de dominios Arslys, Acens, Dinahosting o Nominalia.

Muchas empresas ofrecen servicios de alojamiento junto a la venta de un nombre o viceversa. Con lo que a veces confundimos el registro de dominio con alojamiento web. Cuando contratamos el dominio simplemente reservamos el dominio e indicamos donde se encuentran los servidores de DNS.

Cuando registramos un dominio básicamente necesitamos indicar los siguientes datos:

Registrante o Titular (Registrant): Es la empresa o persona física que registra un nombre de dominio. Es el representante legal sobre el dominio.

Contacto Administrativo (Administrative Contact): La empresa o persona física encargada de interactuar con el dominio en lugar del titular.

Contacto técnico (Technical Contact): La empresa o persona física encargada de realizar las gestiones técnicas, como el mantenimiento de los registros de recursos del servidor DNS del dominio.

Contacto de Pago (Billing contact): Empresa o persona física encargada de realizar el pago inicial y sucesivas renovaciones. Este contacto ya no se suele pedir.

Duración: Durante cuantos años queremos registrar el dominio, como máximo se puede registrar cien años y como mínimo 1. Normalmente se suelen aplicar descuentos por registros de más de 2 años.

Servidores de DNS: Nos van a indicar como mínimo dos servidores donde se encuentran los registros del dominio.

Muchas veces las empresas registradoras nos facilitan el relleno de datos. Ponen nuestro nombre como titular, contacto administrativo y ponen sus datos como contacto técnico. Registran el dominio por un año y ponen como servidor de DNS sus propios servidores.

El pago del dominio inicialmente se solía hacer mediante pago con tarjeta de crédito pero actualmente muchas empresas ofrecen otras alternativas de pago como la domiciliación bancaria por cuenta o el pago por ebay. La domiciliación bancaria te permite olvidarte de renovaciones ya que los proveedores la hacen automáticamente cargándola sobre tu cuenta. Cuidado si una tarjeta de crédito expira ya que no se podrá cargar la renovación del pago de un dominio y se podría perder dicho dominio.

Yo personalmente nunca registraría directamente un dominio en Network Solutions (para dominios gTLD) o en dominios.es (para dominios ccTLD .es) ya que el precio va a ser muy superior a cuando se hace con la propia empresa registradora acreditada. Además muchas veces te ofrecen servicios adicionales gratuitos como DNS, hosting o correo electrónico.

Práctica. El comando whois

Vamos a instalar en un Linux el paquete whois que nos va a dar información detallada de cada dominio y vamos a intentar averiguar los datos con los que se han registrado diferentes dominios.

```
$ apt-get install whois
```

```
$ whois egibide.org
```

```
whois egibide.org
```

Access to .ORG WHOIS information is provided to assist persons in determining the contents of a domain name registration record in the Public Interest Registry registry database. The data in this record is provided by Public Interest Registry for informational purposes only, and Public Interest Registry does not guarantee its accuracy. This service is intended only for query-based access. You agree that you will use this data only for lawful purposes and that, under no circumstances will you use this data to: (a) allow, enable, or otherwise support the transmission by e-mail, telephone, or facsimile of mass unsolicited, commercial advertising or solicitations to entities other than the data recipient's own existing customers; or (b) enable high volume, automated, electronic processes that send queries or data to the systems of Registry Operator, a Registrar, or Afilias except as reasonably necessary to register domain names or modify existing registrations. All rights reserved. Public Interest Registry reserves the right to modify these terms at any time. By submitting this query, you agree to abide by this policy.

Domain ID:D165343777-LROR
Domain Name:EGIBIDE.ORG
Created On:20-Apr-2012 18:49:15 UTC
Last Updated On:07-Sep-2012 00:24:48 UTC
Expiration Date:20-Apr-2014 18:49:15 UTC
Sponsoring Registrar:Nominalia Internet S.L. (R89-LROR)
Status:OK
Registrant ID:a58426ede8cc
Registrant Name:Fundacion Benefico Docente Jesus Obrero
Registrant Organization:Fundacion Benefico Docente Jesus Obrero
Registrant Street1:Calle Francia 32
Registrant Street2:
Registrant Street3:
Registrant City:Vitoria
Registrant State/Province:VI
Registrant Postal Code:01002
Registrant Country:ES
Registrant Phone:+34.945000333
Registrant Phone Ext.:
Registrant FAX:
Registrant FAX Ext.:
Registrant Email:info@area52.es
Admin ID:a5bb3fb6773f
Admin Name:Rojo Martinez Ernesto
Admin Organization:Ernesto Rojo Martinez
Admin Street1:Calle Francia 32
Admin Street2:
Admin Street3:
Admin City:Vitoria
Admin State/Province:VI
Admin Postal Code:01002
Admin Country:ES
Admin Phone:+34.945000333
Admin Phone Ext.:
Admin FAX:
Admin FAX Ext.:
Admin Email:info@area52.es
Tech ID:NIC_PIR_TECH
Tech Name:Technical Department
Tech Organization:Nominalia Internet, S.L
Tech Street1:Josep Pla 2, Torres Diagonal Litoral, Edificio B3, planta 3-D
Tech Street2:
Tech Street3:
Tech City:Barcelona
Tech State/Province:Barcelona
Tech Postal Code:08019
Tech Country:ES
Tech Phone:+34.935074360

Tech Phone Ext.:
Tech FAX:+34.933102360
Tech FAX Ext.:
Tech Email:tecnico@nominalia.com
Name Server:DNS1.NOMINALIA.COM
Name Server:DNS2.NOMINALIA.COM
Name Server:
Name Server:
DNSSEC:Unsigned

Si probamos con un dominio .es vemos que whois nos dice que no tiene datos y nos remite a la página nic.es.

Muchas empresas de registro de dominios ofrecen la posibilidad de ocultar estos datos previo pago.

Ejercicio 1.1

Obtén los datos del titular, contacto administrativo, técnico y servidores DNS de los dominios google.com, google.es, terra.es y yahoo.com.

Ejercicio 1.2

Realiza una tabla comparativa de los precios que ofrecen diferentes empresas registradoras de dominios.

Ejercicio 1.3

Obtener información adicional desde la web whois.sc de los dominios del ejercicio 1.

Ejercicio 1.4

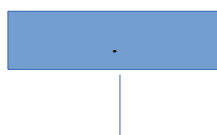
Registra un dominio en .tk en dom.tk

Jerarquía DNS

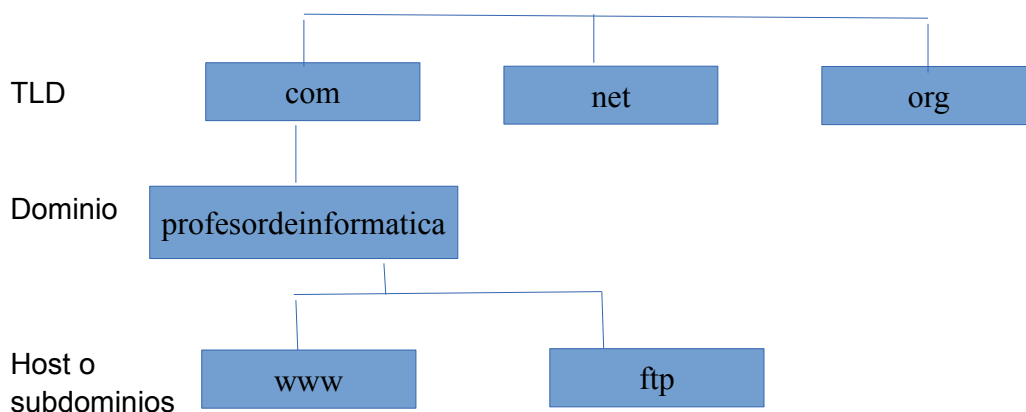
La estructura de los sistemas de nombres de dominios tienen una estructura jerárquica arborescente donde en la parte superior nos encontraremos en el primer nivel 0 con el nodo raíz “.”. En el nivel 1 nos vamos a encontrar con los TLD, en el segundo nivel con los nombres de los dominios y en el tercer nivel con los subdominios.

Todos los elementos de la jerarquía van a contar con al menos un carácter y como máximo 63 caracteres alfanuméricos (además del símbolo -). Nunca comenzarán por “-”.

Raíz



profesordeinformatica.com



El FQDN (Fully Qualified Domain Name o Nombre de dominio completo) se obtiene a partir del árbol, construyendo el dominio desde abajo hasta arriba, incluido el punto final y como máximo tiene 256 caracteres. Por ejemplo: www.profesordeinformatica.com.

Tipo de servidores

Podemos clasificar los servidores en los siguientes tres tipos:

Servidor primario o maestro: Van a tener la información de zonas en sus propios archivos. Cuando queremos añadir modificar cualquier información de las zonas lo tenemos que hacer aquí.

Servidor secundario o esclavo: Obtiene la información de las zonas de un servidor primario (o que tiene autoridad sobre esa zona). Contiene una copia de los archivos de zona.

Servidor caché: No contienen ninguna información ni copia de las zonas, su función es acelerar las consultas realizadas por los clientes sobre nombres de dominios.

Un servidor de nombres **caché** tiene sentido utilizarlo en una empresa con **varios ordenadores** con acceso a Internet, para que a la hora de resolver nombres, no se tenga que salir siempre a Internet. La primera vez que dese un puesto se consulta un dominio se consulta la resolución del mismo en su servidor autoritario y se guarda en la caché del servidor durante un tiempo (**minimun TTL o tiempo de vida mínimo**). Durante ese tiempo de vida cada consulta que se realice cualquier puesto o cliente lo resolverá directamente con la información que contiene la caché. Una vez finalizado ese tiempo la primera consulta se vuelve a realizar en Internet sobre el propio servidor autoritario y se vuelve a guardar en el servidor caché.

Además del tiempo de vida o TTL tenemos otros tres tiempos que afectan a los servidores DNS:

- **Tiempo de actualización** (refresh): Cada cuanto tiempo el servidor secundario obtiene o copia la información de las zonas de un servidor primario.
- **Tiempo de reintentos** (retry): Si el servidor primario está caído cada cuanto tiempo se va a reintentar obtener o copiar la información de las zonas.

- **Tiempo de caducidad** (expire): Si el servidor primario sigue caído durante este tiempo la información de zonas se borrará del servidor secundario.

Delegación de dominios

DNS es una base de datos **distribuida** en diferentes servidores lo que hace que su administración sea **descentralizada**. Esto se consigue gracias a la **delegación de dominios**: Por ejemplo la ICANN delega el dominio de nivel superior ".es" a ESNIC. Cuando un usuario compra un dominio de segundo nivel .es, ESNIC delega la autoridad del dominio sobre el comprador. El comprador a su vez podría crear subdominios y delegar la autoridad de estos en otros usuarios.

El servidor almacena diferentes partes del dominio denominadas **zonas**. Cada servidor puede tener diferentes zonas definidas en él. Por ejemplo, si tenemos un servidor primario que contiene la información de los dominios profe1.com y profe2.com contendrán sus respectivas zonas.

Registro de Recursos

Las zonas a su vez se dividen en los registros de recursos o RR. Su estructura es la siguiente:

Nombre de dominio (FQDN)	TTL	Tipo	Clase	Rdata
--------------------------	-----	------	-------	-------

Donde el **nombre de dominio** es un nombre FQDN (terminado en punto). En caso de que falte el punto el nombre es relativo al dominio principal.

Los registros para cada dominio tienen un **TTL** mínimo, que es el que va a mantenerse en los servidores de DNS caché.

El **tipo** puede ser uno de los siguientes:

- **SOA** (Start of Authority): Comienzo de autoridad.
- **NS** (Name Server): El servidor de nombres.
- **A** (Address): Sirve para hacer la correspondencia entre el nombre y la dirección IP.
- **CNAME** (Canonical Name): Nos permite crear alias de nombres.
- **MX** (Mail eXchanger): Nos indica el servidor de correo electrónico. A donde van los correos del tipo usuario@dominio.
- **PTR** (Pointer): Nos permite hacer resolución inversa.
- **HINFO**: Información hardware del servidor y de su sistema operativo. Es conveniente no dar información ya que puede ser utilizada por piratas informáticos.
- **SRV** (Service): Servicio.
- **TXT**: Almacena cualquier información.
- **AAAA**: Para añadir direcciones IPv6 vamos a utilizar el registro de recursos AAAA.

La **clase** en nuestro caso va a tener el valor **IN**, que indica que se trata de Internet.

El **RDATA** tiene los datos relacionados con el dominio:

A: La dirección IPv4.

AAAA: La dirección IPv6.

CNAME: El nombre del dominio.

MX: La prioridad (16 bits), seguida de nombre del ordenador.

NS: El nombre del servidor de nombres.

PTR: El nombre del dominio. In-arpa-ddr

SOA: nombre_servidor, correo_del_administrador (se sustituye una @ por un ".") y entre paréntesis (Número de serie y los 4 tiempos: refresco, reintento, expiración y mínimo TTL).

Ejemplo de zona del dominio profesordeinformatica.com

\$TTL 38400

```
profesordeinformatica.com. IN      SOA  servidor.profesordeinformatica.com. aitor.kaixo.com.
(2; Serial
 604800; Refresh
 86400; Retry
 2419200; Expire
 38400); Negative Cache TTL
```

```
profesordeinformatica.com.      IN      NS      servidor.profesordeinformatica.com.
profesordeinformatica.com.      IN      A        172.20.24.115
servidor.profesordeinformatica.com. IN      A        172.20.24.115
p01.profesordeinformatica.com.  IN      A        172.20.24.101
p02.profesordeinformatica.com.  IN      A        172.20.24.102
p03                             IN      A        172.20.24.103
p04.profesordeinformatica.com.  IN      A        172.20.24.104
alumno2                         IN      CNAME    p02
alumno3                         IN      CNAME    p03.profesordeinformatica.com.
profesordeinformatica.com.      IN      MX      1      p03.profesordeinformatica.com.
profesordeinformatica.com.      IN      MX      10     p02.profesordeinformatica.com.
profesordeinformatica.com.      IN      AAAA    2600:1800:5::10
```

Podemos definir un TTL general y en el resto de registros no añadir el TTL.

Se dice que para que un dominio sea **autoritario** por lo menos tiene que tener un registro SOA y por lo menos un registro NS.

El **número de serie** del registro SOA sirve para que el servidor secundario sepa cuando se han cambiado los registros de la base de datos. Su valor puede ser un valor correlativo y cada vez que se modifica se añade en 1. También podría tener la fecha con el número correlativo. Ejemplo: 2013091001.

Los **tiempos de los registros** incluidos en los del soa se pueden definir en segundos o utilizando la sintaxis en semanas (w), días(d), horas (h), minutos (m) y segundos (s). Por ejemplo, 1W3D8H20M. Si el valor del TTL es 0 significa que no se almacena en caché.

profesordeinformatica.com

Si un dominio no termina en punto se completa con el dominio principal. Es decir el dominio FQDN de alumno2 es alumno2.profesordeinformatica.com. (acabado en ".").

El comando nslookup

El comando nslookup nos permite realizar peticiones directamente sobre servidores de DNS y es válido tanto para Windows como para Linux.

Tiene dos modos de realizar la solicitud de información:

1.- Modo normal:

nslookup dominio

Dado un nombre de devuelve la dirección IP.

nslookup dirección_ip

Dada una dirección IP te devuelve el nombre.

C:/> nslookup profesordeinformatica.com

Server: 127.0.1.1

Address: 127.0.1.1#53

Non-authoritative answer:

Name: profesordeinformatica.com

Address: 217.160.242.178

C:/> nslookup 217.160.242.178

Server: 127.0.1.1

Address: 127.0.1.1#53

Non-authoritative answer:

178.242.160.217.in-addr.arpa name = clienteservidor.es.

Authoritative answers can be found from:

2.- Modo interactivo

nslookup

Hacemos directamente peticiones sobre el servidor de nombres que tenemos configurado en nuestro ordenador.

nslookup - dirección_ip (o nombre del servidor de nombres)

Hacemos peticiones sobre el servidor de nombres que indicamos después del guión.

Seguidamente vamos a introducir el tipo de registro que queremos introducir así como el dominio del cual queremos obtener información.

\$ nslookup - 8.8.8.8

> set type=NS

> profesordeinformatica.com

Server: 8.8.8.8

profesordeinformatica.com

Address: 8.8.8.8#53

Non-authoritative answer:

profesordeinformatica.com nameserver = ns64.1and1.es.

profesordeinformatica.com nameserver = ns63.1and1.es.

Authoritative answers can be found from:

> set type=A

> ns64.1and1.es

Server: 8.8.8.8

Address: 8.8.8.8#53

Non-authoritative answer:

Name: ns64.1and1.es

Address: 217.160.83.170

\$ nslookup - 8.8.8.8

> set type=SOA

> profesordeinformatica.com

Server: 8.8.8.8

Address: 8.8.8.8#53

Non-authoritative answer:

profesordeinformatica.com

origin = ns63.1and1.es

mail addr = hostmaster.1and1.es

serial = 2012092501

refresh = 28800

retry = 7200

expire = 604800

minimum = 86400

Authoritative answers can be found from:

> **set type=NS**

> **profesordeinformatica.com**

Server: 8.8.8.8

Address: 8.8.8.8#53

Non-authoritative answer:

profesordeinformatica.com nameserver = ns64.1and1.es.

profesordeinformatica.com nameserver = ns63.1and1.es.

Authoritative answers can be found from:

> **set type=A**

> **www.profesordeinformatica.com**

Server: 8.8.8.8

Address: 8.8.8.8#53

profesordeinformatica.com

Non-authoritative answer:

Name: www.profesordeinformatica.com

Address: 217.160.242.178

> **set type=MX**

> **profesordeinformatica.com**

Server: 8.8.8.8

Address: 8.8.8.8#53

Non-authoritative answer:

profesordeinformatica.com mail exchanger = 10 mx01.1and1.es.

profesordeinformatica.com mail exchanger = 10 mx00.1and1.es.

Authoritative answers can be found from:

> exit

Podemos obtener toda la información como en **tipo ANY**.

\$ nslookup

> **set type=ANY**

> **profesordeinformatica.com**

Server: 127.0.1.1

Address: 127.0.1.1#53

Non-authoritative answer:

profesordeinformatica.com

origin = ns63.1and1.es

mail addr = hostmaster.1and1.es

serial = 2012092501

refresh = 28800

retry = 7200

expire = 604800

minimum = 86400

profesordeinformatica.com mail exchanger = 10 mx00.1and1.es.

profesordeinformatica.com mail exchanger = 10 mx01.1and1.es.

Name: profesordeinformatica.com

Address: 217.160.242.178

profesordeinformatica.com nameserver = ns64.1and1.es.

profesordeinformatica.com nameserver = ns63.1and1.es.

Authoritative answers can be found from:

profesordeinformatica.com nameserver = ns63.1and1.es.

profesordeinformatica.com nameserver = ns64.1and1.es.

mx00.1and1.es internet address = 212.227.17.175

mx00.1and1.es internet address = 212.227.15.134

mx01.1and1.es internet address = 212.227.15.150

mx01.1and1.es internet address = 212.227.17.191

ns63.1and1.es internet address = 217.160.82.170

ns63.1and1.es has AAAA address 2001:8d8:fe:53:0:d9a0:52aa:100

profesordeinformatica.com

ns64.1and1.es internet address = 217.160.83.170
ns64.1and1.es has AAAA address 2001:8d8:fe:53:0:d9a0:53aa:100

Ejemplo resolución inversa

\$ nslookup

> set type=PTR

> 8.8.8.8

Server: 127.0.1.1

Address: 127.0.1.1#53

Non-authoritative answer:

8.8.8.8.in-addr.arpa name = google-public-dns-a.google.com.

Authoritative answers can be found from:

Ejercicio 1.5

Escribe los registros zonas de los dominios google.com, google.es, terra.es y yahoo.com a partir de los datos que puedes obtener de Internet.

Ejercicio 1.6

Realiza una comparación de los tiempos que están definidos en cada uno de los dominios.

Ejercicio 1.7

Explica las ventajas y desventajas que puede suponer tener el TTL de 5 minutos o 10 horas.

El comando dig

El comando dig te permite realizar consultas sobre servidores de DNS, pero a diferencia de nslookup solo funciona para linux (En Windows no existe por defecto y habría que instalarlo).

Ejemplo

\$ dig profesordeinformatica.com

; <<>> DiG 9.9.2-P1 <<>> profesordeinformatica.com

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32615

;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:

; EDNS: version: 0, flags:; udp: 4000

;; QUESTION SECTION:

;profesordeinformatica.com. IN A

;; ANSWER SECTION:

profesordeinformatica.com. 84482 IN A 217.160.242.178

profesordeinformatica.com

```
;; Query time: 50 msec
;; SERVER: 127.0.1.1#53(127.0.1.1)
;; WHEN: Mon Sep 16 22:57:13 2013
;; MSG SIZE rcvd: 70
$ dig @8.8.8.8 profesordeinformatica.com
```

Hacemos la petición sobre el servidor de DNS de Google.

```
$ dig MX @8.8.8.8 profesordeinformatica.com
```

Obtenemos información sobre el tipo MX

```
$ dig ANY @8.8.8.8 profesordeinformatica.com
```

Obtenemos toda la información

```
$ dig +trace profesordeinformatica.com
```

Te hace una traza por los servidores que pasa hasta llegar al autoritario.

```
$ dig -x 8.8.8.8
```

Te hace una resolución inversa

Configuración clientes TCP/IP

Básicamente para realizar una configuración de un cliente lo podemos hacer mediante un servidor de **DHCP**, que nos proporciona directamente los datos de configuración de la red o realizarla **manualmente**. Si la realizamos manualmente vamos a necesitar 4 datos.

1. La dirección IP: Que que identifica al equipo de forma única en la red.
2. Máscara de subred: Que nos permite crear subredes.
3. Puerta de enlace: Nos permite enlazar la conexión de nuestro ordenador con otras redes como Internet.
4. Servidor DNS: Es el que te resuelve los nombres en direcciones IP y viceversa.

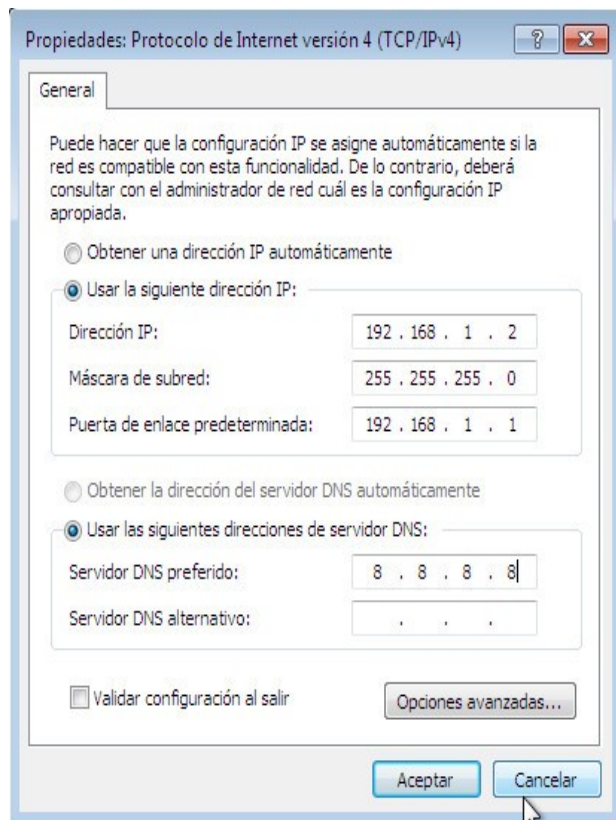
1.- Configuración Red Windows

En Windows XP

- Conexiones de red, hacemos clic en el botón Inicio y pinchamos en el Panel de control.
- En el cuadro de búsqueda, escriba adaptador y, en Centro de redes y recursos compartidos, haga clic en Ver conexiones de red.
- Hacemos clic con el botón derecho en la conexión que queremos cambiar. Pinchamos en Propiedades.
- Hacemos clic en la pestaña funciones de red. En Esta conexión usa los siguientes elementos, hacemos clic en Protocolo de Internet versión 4 (TCP/IPv4) o en Protocolo de Internet versión 6 (TCP/IPv6) y, después, en Propiedades.

En Windows 7

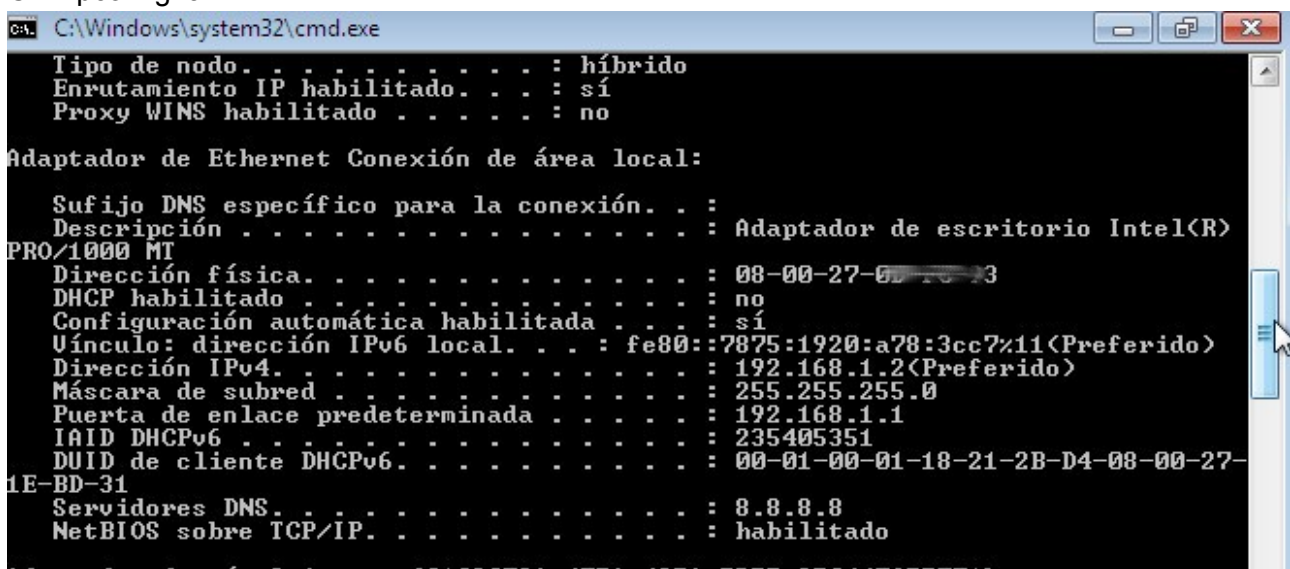
Panel de control → Redes e Internet → Centro de Redes y Recursos Compartidos → Conexión de Área local → Propiedades → Protocolo de Internet versión 4 (TCP/IPv4) o Protocolo de Internet versión 6 (TCP/IPv6) y propiedades.



Si elegimos Obtener una dirección IP automáticamente estamos solicitando dicha dirección al servidor DHCP. Si elegimos obtener la dirección del servidor DNS automáticamente la estamos solicitando al servidor DHCP.

Podemos ver el resultado desde la línea de comandos:

C:/> ipconfig /all



2.- Configuración red Ubuntu

- Pinchamos en el icono de red en la parte superior derecha.
- Pinchamos en "Editar conexiones".
- Seleccionamos la conexión y pinchamos "Editar".
- Seleccionamos la pestaña "Ajustes de IPv4" y rellenamos los datos tras pinchar "Añadir":

Editando Conexión cableada 1

Nombre de la conexión:

General Cableada Seguridad 802.1x **Ajustes de IPv4** Ajustes de IPv6

Método:

Dirección

Dirección	Máscara de red	Puerta de enlace
192.168.1.3	24	192.168.1.1

Servidores DNS:

Dominios de búsqueda:

Los dominios que usar al resolver los nombres de equipos. Use comas para múltiples.

☐ Requiere dirección IPv4 para que esta conexión se complete

Podemos ver la configuración con los siguientes comandos:

- IP y máscara de subred

```
$ ifconfig
```

- Puerta de enlace

```
$ route
```

- DNS

```
$ cat /etc/resolv.conf
```

3.- Configuración red Manual Linux

- Editamos el fichero /etc/network/interfaces

```
# Contenido /etc/network/interfaces
```

```
auto lo
```

```
iface lo inet loopback auto eth0
```

```
# iface eth0 inet dhcp
iface eth0 inet static
address 192.168.1.40
netmask 255.255.255.0
network 192.168.1.0
broadcast 192.168.1.255
gateway 192.168.1.1
```

- Editamos las DNS en el fichero /etc/resolv.conf

```
# Contenido resolv.conf
nameserver 8.8.8.8
```

- La puerta de enlace se puede configurar con el comando route

```
route add default gw 192.168.1.1 eth0
```

- reiniciamos el servidor.

```
# /etc/init.d/networking restart
```

Podemos cambiar la IP de forma temporal mediante el comando ifconfig. Pero una vez que reiniciemos la red o el equipo la ip vuelve a su estado inicial:

```
ifconfig eth0 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
```

4.- Configuración red Manual Ubuntu Network Manager

Ubuntu para gestionar su red utiliza Network Manager en lugar Network admin.

Editamos el fichero **/etc/NetworkManager/system-connections/Conexión cableada 1**

Donde Conexión cableada es el nombre de nuestra conexión. Tenemos una conexión por cada interface de red.

Este fichero tiene el siguiente contenido:

```
[802-3-ethernet]
duplex=full
mac-address=**:04:A6:03:**:58

[connection]
id=Conexión cableada 1
uuid=cd74f119-6317-4ec3-a506-****738cedf
type=802-3-ethernet
timestamp=1347346126

[ipv6]
method=auto
```

```
[ipv4]
method=manual
dns=8.8.8.8;
addresses1=192.168.1.10;24;192.168.1.1;
```

Modificamos las dos últimas líneas para asignarle las dns, ip, máscara de subred o puerta de enlace respectivamente.

Reiniciamos el network manager:

```
#/etc/init.d/network-manager restart
```

El fichero /etc/hosts

El fichero /etc/host de linux y su equivalente en Windows c:\WINDOWS\system32\drivers\etc\hosts nos permite realizar una correspondencia de nombres de dominio y direcciones IP en un equipo local antes de que nos resuelva el servidor. Podemos utilizar el fichero para lo siguiente:

1. Algunos hackers utilizan esta técnica para que redireccione a su página web simplemente entrando en webs como Google. En el ejemplo entrando desde nuestro ordenador a www.google.com le redirecciona a la página de elmundo.es.
2. Cuando estamos desarrollando localmente nuestro proyecto y queremos utilizar el dominio real.
3. Cuando queremos cambiar nuestro proyecto de servidor podemos poner la nueva dirección IP.

Este es el ejemplo del contenido de el fichero:

```
127.0.0.1      localhost
127.0.1.1      dinux
193.110.128.199 www.google.es
127.0.0.1      profesordeinformatica.com www.profesordeinformatica.com
```

```
# The following lines are desirable for IPv6 capable hosts
```

```
::1    ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

DNS Servidor Linux con resolución directa

Vamos a crear el registro de zonas para el dominio **profesordeinformatica.com**

1.- Instalamos bind9
apt-get update

profesordeinformatica.com

apt-get install bind9

2.-Editamos el fichero **/etc/bind/named.conf.local** y añadimos la siguiente líneas:

```
zone "profesordeinformatica.com" {  
type master;  
file"/etc/bind/db.profesordeinformatica";  
};
```

Nota: Te puedes ayudar del archivo /etc/bind/named.conf-default.zones.

3.-Editamos el fichero **/etc/bind/db.profesordeinformatica**

```
;  
; BIND data file for local loopback interface  
;  
$TTL 38400  
profesordeinformatica.com. IN      SOA  servidor.profesordeinformatica.com. aitor.kaixo.com.  
    (2; Serial  
      604800; Refresh  
      86400; Retry  
      2419200; Expire  
      38400); Negative Cache TTL  
  
profesordeinformatica.com.      IN      NS      servidor.profesordeinformatica.com.  
profesordeinformatica.com.      IN      A        172.20.224.115  
servidor.profesordeinformatica.com. IN    A        172.20.224.115  
p01.profesordeinformatica.com.   IN      A        172.20.224.101  
p02.profesordeinformatica.com.   IN      A        172.20.224.102  
p03                              IN      A        172.20.224.103  
p04.profesordeinformatica.com.   IN      A        172.20.224.104  
alumno2                          IN      CNAME     p02  
alumno3                          IN      CNAME     p03.profesordeinformatica.com.  
profesordeinformatica.com.      IN      MX       1      p03.profesordeinformatica.com.  
profesordeinformatica.com.      IN      MX       10     p02.profesordeinformatica.com.
```

Nota: te puedes ayudar del archivo /etc/bind/db.local

4.- Reiniciamos el servidor

```
/etc/init.d/bind9 restart
```

5.- Realizamos pruebas con el comando nslookup

```
# nslookup – localhost  
> set type=MX  
> profesordeinformatica.com
```

profesordeinformatica.com

Server: 172.20.224.115
Address: 172.20.224.115#53

profesordeinformatica.com mail exchanger = 10 p02.profesordeinformatica.com.
profesordeinformatica.com mail exchanger = 1 profesordeinformatica.com.

Resolución Inversa Linux

Se trata de que dado una dirección ip (172.20.224.101) nos devuelva un nombre (p01.profesordeinformatica.com.)

1.- Editamos el fichero **/etc/bind/named.conf.local** y añadimos la siguiente líneas:

```
zone "20.172.in-addr.arpa" {  
    type master;  
    file "/etc/bind/db.172.20";  
};
```

Nota: te puedes ayudar del archivo /etc/bind/named.conf-default.zones

2.- Editamos el fichero **/etc/bind/db.172.20**

```
;  
; BIND reverse data file for local loopback interface  
;  
$TTL 604800  
20.172.in-addr.arpa. IN SOA servidor.profesordeinformatica.com. aitor.profesordeinformatica.com.  
(  
    1; Serial  
    604800; Refresh  
    86400; Retry  
    2419200; Expire  
    604800); Negative Cache TTL  
;  
20.172.in-addr.arpa. IN NS servidor.profesordeinformatica.com.  
  
115.224 IN PTR servidor.profesordeinformatica.com.  
101.224 IN PTR p01.profesordeinformatica.com.  
102.224.20.172.in-addr.arpa. IN PTR p02.aula202.com.
```

Nota: te puedes ayudar del archivo /etc/bind/db.local

Notas: Se puede en lugar de "20.172.in-addr.arpa." se puede utilizar @. Si utilizamos la dirección completa tiene que terminar en "."

3.- Reiniciamos el servidor

/etc/init.d/bind9 restart

profesordeinformatica.com

4.- Realizamos pruebas con el comando nslookup

```
nslookup - localhost  
> 172.20.202.101  
Server: 172.20.202.115  
Address: 172.20.202.115#53
```

101.202.20.172.in-addr.arpa name = p01.profesordeinformatica.com.

Validar configuración bind. Comandos named-checkconf y named-checkzone

Los comandos **named-checkconf** y **named-checkzone** nos pueden ayudar a encontrar los errores en los ficheros de configuración del servicio de nombres.

named-checkconf

Te suele informar de posibles errores en el fichero /etc/bind/named.conf.local. No tiene parámetros.

\$named-checkconf

named-checkzone

Nos suele ayudar a encontrar errores en el fichero de registro de recursos. Tiene dos parámetros el dominio y el fichero:

Ejemplo resolución directa:

\$named-checkzone aula202.com /etc/bind/db.aula202

Ejemplo resolución inversa:

\$named-checkzone 20.172.in-addr.arpa /etc/bind/db.172.20

Webmin

Webmin es un software que te permite configurar los servidores y servicios de estos utilizando

<http://www.webmin.com/deb.html>

1.- Instalamos las dependencias del webmin

```
apt-get install perl libnet-ssleay-perl openssl libauthen-pam-perl libpam-runtime libio-pty-perl  
apt-show-versions python
```

2.- Descargamos la última versión del webmin

wget http://prdownloads.sourceforge.net/webadmin/webmin_1.650_all.deb

profesordeinformatica.com

3.- Instalamos el webmin
dpkg -i [webmin_1.650_all.deb](#)

4.- Accedemos a
<https://direccionip:10000> o <https://localhost:10000>

Hay que aceptar e importar el certificado digital para poder acceder a Webmin.

5.- En el lateral izquierdo dentro de Servidores, seleccionamos el servidor BIND DNS. Vamos a ver en Zonas Existentes, las zonas por defecto que crea BIND así como las creadas por nosotros anteriormente.


Configuración de Módulo


Servidor de DNS BIND


Versión 9.9.2 de BIND


Apply Configuration
Stop BIND
Buscar Documentos..


Opciones Globales del Servidor


 Otros Servidores DNS


 Bitácora y Errores


 Listas de Control de Acceso


 Archivos y Directorios


 Reenvío y Transferencias


 Direcciones y Topología


 Opciones Varias


 Opciones de Interfase de Control


 Claves DNS


 Valores por Defecto de Zona


 Cluster Slave Servers

 Setup RNDNC

 DNSSEC Verification


 DNSSEC Key Re-Signing


 Check BIND Config


 Edit Config File


Zonas DNS Existentes


Seleccionar todo. | Invertir selección. | Crar una nueva zona maestra | Crear una nueva zona subordinada | Crear una nueva zona de sólo caché | Crear una nueva zona de reenvío | Crear zona de delegación. | Crear zonas desde archivo de lotes.

 ☐ Zona raíz

 ☐ 0

 ☐ 127

 ☐ 255

 ☐ localhost

Seleccionar todo. | Invertir selección. | Crar una nueva zona maestra | Crear una nueva zona subordinada | Crear una nueva zona de sólo caché | Crear una nueva zona de reenvío | Crear zona de delegación. | Crear zonas desde archivo de lotes.

6.- Pinchamos en Crear zona maestra e introducimos el nombre del dominio, servidor maestro, dirección de correo electrónico del maestro así como los tiempos del registro SOA.

Índice de Módulo

Crear Zona Maestra

Apply Configuration
Stop BIND

Opciones de nueva zona maestra

Tipo de zona ☒ Reenvío (Nombres a Direcciones) ☐ Inversas (Direcciones a Nombres)

Nombre de Dominio/Red

Archivo de Registros ☒ Automático ☐

Servidor Maestro ☒ ¿Añadir registro NS para servidor maestro?

Dirección de correo

¿Utilizar plantilla de zona? ☐ Si ☒ No Dirección IP para registros de plantilla

Add reverses for template addresses? ☒ Si ☐ No

Tiempo de refresco segundos

Tiempo de reintento de transferencia segundos

Tiempo de expiración segundos

Tiempo-que-está-viva por Defecto segundos

[Regresar a lista de zonas](#)












7.- Podemos añadir diferentes registros de recursos. Dirección (A), Servidor de Nombres (NS), Alias de nombres (NS) o Servidores de correo (MX).







Indice de Módulo

Editar Zona Maestra

aula224.com

[Apply Zone](#)
[Apply Configuration](#)
[Stop BIND](#)

 Dirección (0)	 Servidor de Nombres (1)	 Alias de Nombre (0)	 Servidor de Correo (0)
 Información de Máquina (0)	 Texto (0)	 Sender Permitted From (0)	 Servicio Acreditado (0)
 Persona Responsable (0)	 Dirección Inversa (0)	 Localización (0)	 Dirección del servicio (0)
 Clave pública (0)	 Dirección IPv6 (0)	 Todos los Tipos de Registro (1)	

 Editar Archivo de Registros	 Editar Parámetros de Zona	 Editar Opciones de Zona	 Buscar IPs Libres
 Generadores de Registro	 Mirar Información de WHOIS	 Setup DNSSEC Key	

[Freeze Zone](#) Click this button to freeze a dynamic zone before updating it.

8.- Añadimos un registro de dirección para el equipo p01.

Indice de Módulo

Dirección Registros

En aula224.com

[Apply Zone](#)
[Apply Configuration](#)
[Stop BIND](#)

Añadir Registro Dirección

Nombre Tiempo de vida ☒ Por defecto ☐ segundos

Dirección ...

¿Actualizar Inversas? ☒ Si ☐ Sí (y reemplazar las existentes) ☐ No

[Crear](#)

[Regresar a lista de zonas](#) | [Regresar a tipos de registro](#)

Se la misma forma vamos a crear registros de dirección para el servidor (servidor.aula224.com), equipo p02, p03, etc. Vamos a crear registros MX para el dominio, etc.

El resultado lo guarda en /var/lib/bind/aula224.com.hosts"

```
$ttl 38400
aula224.com. IN SOA server.aula224.com. aitor.kaixo.com. (
1380598158
10800
3600
```

profesordeinformatica.com

```

604800
38400 )
aula224.com. IN NS server.aula224.com.
p01.aula224.com. IN A 172.20.202.104
server.aula224.com. IN A 172.20.202.104

```

En el lateral izquierdo podemos aplicamos la configuración, paramos BIND e iniciamos BIND. Podemos hacer un nslookup y probar el resultado.

Resolución Inversa con Webmin

Pinchamos en Crear zona maestra y seleccionamos el tipo de zona **Inversas**.

Indice de Módulo Apply Configuration
Stop BIND

Crear Zona Maestra

Opciones de nueva zona maestra

Tipo de zona ☐ Reenvío (Nombres a Direcciones) ☒ Inversas (Direcciones a Nombres)

Nombre de Dominio/Red

Archivo de Registros ☒ Automático ☐

Servidor Maestro ☒ ¿Añadir registro NS para servidor maestro?

Dirección de correo

¿Utilizar plantilla de zona? ☐ Si ☒ No Dirección IP para registros de plantilla

Add reverses for template addresses? ☒ Si ☐ No

Tiempo de refresco segundos

Tiempo de reintento de transferencia segundos

Tiempo de expiración segundos

Tiempo-que-está-viva por Defecto segundos

[Regresar a lista de zonas](#)

Vamos a crear registros PTR o de Resolución Inversa.

Indice de Módulo Apply Zone
Apply Configuration
Stop BIND

Dirección Inversa Registros

En 172.20

Añadir Registro Dirección Inversa

Dirección Tiempo de vida ☒ Por defecto ☐

Máquina

¿Actualizar las de Reenvío? ☒ Si ☐ No

Seleccionar todo. | Invertir selección.

Dirección	TTL	Máquina
<input type="checkbox"/> 172.20.224.115	Por defecto	aula224.com.

Seleccionar todo. | Invertir selección.

[Regresar a lista de zonas](#) | [Regresar a tipos de registro](#)

Servidor secundario

Para crear un servidor secundario lo único que tenemos que hacer es añadir en el fichero named.conf.local la zona indicando que va a ser de tipo secundario (slave) y el servidor primario

profesordeinformatica.com

donde vamos a contener información de la zona.

```
zone "aula224.com" {  
    type slave;  
    masters  
        {  
            172.20.224.115;  
        };  
};
```

En webmin habría que pinchar en "Crear zona subordinada".

Indice de Módulo

Crear Zona Subordinada

Apply Configuration
Stop BIND

Opciones de nueva zona subordinada

Tipo de Zona ☒ Reenvío (Nombres a Direcciones) ☐ Inversas (Direcciones a Nombres)

Nombre de Dominio/Red

Archivo de Registros ☐ Ninguno ☒ Automático

Servidores Maestros

Puerto de Servidor ☒ Por defecto ☐ puerto

Crear

Regresar a lista de zonas

DNS Microsoft Windows Server 2008

Una vez instalado el sistema operativo, es aconsejable que un servidor de DNS tenga un ip fija. Realizamos la instalación del servicio DNS:

Inicio → Herramientas administrativas → Administrador del servidor → Agregar funciones → Marcamos la casilla Servidor DNS

Antes de comenzar

Funciones de servidor

Servidor DNS

Confirmación

Progreso

Resultado

Seleccione una o más funciones para instalar en este servidor.

Funciones:

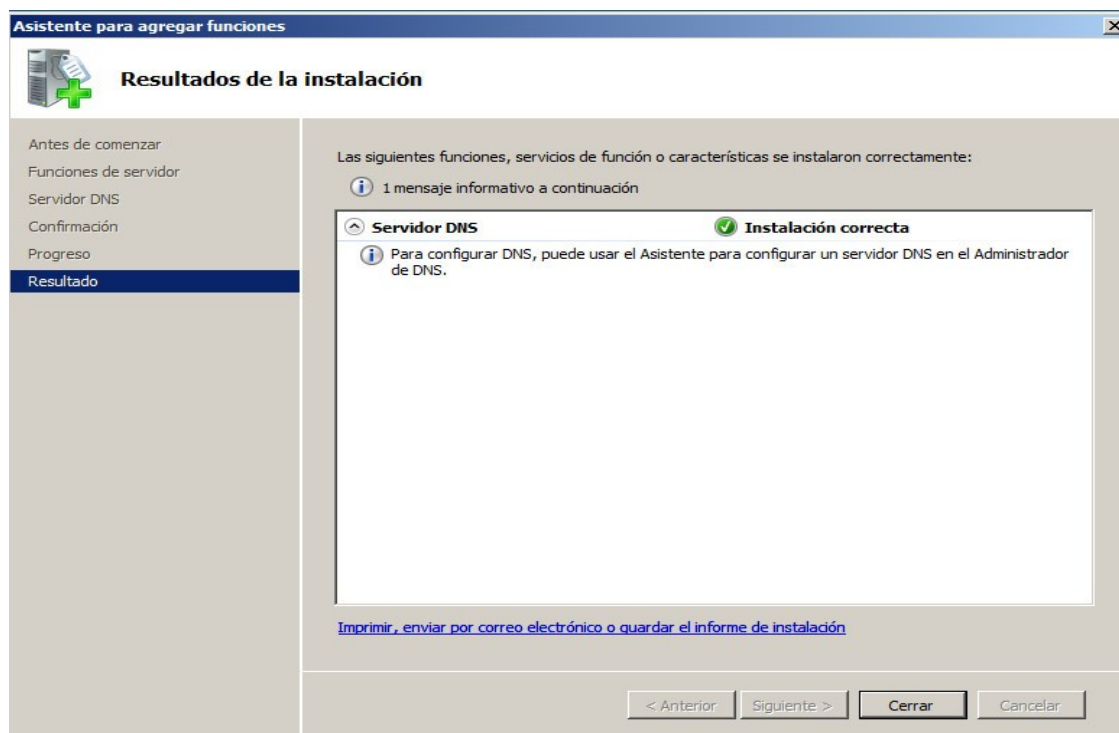
- ☐ Active Directory Rights Management Services
- ☐ Servicios de acceso y directivas de redes
- ☐ Servicios de archivo
- ☐ Servicios de Certificate Server de Active Directory
- ☐ Servicios de directorio ligero de Active Directory
- ☐ Servicios de dominio de Active Directory
- ☐ Servicios de federación de Active Directory
- ☐ Servicios de implementación de Windows (WDS)
- ☐ Servicios de impresión
- ☐ Servicios UDDI
- ☐ Servidor de aplicaciones
- ☐ Servidor de fax
- ☐ Servidor DHCP
- ☒ **Servidor DNS**
- ☐ Servidor web (IIS)
- ☐ Terminal Services
- ☐ Windows Server Update Services

Más información acerca de las funciones de servidor

Descripción:
[Servidor DNS \(Sistema de nombres de dominio\)](#) proporciona resolución de nombres en redes TCP/IP. El servidor DNS se administra con mayor facilidad cuando se instala en el mismo servidor que los Servicios de dominio de Active Directory. Si selecciona la función Servicios de dominio de Active Directory, puede instalar y configurar el servidor DNS y los Servicios de dominio de Active Directory para que funcionen conjuntamente.

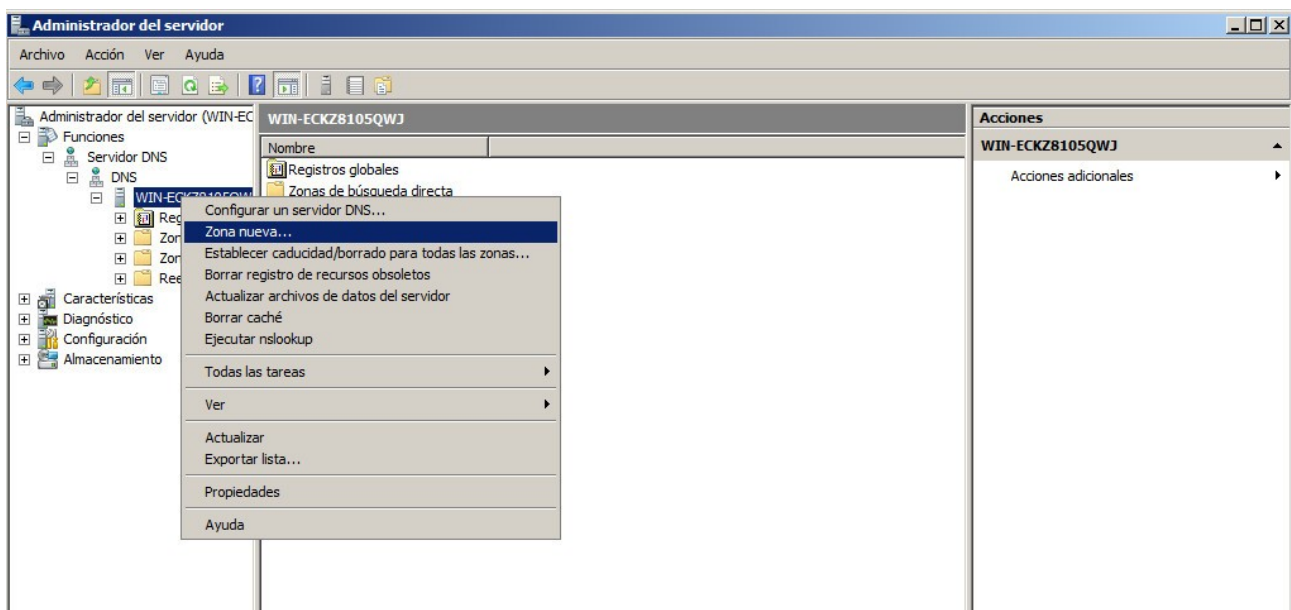
< Anterior Siguiente > Instalar Cancelar

Pinchamos **Siguiente** → **Siguiente** → **Instalar**



Una vez instalado pinchamos en Cerrar. Muchas veces es recomendable cerrar y volver a abrir el administrador del servidor.

Vemos que dentro del administrador del servidor se nos ha creado un arbol con el servicio DNS instalado. Vamos hasta el servidor DNS (WIN-ECK,...)



Podemos realizar una prueba con el comando nslookup – localhost y probando con el dominio localhost.

Resolución directa con Windows 2008

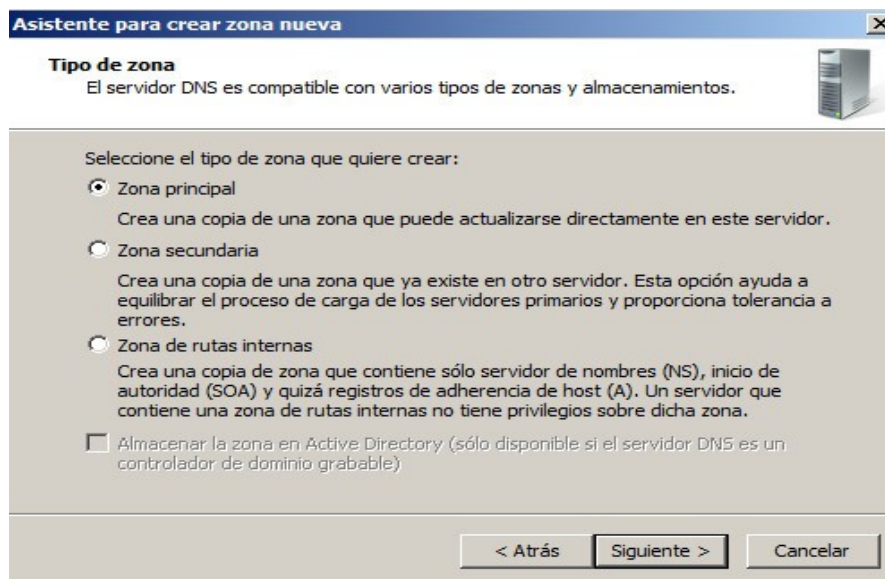
Pinchamos dentro del menú en Acción y Zona nueva (También se puede hacer con el botón derecho). Nos arranca un Asistente. Pinchamos siguiente para continuar.

zona principal: servidor primario.

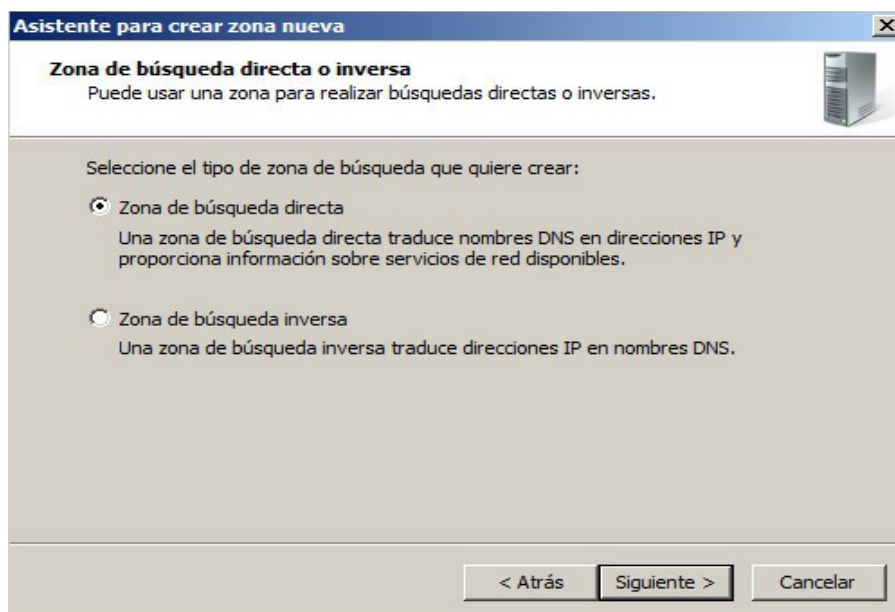
zona secundaria: servidor secundarios.

zona de rutas internas: Servidor no autoritario.

En nuestro caso creamos un servidor primario



Zona directa o zona inversa. En nuestro caso directa.



Elegimos la zona: aula224.com

Asistente para crear zona nueva

Nombre de zona
¿Qué nombre tiene la zona nueva?

El nombre de zona especifica la parte del espacio de nombres DNS para el que actúa el servidor de autorización. Puede ser el nombre de dominio de la organización (por ejemplo, microsoft.com) o una parte del nombre de dominio (por ejemplo, nuevazona.microsoft.com). El nombre de zona no es el nombre del servidor DNS.

Nombre de zona:
aula224.com

< Atrás Siguiendo > Cancelar

Elegimos que nos cree el fichero donde se va a guardar la zona: aula224.com.dns

Asistente para crear zona nueva

Archivo de zona
Puede crear un archivo de zona nuevo o usar un archivo copiado de otro servidor DNS.

¿Desea crear un archivo nuevo de zona o usar el archivo existente que copió de otro servidor DNS?

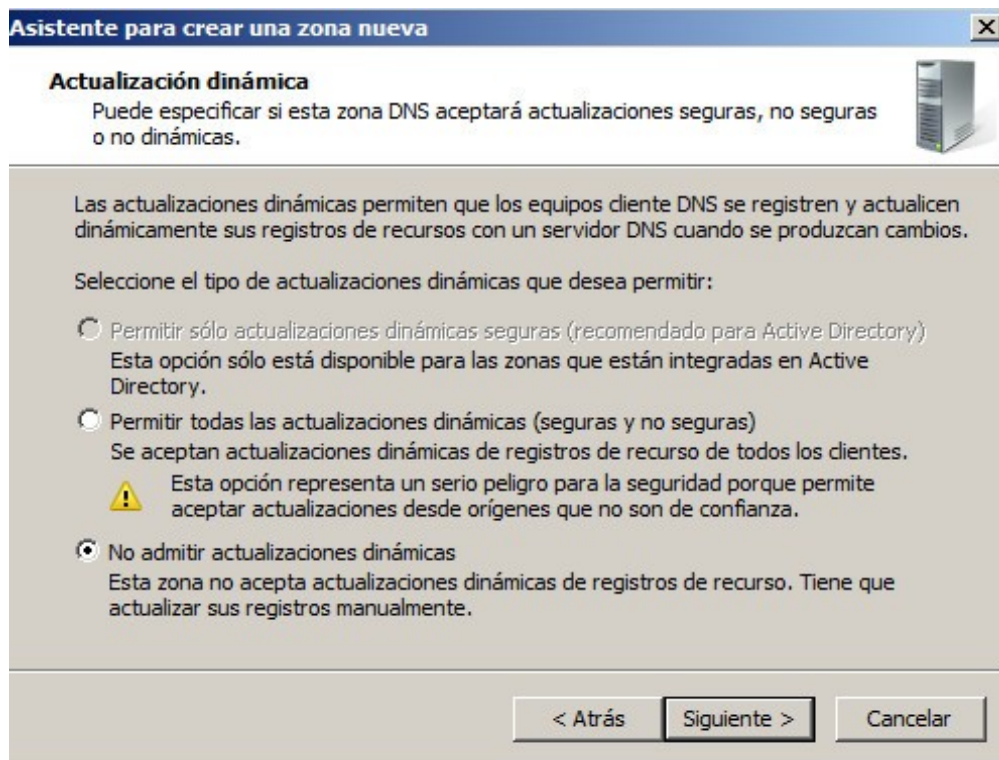
☒ Crear un archivo nuevo con este nombre de archivo:
aula224.com.dns

☐ Usar este archivo:

Para usar este archivo existente, asegúrese primero de que se ha copiado en la carpeta %SystemRoot%\system32\dns en este servidor y haga luego clic en Siguiendo.

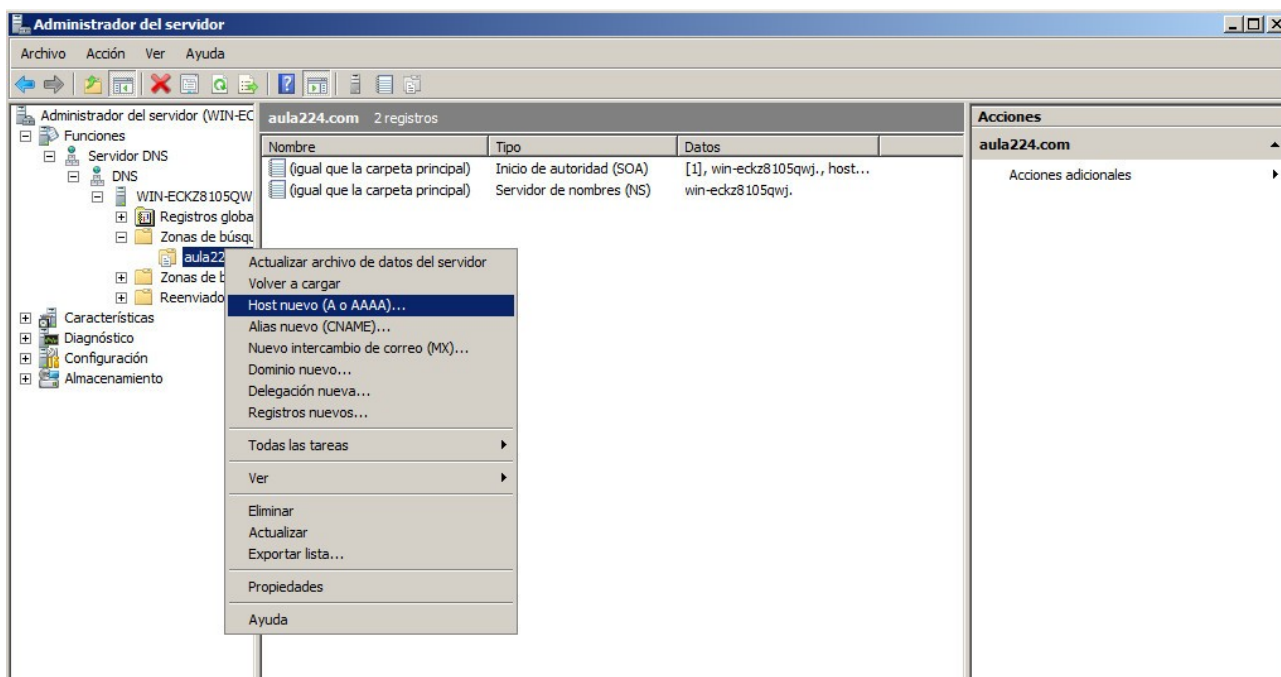
< Atrás Siguiendo > Cancelar

Seleccionamos por defecto que no nos permita actualizaciones dinámicas. Se actualizan los registros manualmente.

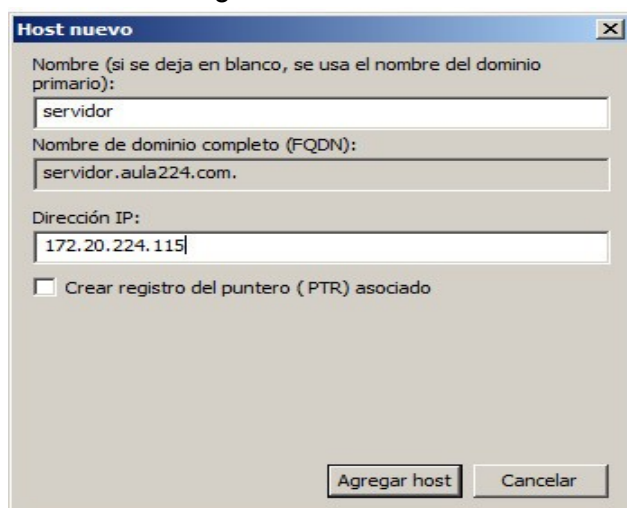


Pinchamos en Siguiendo y Finalizar.

Vemos que en las zonas directas se nos ha creado el registro de zonas de búsqueda directa con los registros de recursos SOA y NS. Podemos añadir nuestros registros A, AAAA, CNAME o MX desde el botón derecho o acción.



Añadimos un registro A:



Host nuevo

Nombre (si se deja en blanco, se usa el nombre del dominio primario):
servidor

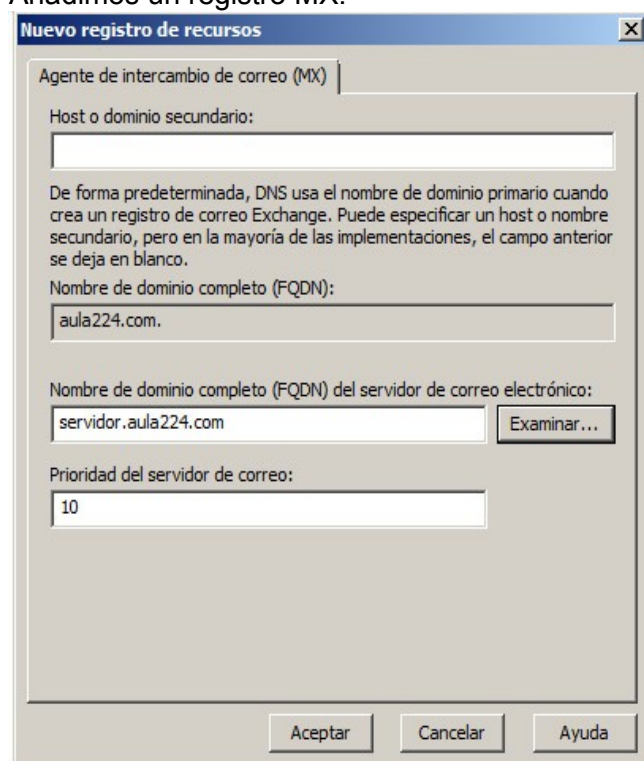
Nombre de dominio completo (FQDN):
servidor.aula224.com.

Dirección IP:
172.20.224.115

☐ Crear registro del puntero (PTR) asociado

Agregar host Cancelar

Añadimos un registro MX:



Nuevo registro de recursos

Agente de intercambio de correo (MX)

Host o dominio secundario:

De forma predeterminada, DNS usa el nombre de dominio primario cuando crea un registro de correo Exchange. Puede especificar un host o nombre secundario, pero en la mayoría de las implementaciones, el campo anterior se deja en blanco.

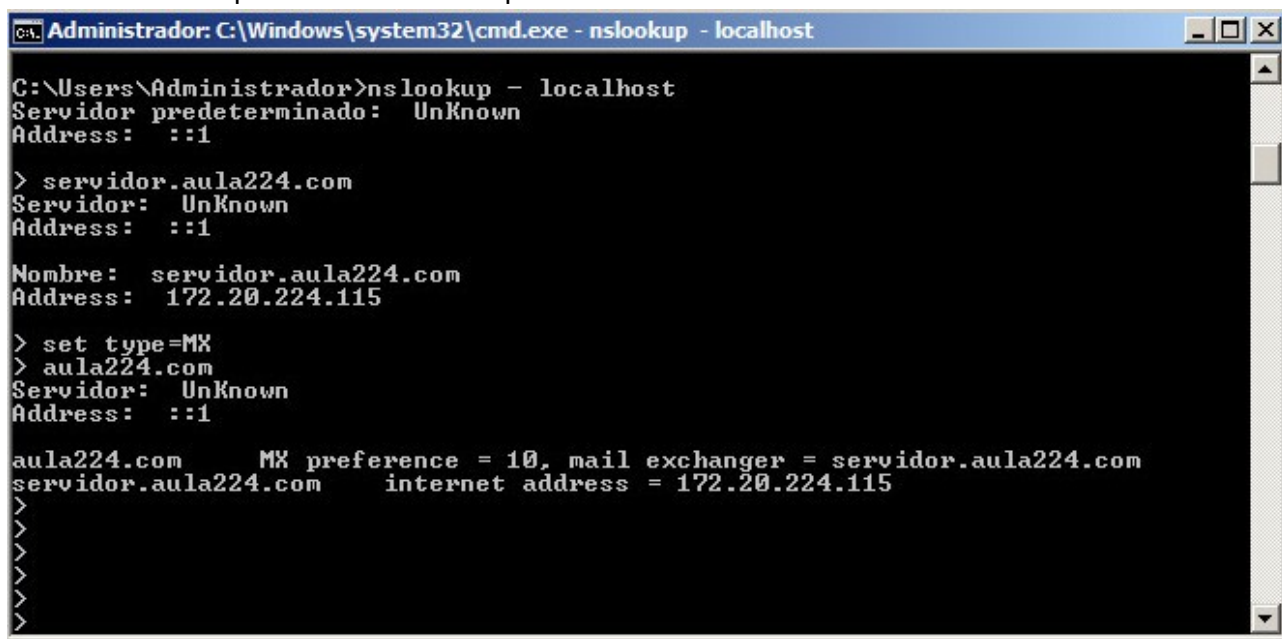
Nombre de dominio completo (FQDN):
aula224.com.

Nombre de dominio completo (FQDN) del servidor de correo electrónico:
servidor.aula224.com Examinar...

Prioridad del servidor de correo:
10

Aceptar Cancelar Ayuda

Hacemos pruebas con nslookup:



```
C:\Users\Administrador>nslookup - localhost
Servidor predeterminado: UnKnown
Address: ::1

> servidor.aula224.com
Servidor: UnKnown
Address: ::1

Nombre: servidor.aula224.com
Address: 172.20.224.115

> set type=MX
> aula224.com
Servidor: UnKnown
Address: ::1

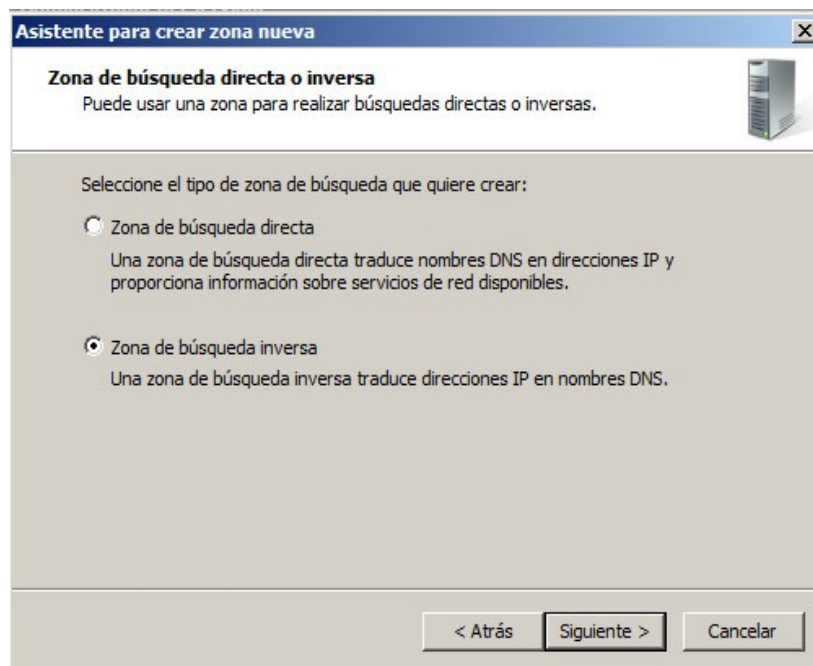
aula224.com      MX preference = 10, mail exchanger = servidor.aula224.com
servidor.aula224.com  internet address = 172.20.224.115
>
>
>
>
>
```

Resolución inversa con Windows 2008

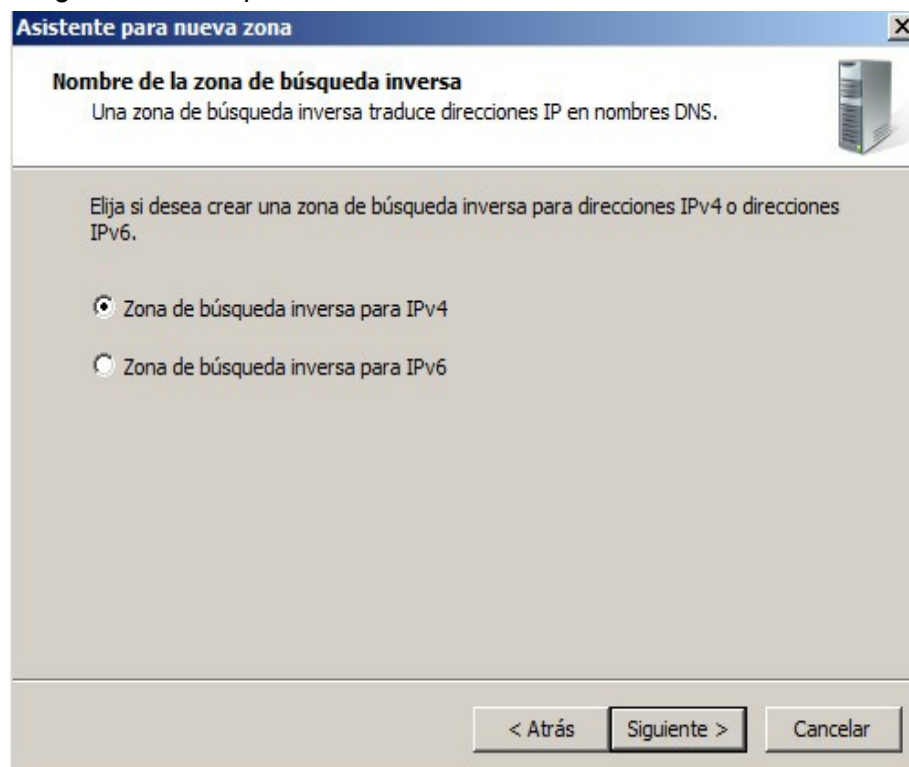
Pinchamos dentro del menú en Acción y Zona nueva (También se puede hacer con el botón derecho). Nos arranca un Asistente. Pinchamos siguiente para continuar.

Elegimos **zona principal**: servidor primario.

Elegimos **zona de búsqueda inversa**:



Elegimos la zona para IPv4:



Asistente para nueva zona

Nombre de la zona de búsqueda inversa
Una zona de búsqueda inversa traduce direcciones IP en nombres DNS.

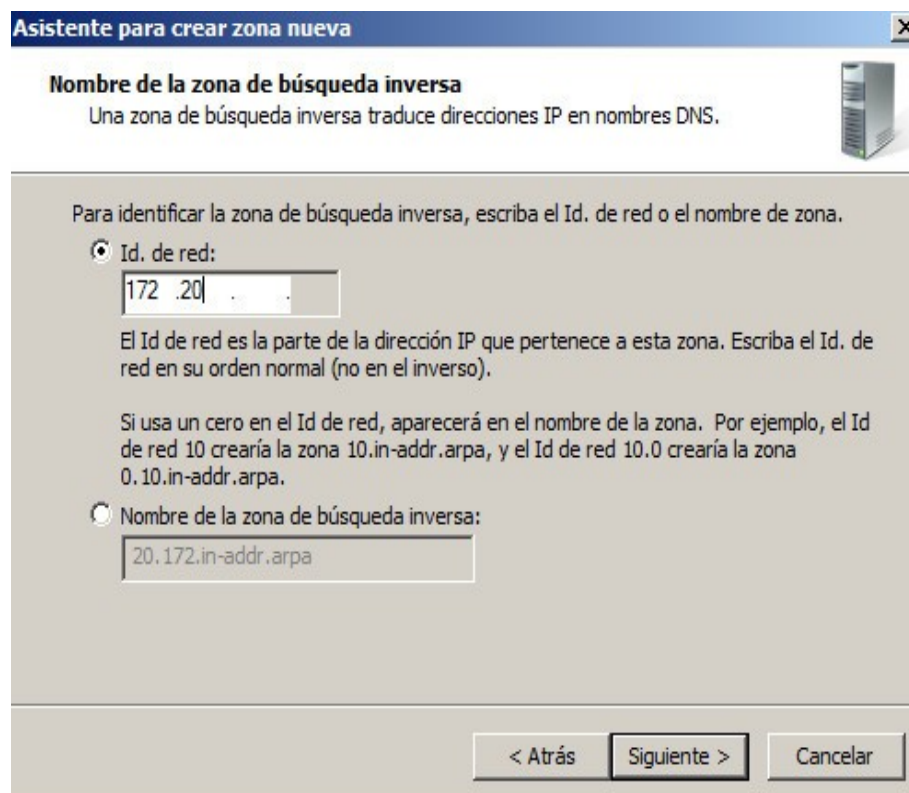
Elija si desea crear una zona de búsqueda inversa para direcciones IPv4 o direcciones IPv6.

☒ Zona de búsqueda inversa para IPv4

☐ Zona de búsqueda inversa para IPv6

< Atrás Siguiendo > Cancelar

Elegimos la red. En nuestro caso como se trata de una red de tipo B 172.20:



Asistente para crear zona nueva

Nombre de la zona de búsqueda inversa
Una zona de búsqueda inversa traduce direcciones IP en nombres DNS.

Para identificar la zona de búsqueda inversa, escriba el Id. de red o el nombre de zona.

☒ Id. de red:

172.20

El Id de red es la parte de la dirección IP que pertenece a esta zona. Escriba el Id. de red en su orden normal (no en el inverso).

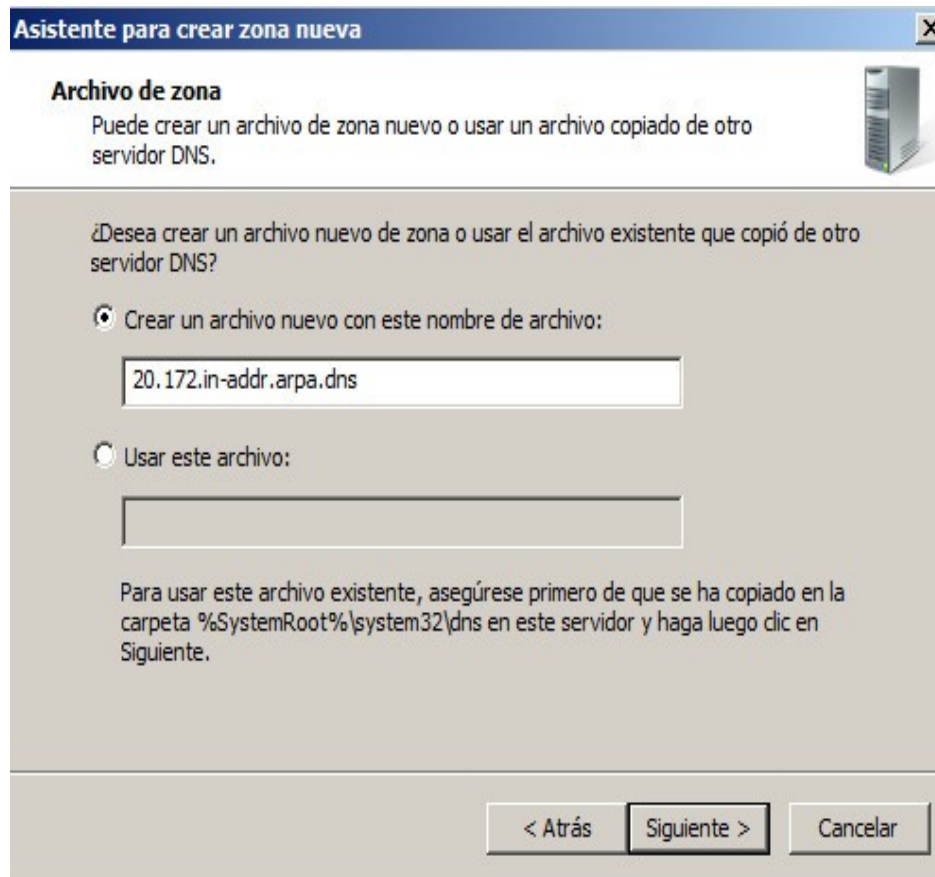
Si usa un cero en el Id de red, aparecerá en el nombre de la zona. Por ejemplo, el Id de red 10 crearía la zona 10.in-addr.arpa, y el Id de red 10.0 crearía la zona 0.10.in-addr.arpa.

☐ Nombre de la zona de búsqueda inversa:

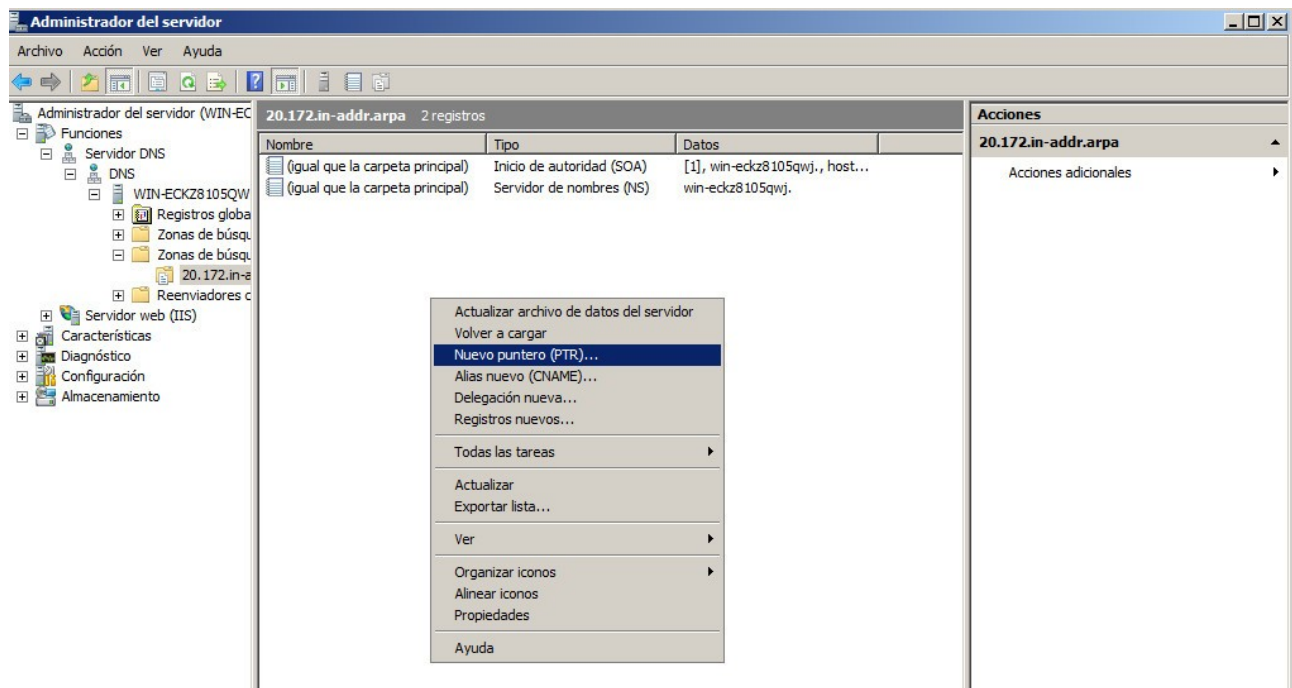
20.172.in-addr.arpa

< Atrás Siguiendo > Cancelar

Elegimos el archivo donde se va a guardar los registros de recursos de la zona inversa. En nuestro caso



Ya hemos creado la zona inversa. Ya solo nos falta crear los registros PTR o CNAME:



Creamos el registro PTR de resolución inversa:

Nuevo registro de recursos

Puntero (PTR)

Dirección IP del host:
172.20.224.115

Nombre de dominio completo (FQDN):
115.224.20.172.in-addr.arpa

Nombre de host:
servidor.aula224.com Examinar...

Aceptar Cancelar

Realizamos las pruebas con el comando nslookup:

```
Administrador: C:\Windows\system32\cmd.exe - nslookup - localhost
Microsoft Windows [Versión 6.0.6002]
Copyright (c) 2006 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Administrador>nslookup - localhost
Servidor predeterminado: UnKnown
Address: ::1

> servidor.aula224.com
Servidor: UnKnown
Address: ::1

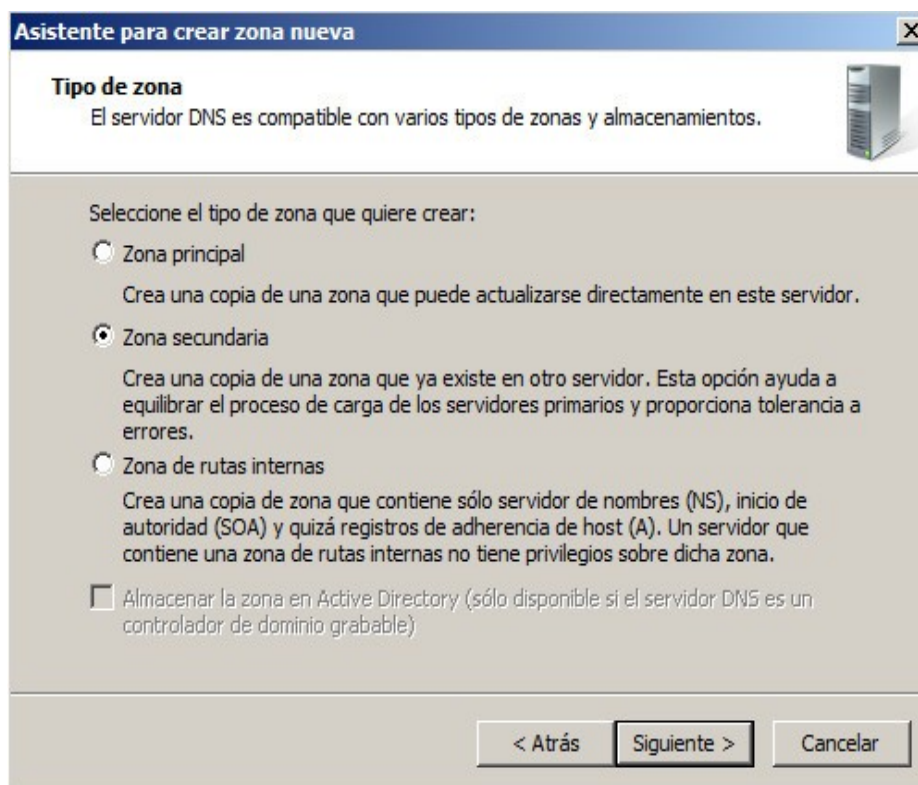
Nombre: servidor.aula224.com
Address: 172.20.224.115

>
```

Servidor secundario con Windows 2008

En nuestro caso podemos montar otro servidor en cualquier puesto del alumno. La zona aula224.com y nuestro servidor primario puede ser del profesor 172.20.224.115.

Creamos una nueva zona de tipo secundaria:



Asistente para crear zona nueva

Tipo de zona
El servidor DNS es compatible con varios tipos de zonas y almacenamientos.

Seleccione el tipo de zona que quiere crear:

☐ Zona principal
Crea una copia de una zona que puede actualizarse directamente en este servidor.

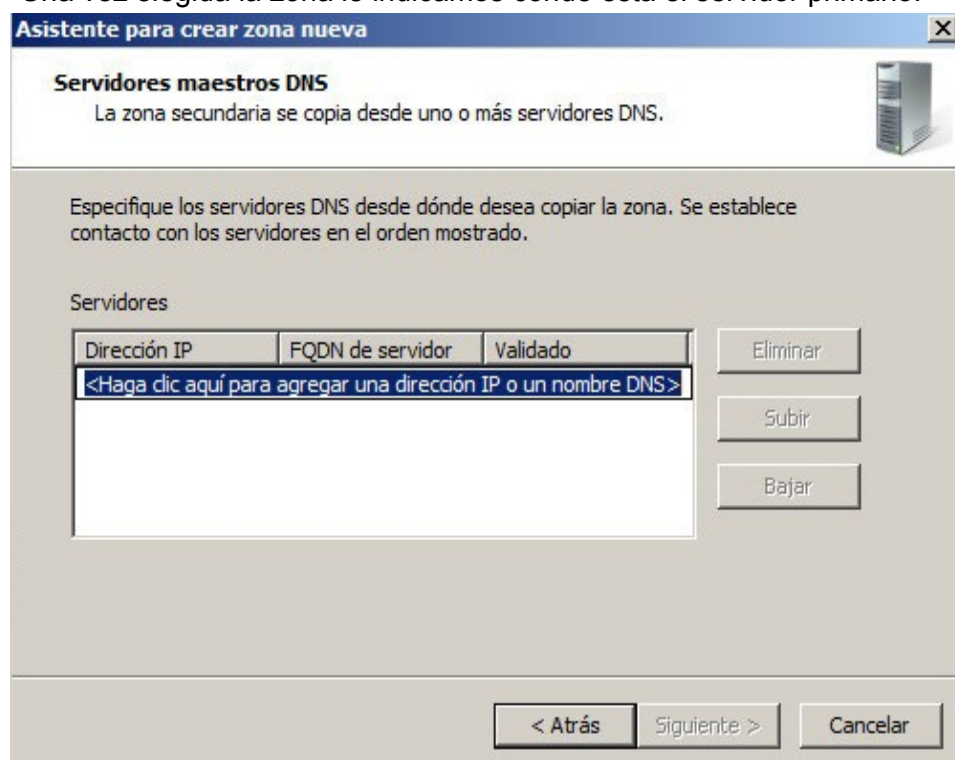
☒ Zona secundaria
Crea una copia de una zona que ya existe en otro servidor. Esta opción ayuda a equilibrar el proceso de carga de los servidores primarios y proporciona tolerancia a errores.

☐ Zona de rutas internas
Crea una copia de zona que contiene sólo servidor de nombres (NS), inicio de autoridad (SOA) y quizá registros de adherencia de host (A). Un servidor que contiene una zona de rutas internas no tiene privilegios sobre dicha zona.

☐ Almacenar la zona en Active Directory (sólo disponible si el servidor DNS es un controlador de dominio grabable)

< Atrás Siguiendo > Cancelar

Una vez elegida la zona le indicamos donde está el servidor primario:



Asistente para crear zona nueva

Servidores maestros DNS
La zona secundaria se copia desde uno o más servidores DNS.

Especifique los servidores DNS desde dónde desea copiar la zona. Se establece contacto con los servidores en el orden mostrado.

Servidores

Dirección IP	FQDN de servidor	Validado
<Haga clic aquí para agregar una dirección IP o un nombre DNS>		

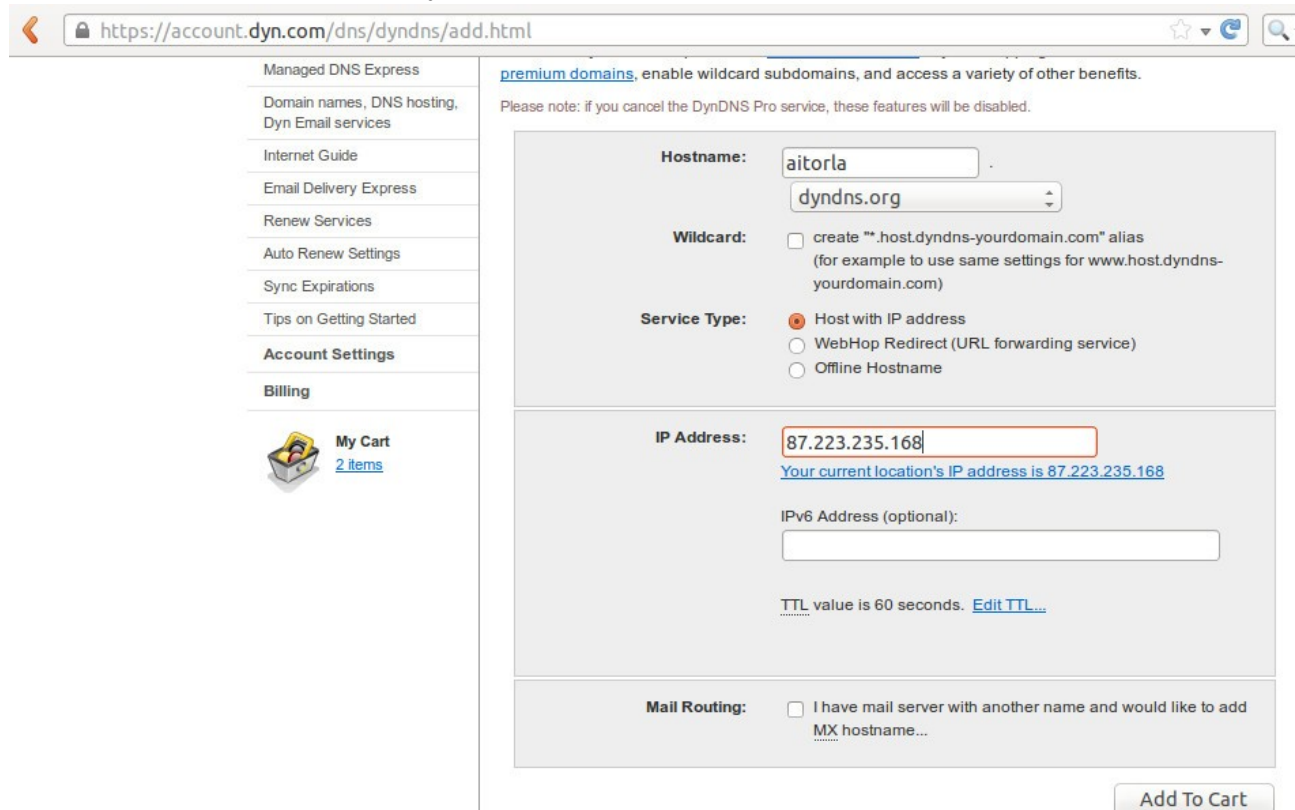
Eliminar
Subir
Bajar

< Atrás Siguiendo > Cancelar

DNS dinámico o DDNS

El DDNS o DNS dinámico nos permite asociar un nombre a una dirección IP dinámica. Nuestros proveedores de Internet nos asignan una dirección IP pública a nuestros routers que cambia (cada cierto tiempo o cuando reiniciamos el router). Existen diferentes herramientas como dyndns.org o no-ip.com. El proceso es sencillo:

1.- Creamos una cuenta y la asociamos a nuestra dirección IP



The screenshot shows the DynDNS account setup page in a web browser. The address bar displays <https://account.dyn.com/dns/dyndns/add.html>. On the left is a sidebar menu with links: Managed DNS Express, Domain names, DNS hosting, Dyn Email services, Internet Guide, Email Delivery Express, Renew Services, Auto Renew Settings, Sync Expirations, Tips on Getting Started, Account Settings, and Billing. Below the menu is a 'My Cart' section showing '2 items'. The main content area has a header with a link to 'premium domains' and a note about canceling the DynDNS Pro service. The form contains the following fields and options:

- Hostname:** 'aitorla' and 'dyndns.org' (dropdown).
- Wildcard:** ☐ create "*.host.dyndns-yourdomain.com" alias (for example to use same settings for www.host.dyndns-yourdomain.com).
- Service Type:** ☒ Host with IP address, ☐ WebHop Redirect (URL forwarding service), ☐ Offline Hostname.
- IP Address:** '87.223.235.168' (highlighted in red). Below it, a link says 'Your current location's IP address is 87.223.235.168'. There is also an optional IPv6 Address field.
- TTL value:** 60 seconds. A link 'Edit TTL...' is provided.
- Mail Routing:** ☐ I have mail server with another name and would like to add MX hostname...

An 'Add To Cart' button is located at the bottom right of the form.

2.- Accedemos al router y lo configuramos con ese usuario y password que hemos creado.

The screenshot shows the web interface of a Comtrend ADSL Router. The browser address bar displays '192.168.1.1'. The page title is 'COMTREND ADSL Router'. On the left is a navigation menu with the following items: Device Info, Advanced Setup, Layer2 Interface, WAN Service, LAN, NAT, Security, Quality of Service, Routing, DNS (highlighted), DNS Server, Dynamic DNS, DSL, UPnP, DNS Proxy, Print Server, and Storage Service. The main content area is titled 'Add Dynamic DNS' and includes the text: 'This page allows you to add a Dynamic DNS address from DynDNS.org or TZO.' The configuration fields are as follows: 'D-DNS provider' is a dropdown menu set to 'DynDNS.org'; 'Hostname' is a text input field containing 'aitoria.dyndns.org'; 'Interface' is a dropdown menu set to 'internet/ppp0.1' (highlighted with a red box); 'DynDNS Settings' includes a 'Username' field with 'aitoria' and a 'Password' field with masked characters '*****'. An 'Apply/Save' button is located at the bottom right of the form.

Ejercicio 5

Simula un servidor de DNS para el dominio google.com con los datos que se han obtenido en el ejercicio anterior.

Ejercicios Resueltos

Ejercicio 1.1

Obtén los datos del titular, contacto administrativo, técnico y servidores DNS de los dominios google.com, google.es, terra.es y yahoo.com.

Vemos que con el comando whois google.es y terra.es no obtenemos información pero si de los otros dos. Por Ejemplo google.com:

Dns Admin
Google Inc.
Please contact contact-admin@google.com 1600 Amphitheatre Parkway
Mountain View CA 94043
US
dns-admin@google.com +1.6502530000 Fax: +1.6506188571

Domain Name: google.com

Registrar Name: Markmonitor.com
Registrar Whois: whois.markmonitor.com
Registrar Homepage: <http://www.markmonitor.com>

Administrative Contact:

DNS Admin
Google Inc.
1600 Amphitheatre Parkway
Mountain View CA 94043
US
dns-admin@google.com +1.6506234000 Fax: +1.6506188571

Technical Contact, Zone Contact:

DNS Admin
Google Inc.
2400 E. Bayshore Pkwy
Mountain View CA 94043
US
dns-admin@google.com +1.6503300100 Fax: +1.6506181499

Created on.....: 1997-09-15.

Expires on.....: 2020-09-13.

Record last updated on...: 2013-10-02.

Domain servers in listed order:

ns4.google.com
ns3.google.com

Para obtener la información de los dominios .es vamos a la web de nic.es

Información de Dominio

DATOS DEL TITULAR

Nombre del Dominio	google.es
Estado	Activado
Identificador	GI1137-ESNIC-F4
Titular	GOOGLE INC.
Fecha de Alta	16-09-2003
Fecha de Caducidad	16-09-2014
Agente Registrador	MARKMONITOR

PERSONA DE CONTACTO ADMINISTRATIVO

Identificador	TT624-ESNIC-F4
Nombre	Tu Tsao
Email	dns-admin@google.com

PERSONA DE CONTACTO TECNICO

Identificador	TT624-ESNIC-F4
Nombre	Tu Tsao
Email	dns-admin@google.com

SERVIDORES DNS

Nombre Servidor	IP
ns2.google.com	
ns1.google.com	

Ejercicio 1.2

Realiza una tabla comparativa de los precios que ofrecen diferentes empresas registradoras de dominios.

	.com	.net	.org	.es	.eu
1and1.es	9,99 €	9,99 €	9,99 €	9,99 €	9,99 €
Godaddy.com	9,99 €	7,99 €	7,99 €	12,89 €	8,99 €
Arsys.es	10 €	25 €	25 €	10 €	25 €

Nota: Precio anual sin tener en cuenta las ofertas del primer año.

Ejercicio 1.3

Obtener información adicional desde la web whois.sc de los dominios del ejercicio 1.

Whois Record Site Profile Registration Server Stats For Sale

Whois Record

Reverse Whois: **"Dns Admin" owns about 55,266 other domains**


Email Search: contact-admin@google.com is associated with about **66 domains**
dns-admin@google.com is associated with about **13,144 domains**


Registrar History: [3 registrars](#)

IP History: [215 changes](#) on **35** unique IP addresses over **9** years.

Whois History: [3,956 records](#) have been archived **since 2001-05-03**.

Reverse IP: [133 other sites](#) hosted on this server.

 [Log In](#) or [Create a FREE account](#) to start monitoring this domain name

 Preview the complete [Domain Report for google.com](#)


Registrant:
Dns Admin
Google Inc.
Please contact contact-admin@google.com 1600 Amphitheatre Parkway
Mountain View CA 94043
US
dns-admin@google.com +1.6502530000 Fax: +1.6506188571

Domain Name: google.com

Registrar Name: Markmonitor.com
Registrar Whois: whois.markmonitor.com
Registrar Homepage: <http://www.markmonitor.com>

Whois Record **Site Profile** Registration Server Stats For Sale







Site Profile and Search Rank

Website Title:  [Google](#)

Title Relevancy **100%**

Meta Description: Search the world's information, including webpages, images, videos and more.
Google has many special features to help you find exactly what you're looking for.

Description Relevancy: **24%** relevant.

Visitors by Country:  United States 30%  Brazil 3.3%
 India 8.8%  Great Britain (UK) 2.9%
 Russian Federation 3.3%  Japan 2.5%

Alexa Trend/Rank: **#1**: for the last three months.

Quantcast Rank: **#1**

DMOZ: [2,491 listings](#)


Y! Directory: [145 listings](#)

Yahoo Title: **Google**

Yahoo Description: Google allows users to search the Web for images, news, products, video, and other content.

Yahoo Category: / Business and Economy / Business to Business / Communications and Networking / Internet and World Wide Web /

Wikipedia: [666,533 pages](#)

AboutUs:  [Wiki article on Google.com](#)

[SEO Score](#): **96%**

Terms: **64** (Unique: 57, Linked: 41)

Images: **1** (Alt tags missing: 0)

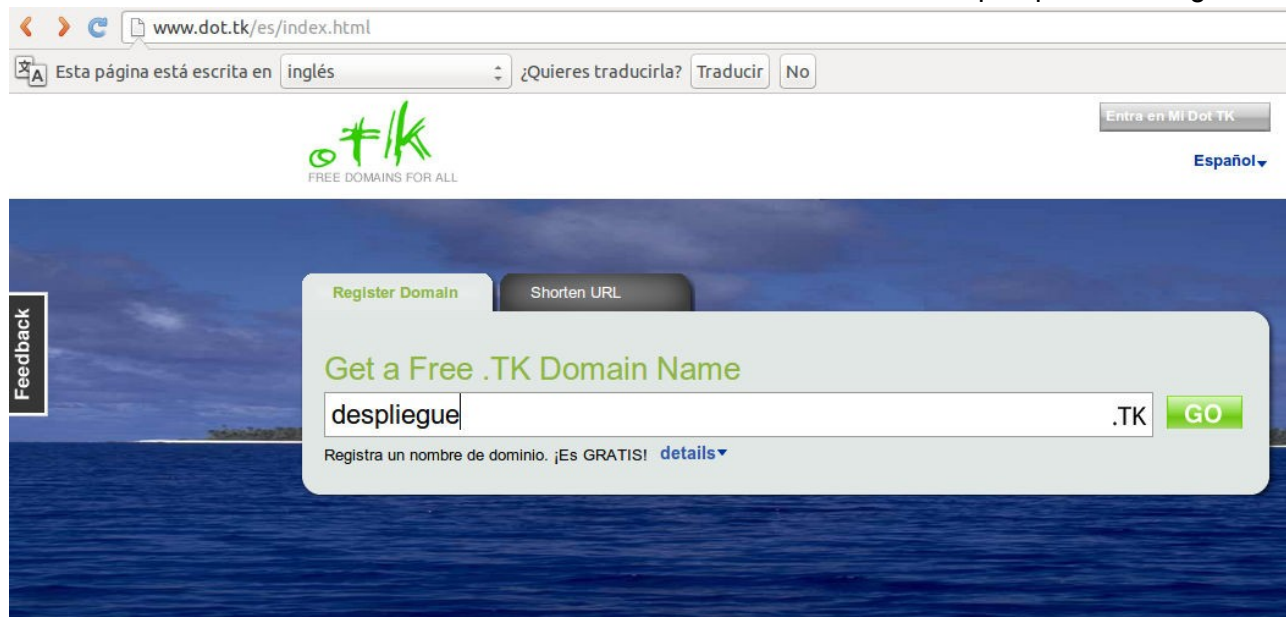
Links: **32** (Internal: 30, Outbound: 2)

Related Sites: [amazon.com](#), [yelp.com](#), [yahoo.com](#), [wikipedia.org](#), [statcounter.com](#), [myspace.com](#),
[linkedin.com](#), [imdb.com](#), [ask.com](#), [youtube.com](#)

Ejercicio 1.4

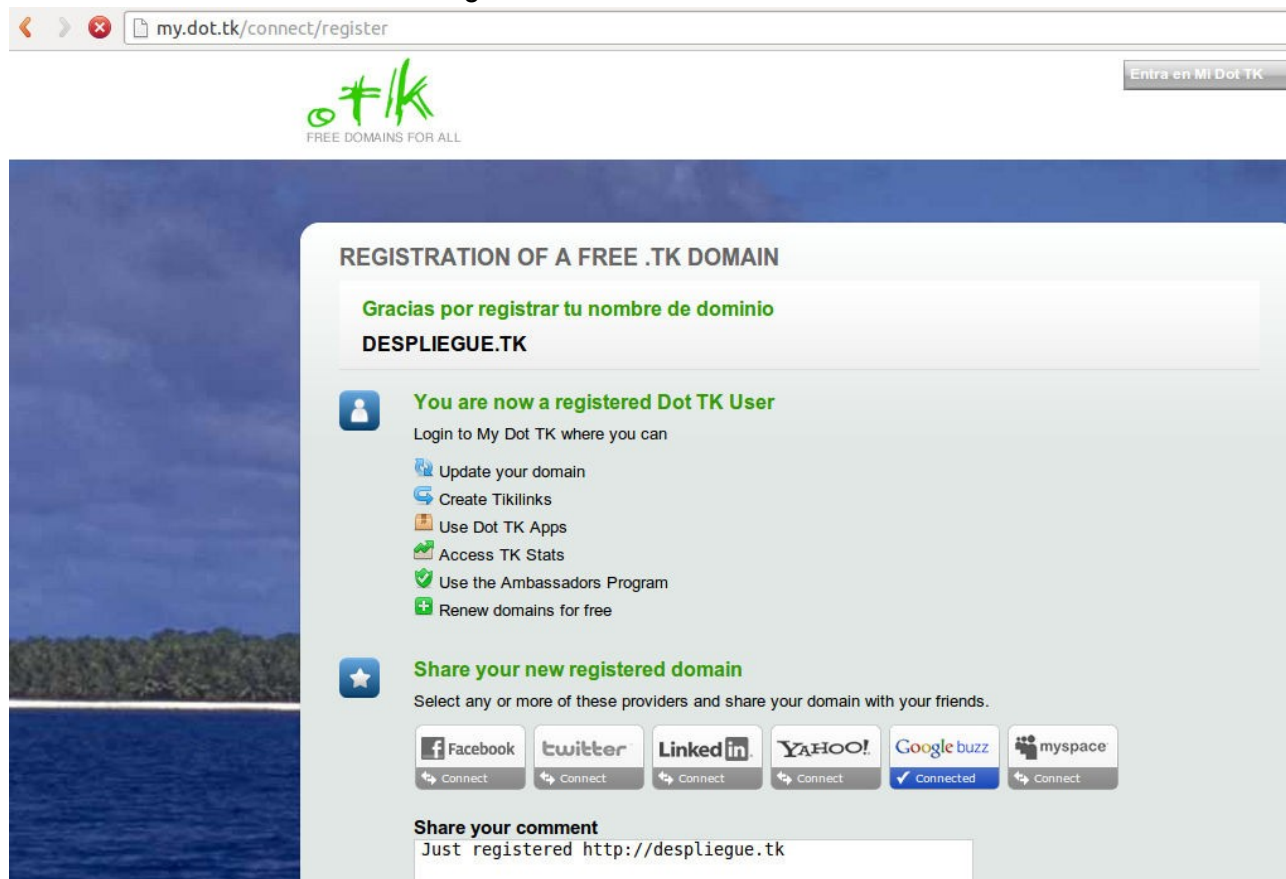
Registra un dominio en .tk en dom.tk

Accedemos a la web dot.tk e introducimos un nombre del dominio que queremos registrar.



The screenshot shows the website www.dot.tk/es/index.html. The page has a header with a language selector set to 'inglés' and a 'Traducir' button. The main content area features a 'Register Domain' button and a 'Shorten URL' button. Below these, there is a form titled 'Get a Free .TK Domain Name' with a text input field containing 'despliegue', a '.TK' dropdown menu, and a 'GO' button. A 'Feedback' link is visible on the left side of the page.

Si el dominio está libre lo registramos



The screenshot shows the my.dot.tk/connect/register page. The page displays the 'REGISTRATION OF A FREE .TK DOMAIN' confirmation. It includes a message: 'Gracias por registrar tu nombre de dominio DESPLIEGUE.TK'. Below this, it states 'You are now a registered Dot TK User' and provides a list of actions: 'Login to My Dot TK where you can', 'Update your domain', 'Create Tikilinks', 'Use Dot TK Apps', 'Access TK Stats', 'Use the Ambassadors Program', and 'Renew domains for free'. There is also a section for 'Share your new registered domain' with social media sharing options (Facebook, Twitter, LinkedIn, Yahoo!, Google Buzz, MySpace) and a 'Share your comment' section with the text 'Just registered http://despliegue.tk'.

Podemos reenviar el dominio, o utilizar las propias DNS que queramos nosotros.

my.dot.tk/registration/register?domainname=despliegue

Esta página está escrita en inglés ¿Quieres traducirla? Traducir No

You are registering the following domain name...
DESPLIEGUE.TK

Use your new domain

☐ Forward this domain to ☒ Use DNS ☐ Build a new website with [Imcreator.com](#)

Servicio DNS de Dot TK [Tu propio DNS](#)

Usa el servicio gratuito DNS de Dot TK (introduce aquí tus archivos clave)

Host name	www.despliegue.tk	Dirección IP	217.160.82.170
Host name	despliegue.tk	Dirección IP	217.160.83.170

Registration length

Domains can be registered from 1 to 12 months. Registered Dot TK users are allowed an unlimited number of domain renewals.

3 months

Introduce los caracteres que ves aquí

A D V L K G H N

Introduce los caracteres que ves aquí

Aquí tenemos una vista del panel de control.

my.dot.tk/cgi-bin/domainpanel.taloha

Esta página está escrita en inglés ¿Quieres traducirla? Traducir No

MY dot TK [Logout](#)

[Home](#) [Profile](#) **[Domain Panel](#)** [TK Shop](#) [Apps](#) [Ambassadors](#)

[Domain Panel](#) [Add a domain](#) [Transfer domain](#)

Domain Panel [More...](#)

Domain Name	Type	Status	Reg. Date	Exp. Date	
DESPLIEGUE.TK	Free	Authorized	15/09/2013	15/09/2014	Upgrade Renew Modify Cancel

Special deals for despliegue.tk

[tiendy](#) crea tu tienda [TRENDY Site Builder](#) Try Free!

Domains per page: 10

Ejercicio 1.5

Escribe los registros zonas de los dominios google.com, google.es, terra.es y yahoo.com a partir de los datos que puedes obtener de Internet.

nslookup

>set type=ANY

>google.com

profesordeinformatica.com

> set type=SOA

> google.com

A partir del comando anterior construimos el registro de zona:

```
google.com IN SOA ns1.google.com . dns-admin.google.com (  
    2013100300 ; serial  
    7200 ; refresh  
    1800 ; retry  
    1209600 ; expire  
    300 ; minimun TTL  
)
```

google.com. IN NS ns2.google.com.

google.com. IN NS ns3.google.com.

google.com. IN NS ns4.google.com.

google.com. IN NS ns1.google.com.

google.com. IN AAAA 2a00:1450:4003:802::1006

google.com. IN A 173.194.34.230

google.com. IN A 173.194.34.233

google.com. IN A 173.194.34.228

google.com. IN A 173.194.34.226

google.com. IN A 173.194.34.224

google.com. IN A 173.194.34.225

google.com. IN A 173.194.34.238

google.com. IN A 173.194.34.229

google.com. IN A 173.194.34.227

google.com. IN A 173.194.34.231

ns3.google.com. IN A 216.239.36.10

ns4.google.com. IN A 216.239.38.10

ns1.google.com. IN A 216.239.32.10

ns2.google.com. IN A 216.239.34.10

alt2.aspmx.l.google.com. IN A 173.194.69.26

alt4.aspmx.l.google.com. IN A 173.194.79.26

google.com. IN MX 40 alt3.aspmx.l.google.com.

google.com. IN MX 30 alt2.aspmx.l.google.com.

google.com. IN MX 50 alt4.aspmx.l.google.com.

google.com. IN MX 10 aspmx.l.google.com.

google.com. IN MX 20 alt1.aspmx.l.google.com.

Ejercicio 1.6

Realiza una comparación de los tiempos que están definidos en cada uno de los dominios.

Dominio/Tiempos	Refresco	Reintento	Expiración	TTL Mínimo
Google.com	7200	1800	1209600	300
Google.es	900	900	1800	60
Terra.es	28800	7200	2592000	172800
Yahoo.com	3600	300	1814400	600

Ejercicio 1.7

Explica las ventajas y desventajas que puede suponer tener el TTL de 5 minutos o 10 horas.

En el ejercicio anterior vemos que Google.es solo tiene un TTL de 1 minuto comparando con las 48 horas que tiene Terra. La ventaja que tiene Google.es es que cada vez que realice modificaciones sobre su servidor de DNS su tiempo de propagación al resto de servidores de DNS del mundo va a durar muy poco 1 minuto. Como desventaja hay que decir que en el servidores cachés solo va a mantenerse la información 1 minuto con lo que la solicitud de esta información sobre el servidor de DNS google.es va a ser muy frecuente.

Por otra parte el servidor de Terra cuando se realice cualquier cambio va a tardar en propagarse 48 horas. Como el tiempo es grande se va a mantener la información en los servidores cachés durante 48 horas, lo que hace que las solicitudes a este servidor sean menos frecuentes.

Capítulo 2

Servicio HTTP

Introducción

El servicio HTTP (Hypertext transfer protocol) o protocolo de transferencia de Hipertexto, nos permite transferir información entre un cliente o navegador web y un servidor web a través del puerto 80. Actualmente existen dos versiones del protocolo HTTP, la versión 1.0 y la versión 1.1.

HTTPS es la versión segura del protocolo HTTP y utiliza el puerto 443 y un cifrado basado en SSL/TLS (Secure Socket Layer y su sucesor Transfer Layer Security).

Se pueden ver los diferentes puertos predefinidos en el fichero **/etc/services**.

Los tipos MIME

Los tipos MIME son unas especificaciones usadas para dar formato a mensajes no ASCII. **IANA** es el organismo internacional que define los tipos MIME. Inicialmente el protocolo HTTP te permitía enviar únicamente texto pero actualmente HTTP permite enviar mensajes con encabezados que describen el contenido del propio mensaje mediante codificación MIME.

Podemos ver los diferentes tipos MIME que existen en el fichero de linux **/etc/mimes.types** o en su fichero de Windows equivalente.

Ejemplos: text/html, text/css, video/mpeg.

Las situaciones en las cuales se usan los tipos MIME son las siguientes:

- Para informar al navegador del tipo de datos que esta recibiendo del servidor. Se define usando: **<content-type>** dependiendo de si el navegador puede interpretarlo puede visualizar el documento (si se trata de <text/html>), llamar a una aplicación externa (<application/pdf> o preguntar al usuario que hacer (<image/x-fwf>).
- Para permitir negociar el contenido: **allow: application/zip**
- Para encapsular uno o mas objetos dentro del cuerpo del mensajes por medio de los MIME multipart (<multipart/formdata>, para enviar los datos de un formulario)

La direccies web URN/URL/URI

URL (Localizadores Uniformes de Recursos) son secuencias de caracteres (ASCII) que se utilizan para localizar recursos en Internet (paginas web, sonidos, vídeos, imágenes, etc).

Los **URN** (Nombres Uniformes de Recursos) identifican a un recurso en Internet, pues no los localizan.

Los **URI** (Identificadores Uniformes de Recursos) identifican inequívocamente a un recurso.

URN: Esquema://userinfo@dominio:Puerto

URL: Esquema://[userinfo@dominio](#):Puerto/camino?Query

URI: Esquema://userinfo@dominio:Puerto/camino?Query#Fragmento

Esquema: es el protocolo empleado para la comunicación: http, https, ftp.

UserInfo: son los datos para autentificación del usuario: usuario@

Dominio: es el FQDN o la dirección IP donde se encuentra el recurso en Internet.

Puerto: es el protocolo de comunicación que empleara el protocolo.

Camino: es la ruta donde se aloja el recurso dentro del servidor.

Query: son datos que se envían al servidor en modo consulta.

Fragmento: especifica una parte/posición dentro del recurso.

Funcionamiento

1. El usuario teclea en el cliente web la url de la pagina a consultar.
2. El cliente http decodifica la información obteniendo el protocolo, la IP o el nombre del servidor web, puerto, etc.
3. El cliente (navegador) conecta con el servidor web y le solicita la pagina web.
4. El servidor envía la pagina web o devuelve el código de error correspondiente.
5. Cliente interpreta los códigos html recibidos.
6. La conexión se cierra.

Si el navegador solicita una pagina web con 2 imágenes, van a existir 3 conexiones:

- La primera para el documento HTML.
- La segunda para el envío de los archivos.

Cuando se establece una conexión HTTP tenemos dos tipos de mensajes entre un cliente y un servidor: una petición por parte del cliente y una respuesta por parte del servidor.

Petición HTTP (Request)

El formato es el siguiente: <método> <URI> <versión>

Ejemplo:

GET /index.html HTTP/1.0

Donde el **método** indica al servidor que hacer con el URI, pueden ser:

HTTP 1.0:

- **GET:** Obtiene información del servidor.
- **HEAD:** Obtiene la cabecera de la pagina.
- **POST:** Empleado para enviar información al servidor.

HTTP 1.1:

- Los métodos del HTTP 1.0.
- **PUT:** Envía un objeto al servidor.
- **DELETE:** Solicita al servidor que borre el recurso indicado en el mensaje.
- **OPTIONS:** Para negociar parámetros de la comunicación entre cliente y servidor.

URI - Indica al recurso de forma exacta en el servidor.

La **versión** indica la versión del protocolo HTTP (HTTP/1.0 o HTTP/1.1)

Respuesta (Response)

Tiene el siguiente formato: <version> <codigo de estado> <texto explicativo>

Puede devolver:

- Respuesta Positiva => HTTP 1.1 200 OK
- Respuesta Negativa => HTTP 1.1 405 Method Not Allowed

Códigos de Estado HTTP

Un servidor web nos puede dar diferentes respuestas tras solicitarle una página web. Están agrupados por centenas. Ponemos ejemplos de los códigos devueltos habitualmente.

1xx: Respuestas informativas

100 111 Conexión
rechazada

2xx: Peticiones correctas

200 OK
201-203 Información no oficial
204 Sin Contenido
205 Contenido para
recargar
206 Contenido parcial

3xx: Redirecciones

301 Movido
permanentemente
302 Encontrado
303 Vea otros

304 No modificado
305 Utilice un proxy
307 Redirección temporal

4xx Errores del cliente

400 Solicitud incorrecta
401 No autorizado
402 Pago requerido
403 Prohibido
404 No encontrado
409 Conflicto
410 Ya no disponible
412 Falló precondition

5xx Errores de servidor

500 Error interno
501 No implementado
502 Pasarela incorrecta
503 Servicio no disponible
504 Tiempo de espera de la pasarela agotado
505 Versión de HTTP no soportada

Servidor Web Seguro

SSL -> HTTPS

SSL: trabaja con dos tipos de criptografía:

- **Simétrica:** durante la transmisión de datos.
- **Asimétrica:** en la fase de autentificación.

SSL proporciona:

- Confidencialidad.
- Integridad del Mensaje.
- Autentificación del Servidor.
- Autentificación del Cliente.

Proceso de Negociación

1. El cliente se conecta al servidor.
2. Se inicia la fase de saludo HandShake, en la que el cliente envía un mensaje llamado "Client Hello".
3. El servidor responde con un mensaje "Server Hello".
4. Servidor envía un mensaje "Certificate" en el cual envía un certificado X509 que contiene la clave publica del servidor.

5. Opcionalmente el servidor puede requerir la autentificación al cliente en un mensaje llamado "Certificate Request".
6. Cliente autentica al servidor un envía un mensaje "Certificate Verify".
7. El cliente genera la clave de sesión y la envía codificada con la clave publica del servidor en un mensaje "Key Exchange".
8. Opcionalmente, el cliente envía su certificado en un mensaje "Certificate".
9. Cliente y Servidor determinan una clave de sesión.
10. El cliente y el servidor intercambian mensajes "Change Cipher Spec" para indicar al otro que desde ese momento se emplea la clave acordada.
11. Termina la fase del saludo y el servidor envía el mensaje "Finished".
12. Comienza la fase de Comunicación.

Últimamente se esta implementando un nuevo protocolo de seguridad, llamado **TLS** que esta sustituyendo al SSL (versión 3).

Autoridades de certificación (CA)

El SSL (Secure Socket Layer) es un protocolo de seguridad, desarrollado por la empresa Netscape Communications, para lograr que la transmisión de datos entre un servidor y un usuario, o viceversa, a través de Internet, sea completamente segura. Existen multiples agencias que emiten certificados como Verisign, geotrust, thawte. En la siguiente tabla se puede ver una comparativa de precios:

http://www.entorno.es/certificados_ssl.php

Servidor Web

Los servidores suministran 2 tipos de paginas web HTML:

1. Estática: el navegador interpreta los códigos Html y muestra imágenes, fuentes, formatos, etc.

2. Dinámicas : permiten crear aplicaciones dentro de la misma pagina web permitiendo interactuar con el usuario y la pagina web se crea según las peticiones de ese usuario.

En ocasiones el lenguaje usado para crear la pagina web es interpretado por el navegador que visualiza la pagina (javascript o VisualBasic Script).

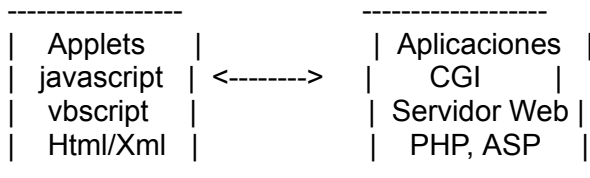
En otras ocasiones el servidor web es quien interpreta, genera y envía la pagina web generada al cliente web que la visualiza (PHP, ASP, JSP, Perl, Python, manejadores de bases de datos como MySQL, Oracle, SQL Server).

En este segundo caso se consiguen dos objetivos:

1. Que no haga errores de interpretación de código.
2. Que el usuario final no tenga acceso al código original de la pagina web y pueda plagiarla.

Una aplicación web dinámica se compone de 2 elementos:

-----		-----
Cliente		Servidor



- **Applets** = componentes de una aplicación que se ejecuta en el contexto de otro programa.
- **CGI** = estándar que permite la conexión de cualquier aplicación hecha en cualquier lenguaje para que pueda interactuar con el servidor web.
- **Servlets** = componentes que se ejecutan en los servidores web para generar contenidos dinámicos.
- **JavaBeans** = componente de software, utilizable para Java.

2.2.- Clientes web

El navegador es el software que nos va a permitir visualizar una página web. Existen en el mercado infinidad de navegadores, versiones para diferentes sistemas operativos. Lo habitual suele ser que una vez que hemos realizado nuestra página web no se visualice de la misma forma en un navegador u otro. Nuestro trabajo como desarrolladores web debería ser que se visualizase lo mejor posible en todos y sobre todo en los que se utilizan en el mercado.

En los comienzos utilizábamos navegadores de texto como Lynx.

Actualmente, los navegadores más utilizados son Firefox, Chrome, Explorer, Opera y Safari aunque existen muchos otros más. La parte más importante de los navegadores es su núcleo, la parte encargada en dibujar las páginas. Vamos a ver alguna característica de estos navegadores:

Chrome: Es gratuito y de código abierto desarrollado por Google. Es actualmente el navegador más utilizado ya que lo utilizan más del 50% de los usuarios. Su motor es **Blink** es un proyecto derivado de **WebKit** como WebKit derivó en su día de KHTML.

Firefox: Desarrollado por la fundación Mozilla es de código abierto y gratuito. Está desarrollado para plataformas Windows, Mac OS X, Linux y Android. Utiliza el motor **Gecko**.

Explorer: Es el navegador que se incorpora de forma gratuita con los sistemas operativos Windows lo que le ha ocasionado diferentes demandas a Microsoft. Utiliza el motor **Trident**.

Safari: Es un navegador de código cerrado desarrollado por Apple. Actualmente funciona tanto en Windows como el Mac. Utiliza el motor **Blink**.

Opera: Desarrollado por la empresa noruega Telenor. Es gratuito y intentan ofrecer el navegador con menos peso y veloz de Internet. Abandonó su propio motor Presto por **WebKit**.

Ejercicio 1

Instala lynx y navega por internet.

Ejercicio 2

Instala y prueba diferentes navegadores. Así como diferentes pluguins o complementos.

Ejercicio 3

Prueba el correcto funcionamiento del servidor con el comando telnet.

2.3.- Servidores web

Un servidor web o servidor HTTP es un software que está en ejecución dentro de un servidor esperando a que el cliente le haga una petición HTTP. El resultado de esta petición puede ser una página web o cualquier elemento MIME.

Actualmente existen dos servidores que son los más utilizados en Internet. **Apache**, que es de código abierto y multiplataforma e **Internet Information Server o ISS** es de Microsoft, solo funciona en sistemas operativos Windows. Existen otros servidores que no se utilizan prácticamente como NCSA, Cheeroke o Iplanet Web Server.

Práctica 1: Instalación del servidor Apache

Para instalar el servidor apache simplemente tenemos que instalar el módulo apache2:

apt-get install apache2

Para comprobar que el servidor apache funciona correctamente, podemos abrir un navegador y poner en la url **localhost** o **127.0.0.1**.

En el directorio **/var/www** vamos a guardar por defecto nuestras webs. En ella vamos a tener el fichero **index.html** que va a contener la primera página web. Nuestro servidor está configurado de tal forma que si nosotros sólo ponemos el directorio el servidor va a visualizar el fichero index.html, index.htm o index.php.

Para arrancar el servidor lo podemos hacer de la siguiente forma:

/etc/init.d/apache2 start

Para parar el servidor lo podemos hacer de la siguiente forma:

/etc/init.d/apache2 stop

Para reiniciar el servidor lo podemos hacer de la siguiente forma:

/etc/init.d/apache2 restart

Podemos comprobar que funciona el servidor correctamente haciendo un telnet al puerto 80.

Podemos ver los procesos que genera el servidor

ps -ef | grep apache2

```
root    6457    1 0 19:56 ?        00:00:00 /usr/sbin/apache2 -k start
www-data 6460  6457 0 19:56 ?        00:00:00 /usr/sbin/apache2 -k start
www-data 6462  6457 0 19:56 ?        00:00:00 /usr/sbin/apache2 -k start
www-data 6463  6457 0 19:56 ?        00:00:00 /usr/sbin/apache2 -k start
```

Podemos matar los procesos apache de la siguiente forma:

```
ps -ef | grep apache2 | awk -F" " '{print $2}' | xargs kill -9
```

Ejercicio

Vamos a acceder al contenido del servidor con el comando telnet

telnet localhost 80

Trying 127.0.0.1...

Connected to localhost.

Escape character is '^['.

GET / HTTP/1.1

HOST:localhost

HTTP/1.1 200 OK

Date: Mon, 14 Oct 2013 20:55:04 GMT

Server: Apache/2.2.22 (Ubuntu)

Last-Modified: Mon, 14 Oct 2013 17:56:15 GMT

ETag: "66091b-b1-4e8b72f36510b"

Accept-Ranges: bytes

Content-Length: 177

Vary: Accept-Encoding

Content-Type: text/html

X-Pad: avoid browser bug

<html><body><h1>It works!</h1>

<p>This is the default web page for this server.</p>

<p>The web server software is running but no content has been added, yet.</p>

</body></html>

Monitorización Servidor Apache

Existe el fichero **/var/log/apache2/access.log** que nos da información de las páginas web que se han visitado.

Existe el fichero **/var/log/apache2/error.log** que nos da información de los errores que se producen en nuestro servidor web.

Podemos utilizar herramientas como **webalizer** para obtener estadísticas de nuestro sitio web.

Práctica 2: Acceso Restringido en apache

Podemos restringir el acceso a determinados directorios de nuestro servidor, de manera que cuando accedamos a ellos con el navegador se nos pedirá un nombre de usuario y una clave. Esos usuarios y esas claves no van a guardar ninguna relación con los del sistema Linux.

Queremos que cuando accedamos al `/var/www/restringido` (<http://www.localhost/resgringido>) nos pida un usuario y un password para ver su contenido.

1.- Creamos un directorio `clavesapache` donde vamos a guardar el fichero de usuarios y claves `.clavesapache` y le damos permiso de lectura, escritura y ejecución para el usuario `www-data`, que es el usuario por defecto de apache, lectura y ejecución para el grupo `www-data` y sin permisos para el resto de usuarios:

```
# mkdir /var/www/clavesapache
# chown www-data /var/www/clavesapache
# chmod 750 /var/www/clavesapache
```

Vamos a añadir diferentes `/var/www/clavesapache` y ajustamos propietario y permisos:

```
# htpasswd -c /var/www/clavesapache/.clavesusuarios usuario1
(introducimos el password y la verificación de su password, por ejemplo clave1)
# chown www-data:www-data /var/www/clavesapache/.clavesusuarios
# chmod 640 /var/www/clavesapache/.clavesusuarios
```

Podemos añadir más usuarios (sin el `-c`, ya que ya existe el fichero `.clavesusuarios`):

```
# htpasswd /var/www/clavesapache/.clavesusuarios usuario2
```

Vemos que hemos creado los usuarios con sus respectivas claves encriptadas en el fichero oculto `.clavesusuarios`:

cat .clavesusuario

```
usuario1:$apr1$e4xMilKy$9NzX8TEmqWiMTkqGHp.0G0
```

```
usuario2:$apr1$YC5LjTIX$lyXaaKqNK997HSV78BtUF1
```

2.- Configuramos el fichero `/etc/apache2/sites-available/default` para que nos solicite un password. Para ello sustituimos `AllowOverride None` por **AllowOverride AuthConfig**:

```
<Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride AuthConfig
    Order allow,deny
    allow from all
</Directory>
```


3.- Creamos el directorio restringido y dentro creamos el fichero

/var/www/restringido/.htaccess con el siguiente contenido:

```
AuthType Basic
AuthName "Acceso Restringido"
AuthUserFile /var/www/clavesapache/.clavesusuarios
Require user usuario1
```

Además creamos el fichero index.html con una página de bienvenida y les damos permisos con el usuario y grupo www-data.

```
cd /var/www/restringido
chown www-data:www-data /var/www/restringido/.htaccess
chown www-data:www-data /var/www/restringido/index.html
chown www-data:www-data /var/www/restringido
```

4.- Reiniciamos y probamos:

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

Entramos en el navegador a localhost/restringido

Práctica 3: Servidores virtuales

Con la configuración de servidores virtuales, en un servidor HTTP podemos tener varios dominios con una misma dirección IP. Vamos a suponer que tenemos varios dominios registrados: dominio1.com, dominio2.com y dominio3.com, la idea es que cuando visualicemos dominio1.com nos aparezca la página de dominio1 y cuando pongamos en la url dominio2.com nos aparezca la página de dominio2.

1.- Borramos todo el apache

```
apt-get remove --purge apache2 apache2.2-common
rm -rf /etc/apache2
rm -rf /var/www/dominio2
rm -rf /var/www/dominio1
```

2.- Creamos dos dominios en DNS o en /etc/hosts que apunte a nuestro servidor

```
127.0.0.1    localhost  www.dominio2.com www.dominio1.com
```

3.- Creamos las paginas web de dominio2 y dominio1 en:

```
/var/www/dominio2/index.html
/var/www/dominio1/index.html
```

4.- Instalamos apache

```
apt-get install apache2
```

5.- Creamos los ficheros para los dominios virtuales

profesordeinformatica.com

Creamos en **/etc/apache2/sites-available/www.dominio3.com.conf** con la siguiente información

```
<VirtualHost *:80>
DocumentRoot "/var/www/dominio2"
ServerName www.dominio2.com
<Directory "/var/www/dominio2">
allow from all
Options +Indexes
</Directory>
</VirtualHost>
```

```
cd /etc/apache2
creamos un enlace simbolico:
ln -s /etc/apache2/sites-available/www.dominio2.com.conf
/etc/apache2/sites-enabled/www.dominio2.com.conf
```

Activamos el virtual host
a2ensite www.dominio2.com.conf

/etc/init.d/apache2 restart

CON WEBMIN

El último punto lo podemos hacer con webmin

Webmin → Servidores → Servidor web apache → Create virtual host

Nota: Si vemos que entre los servidores no tenemos el servidor web apache buscamos apache entre los módulos.

Rellenamos:

Raiz para documentos: **/var/www/dominio3**
Nombre del servidor: **www.dominio3.com**

Pinchamos en **Crear ahora**

(Lo mismo para dominio2.com)

Pinchamos en **aplicar cambios**

y probamos en el navegador firefox

(Importante borrar caché por si acaso,....)

Práctica 4: SERVIDOR WEB SEGURO

1.- Creamos el certificado seguro y la clave

Generamos la clave

profesordeinformatica.com

apt-get install openssl

openssl genrsa -des3 -out server.key 1024

Generating RSA private key, 1024 bit long modulus

.....++++++

...++++++

e is 65537 (0x10001)

Enter pass phrase for server.key:

Enter pass phrase for server.key:

Verifying - Enter pass phrase for server.key:

Generamos el certificado:

openssl req -new -key server.key -out server.csr

Enter pass phrase for server.key:

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:**ES**

State or Province Name (full name) [Some-State]:**Alava**

Locality Name (eg, city) []:**Vitoria-Gasteiz**

Organization Name (eg, company) [Internet Widgits Pty Ltd]:**Egibide**

Organizational Unit Name (eg, section) []:**Informatica**

Common Name (e.g. server FQDN or YOUR name) []:**www.dominio1.com**

Email Address []:aitor@kaixo.com

Please enter the following 'extra' attributes to be sent with your certificate request

A challenge password []:linux

An optional company name []:

Quitamos la palabra de paso a la clave:

cp server.key server.key.org

openssl rsa -in server.key.org -out server.key

Enter pass phrase for server.key.org:

writing RSA key

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

Signature ok

subject=/C=ES/ST=Alava/L=Vitoria-Gasteiz/O=Egibide/OU=Informatica/CN=www.dominio1.com/e

mailAddress=aitor@kaixo.com

Getting Private key

Copiamos y renombrado el certificado y la clave generadas a /etc/apache2/ssl

cp server.crt /etc/apache2/ssl/ssl.crt

cp server.key /etc/apache2/ssl/ssl.key

3.- Colocamos por defecto el certificado en todo el servidor en el fichero
/etc/apache2/sites-available/default-ssl

SSLEngine on

SSLCertificateFile /etc/apache2/ssl/ssl.crt

profesordeinformatica.com

SSLCertificateKeyFile /etc/apache2/ssl/ssl.key

Comprobamos que tenemos el puerto 443 habilitado en /etc/apache2/ports
NameVirtualHost *:80
Listen 80

```
<IfModule mod_ssl.c>
# If you add NameVirtualHost *:443 here, you will also have to change
# the VirtualHost statement in /etc/apache2/sites-available/default-ssl
# to <VirtualHost *:443>
# Server Name Indication for SSL named virtual hosts is currently not
# supported by MSIE on Windows XP.
Listen 443
</IfModule>

<IfModule mod_gnutls.c>
Listen 443
</IfModule>
```

4.- O podemosa añadirlo a un servidor virtual:

```
<VirtualHost *:80>
DocumentRoot "/var/www/dominio2"
ServerName www.dominio2.com
<Directory "/var/www/dominio2">
allow from all
Options +Indexes
</Directory>
</VirtualHost>
<VirtualHost *:443>
DocumentRoot "/var/www/dominio2"
ServerName www.dominio2.com
<Directory "/var/www/dominio2">
allow from all
Options +Indexes
</Directory>
SSLEngine On
SSLCertificateFile /etc/apache2/ssl/ssl.crt
SSLCertificateKeyFile /etc/apache2/ssl/ssl.key
</VirtualHost>
```

5.- Añadimos el módulo de servidor seguro

sudo a2ensite default-ssl

6.- Probamos la configuración correcta del servidor.

Podemos comprobar que nuestra configuración es correcta antes de reiniciar el servidor con el siguiente comando. Si reiniciamos el servidor con un error el servicio dejará de funcionar hasta que solucionemos el problema.

apachectl configtest

7.- Reiniciamos el servidor y probamos

/etc/init.d/apache2 restart

<https://www.dominio1.com>

URL Amigables

Cuando utilizamos lenguajes de programación en el servidor y pasamos parámetros las URLs que generamos se dice que no son amigables. Es decir, que son difíciles de recordar y hacen que pierdan peso en la indexación de Google.

Ejemplo de url no amigable:

www.dominio.com/buscar.php?palabra=deporte

Ejemplo de url amigable:

www.dominio.com/buscar/deporte

Expresiones regulares:

^ Comienzo de la expresión

\$ Final de la expresión

(valor1|valor2) valor 1 o valor 2

. Encuentra cualquier cosa

* ninguno o varios

+ uno o varios

() variables que van a ser utilizadas posteriormente

(.*) cualquier contenido

(.+) Cualquier contenido que tenga al menos una palabra

Terminaciones:

[L] Última regla

[R] Indica redirección

[NC] Para indicar que no se distingan entre mayúsculas y minúsculas

Rewriterule ^palabra/(.*)\$ buscar.php?palabra=\$1

Práctica 5: Url's amigables.

La idea es que en lugar de utilizar url's no amigables como <http://localhost/buscar.php?palabra=deporte> utilicemos url's amigables como <http://localhost/palabra/deporte>.

1.- Activamos el módulo mod_rewrite

a2enmod rewrite

profesordeinformatica.com

Después editamos el fichero que tengamos en **/etc/apache2/sites-available/default**, sustituyendo

Options Indexes FollowSymLinks MultiViews
AllowOverride None

por

**Options Indexes FollowSymLinks MultiViews
AllowOverride All**

Instalamos php5

apt-get install php5-common libapache2-mod-php5

Creamos el programa /var/www/buscar.php

Estas buscando la palabra

```
<form method="get" action="buscar.php">  
<input name="palabra" value="<?php echo $_GET[palabra]; ?>">  
<input type="submit" value="buscar" />  
</form>
```

2.- Editamos el fichero /var/www/.htaccess

Options +FollowSymLinks
RewriteEngine on
RewriteRule ^palabra/(.*)\$ buscar.php?palabra=\$1 [L]
RewriteRule ^aaa.html\$ bbb.html [L]

Nota: Si en lugar de a ^palabra/(.*)\$ ponemos ^buscar/(.*)\$ no nos funciona.

3.- Realizamos la prueba en el navegador y vemos que las siguientes urls nos dan el mismo resultado:

<http://localhost/buscar.php?palabra=deporte>

<http://localhost/palabra/deporte>.

Más ayuda en:

http://www.peoplecnc.com/mod_rewrite_basico.html

<http://www.dwdpa.com/content/view/27/72/>

Ejemplo

El contenido del **.haccess** profesor de informatica.com

le decimos donde esta la base

RewriteEngine on

profesordeinformatica.com

```
Options +FollowSymLinks
Options +SymlinksIfOwnerMatch
# le decimos donde esta la base
RewriteBase /
```

```
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-f
##### html
RewriteRule ^autor$ index.php?seccion=autor&url=autor [L]
RewriteRule ^html/resultado/(.*)$ resultado.php?num=$1 [L]
RewriteRule ^html/prueballo/(.*)$ prueballo.php?num=$1 [L]
RewriteRule ^html/etiqueta/(.*)$ etiqueta.php?etiqueta=$1 [L]

RewriteRule ^html$ index.php?seccion=html&url=introduccion [L]
RewriteRule ^html/$ index.php?seccion=html&url=introduccion [L]
RewriteRule ^html/(.*)$ index.php?seccion=html&url=$1 [L]

RewriteRule ^css$ index.php?seccion=css&url=introduccion [L]
RewriteRule ^css/$ index.php?seccion=css&url=introduccion [L]
RewriteRule ^css/(.*)$ index.php?seccion=css&url=$1 [L]

RewriteRule ^html5$ index.php?seccion=html5&url=introduccion [L]
RewriteRule ^html5/$ index.php?seccion=html5&url=introduccion [L]
RewriteRule ^html5/(.*)$ index.php?seccion=html5&url=$1 [L]
```

Redirecciones 301

Como ya hemos visto anteriormente el código 301 significa que nuestro contenido se ha movido permanentemente. Esto suele ocurrir cuando cambiamos de plataforma o hacemos una migración de nuestra página web. Si no queremos perder posicionamiento o el valor del Page Rank o importancia que tienen nuestras páginas web en los buscadores, es recomendable controlar este error e indicarle a los buscadores cual va a ser el contenido de una página web.

Práctica 6: Redirecciones 301 con apache

Esto se realiza con el RedirectMatch del módulo mod_alias de apache. Vamos a tener que activar el redireccionamiento (igual que en url's amigables).

```
redirectMatch 301 /home.html http://www.google.com
```

Ejemplo:

Si queremos redireccionar:

http://www.socengine.com/seo/categorydetail.php?CAT_ID=12345

a: http://www.seomoz.org/artcat.php?CAT_ID=12345

RedirectMatch 301 /seo/categorydetail.php(.*?) http://www.seomoz.org/artcat.php\$1

Redirección example.com a www.example.com:

```
RewriteEngine On
RewriteCond %{HTTP_HOST} !^www.example.com$ [NC]
RewriteRule ^(.*)$ http://www.example.com/$1 [L,R=301]
```

Redirección de www.example.com a example.com:

```
RewriteEngine on
RewriteCond %{HTTP_HOST} ^www\.example\.com$
RewriteRule ^/?$ "http://example.com/" [R=301,L]
```

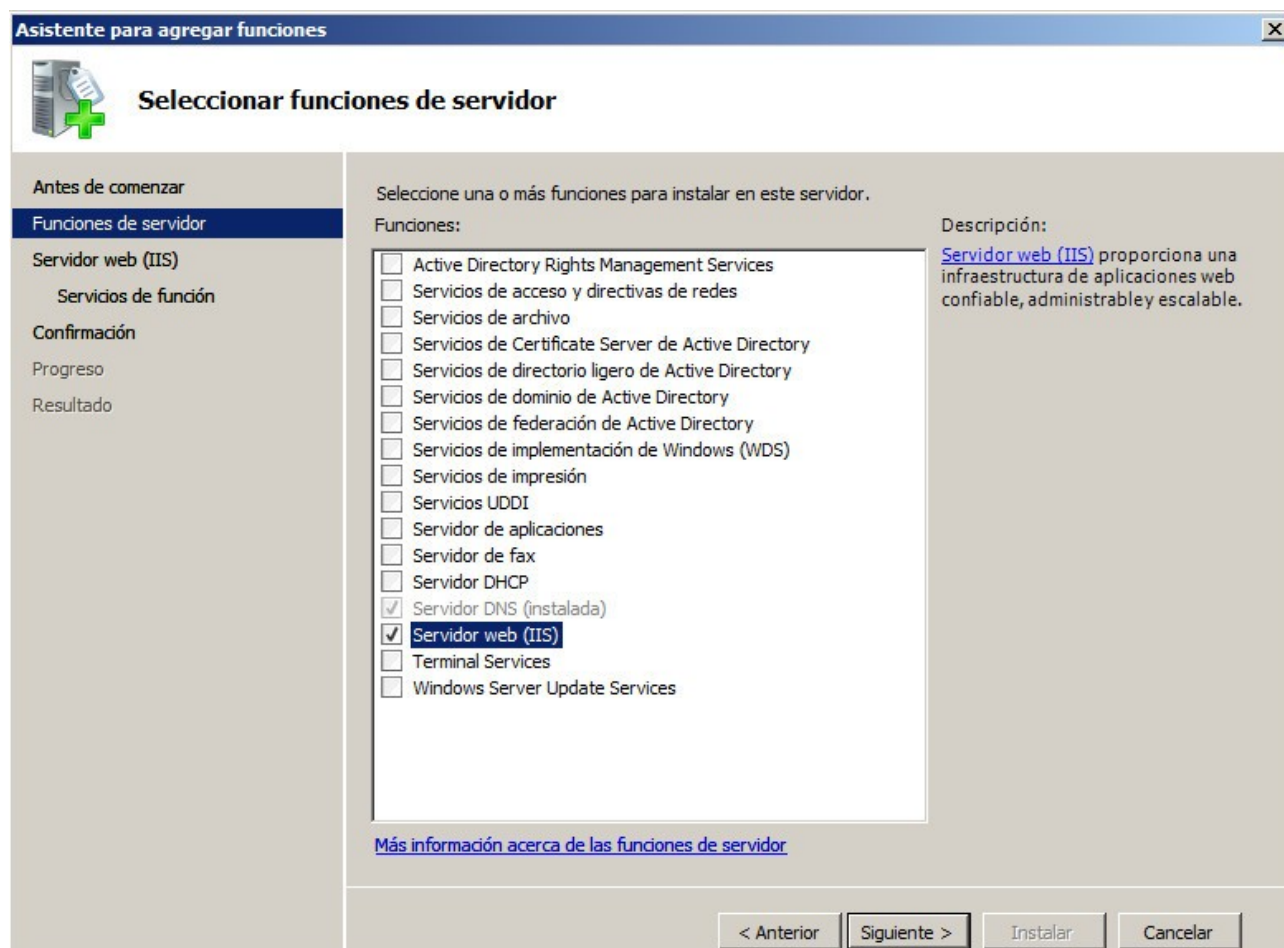
Más ejemplos en:

<http://moz.com/learn/seo/redirection>

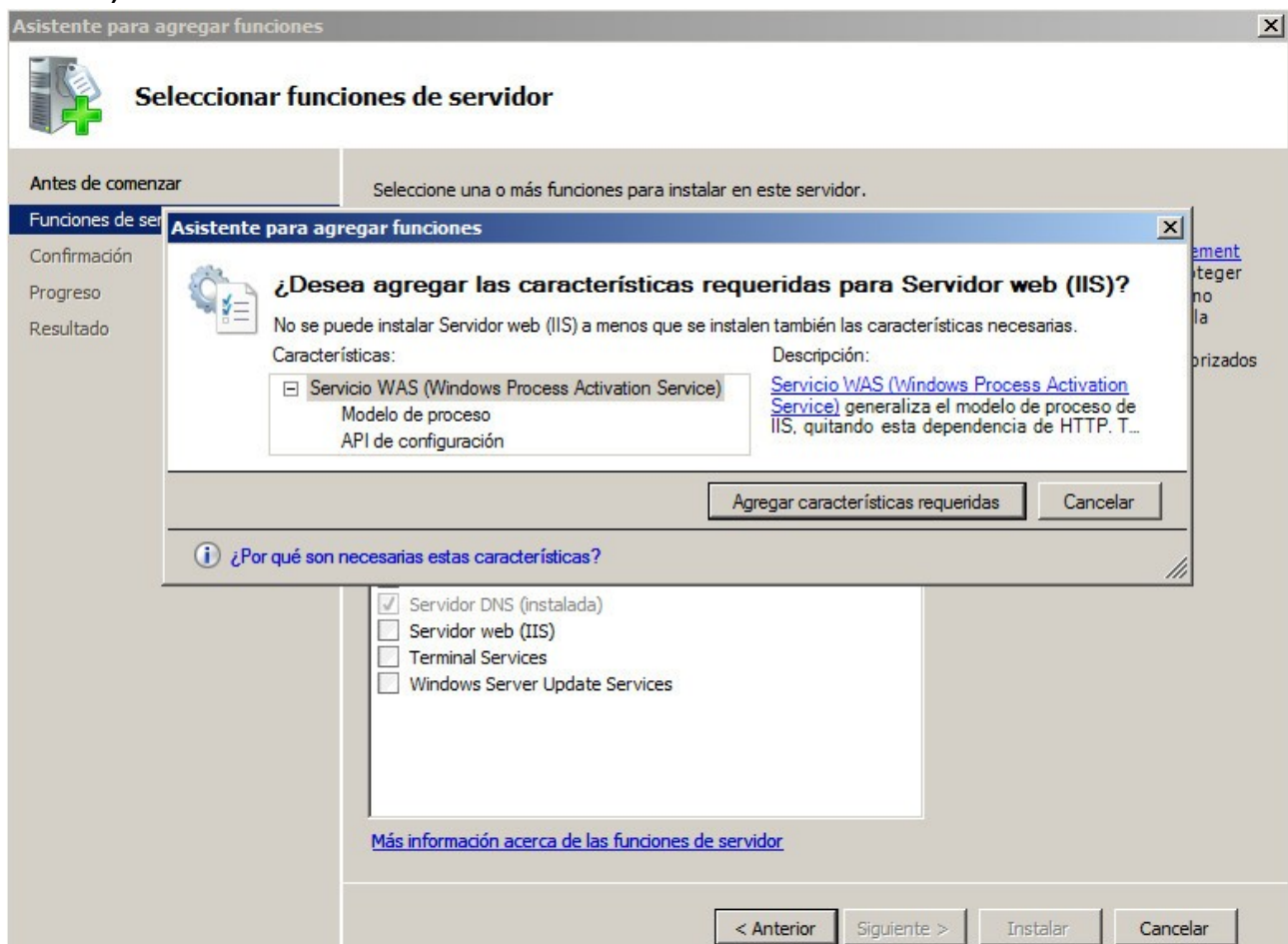
Práctica 7: IIS

IIS o Internet Information Server es el servidor web que incluyen los sistemas operativos Windows. Vamoa a realizar su instalación:

Inicio → **Herramientas administrativas** → **Administrador del servidor**→ Agregar funciones → Marcamos la casilla **Servidor IIS**



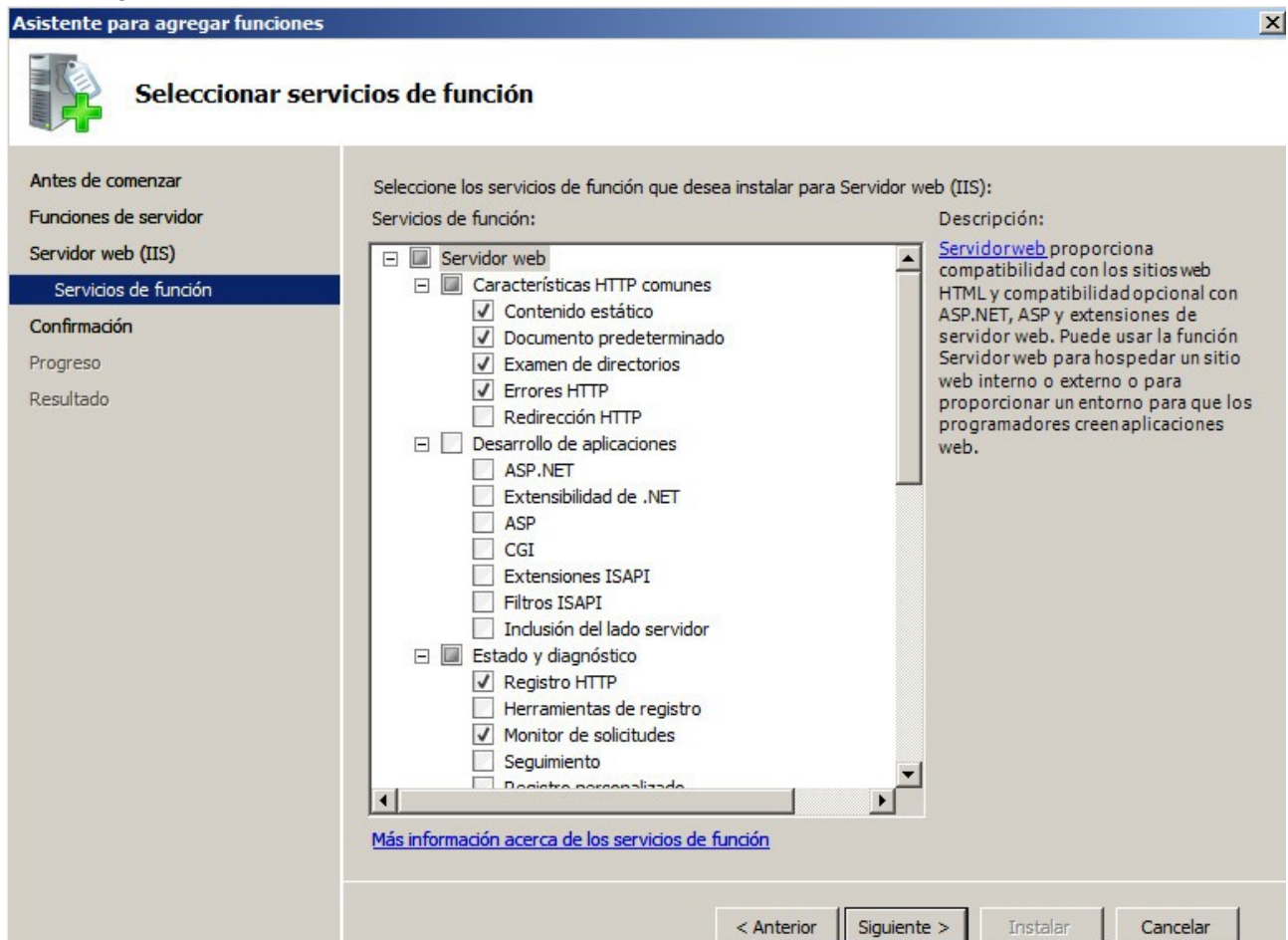
El asistente nos va a pedir añadir el servicio **WAS (Windows Process Activation Service)**



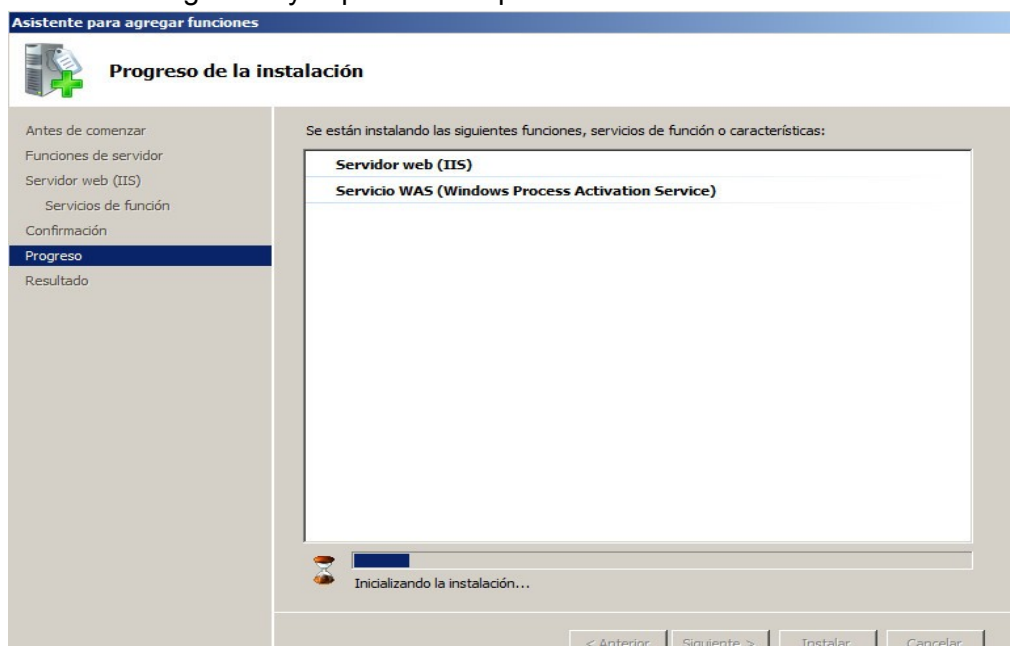
El sistema nos da una introducción de IIS. Pinchamos en **Siguiente**



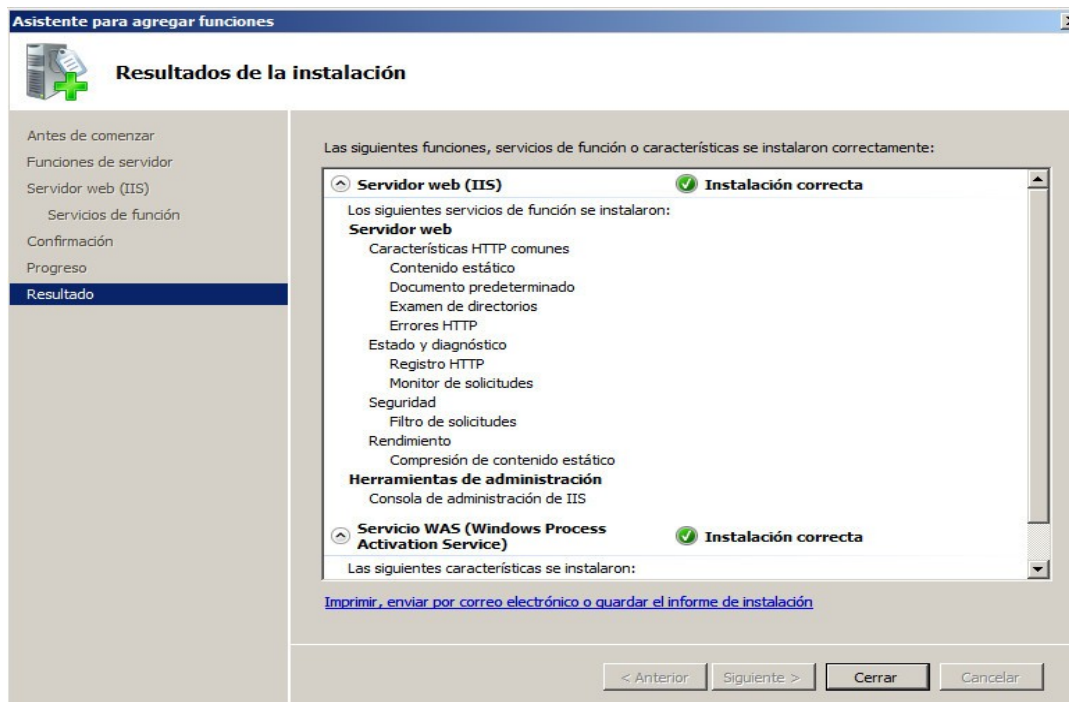
Podemos elegir los diferentes servicios que podemos añadir. Marcamos redirección HTTP, ASP (paginas web dinámicas con ASP) ASP.net (Dinamicas con ASP.net):



Pinchamos en siguiente y esperamos a que se realice la instalación:



Vemos que la instalación se ha realizado correctamente:

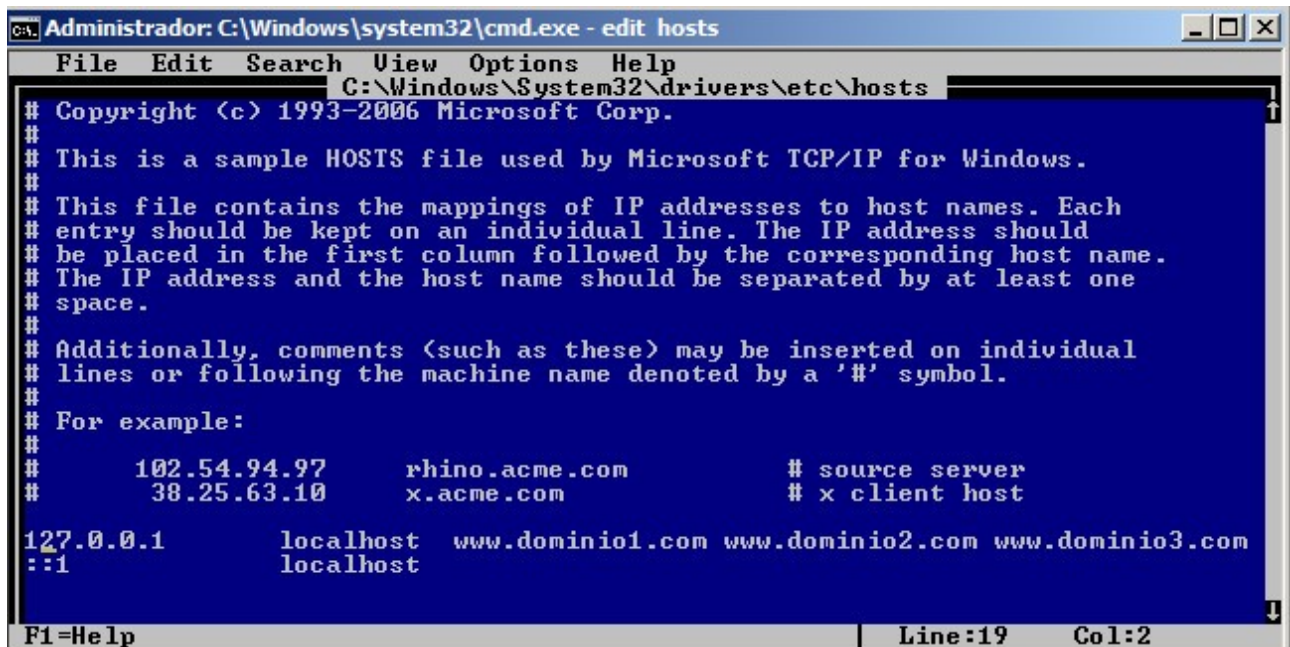


Vamos al navegador y vamos a la dirección <http://localhost>:



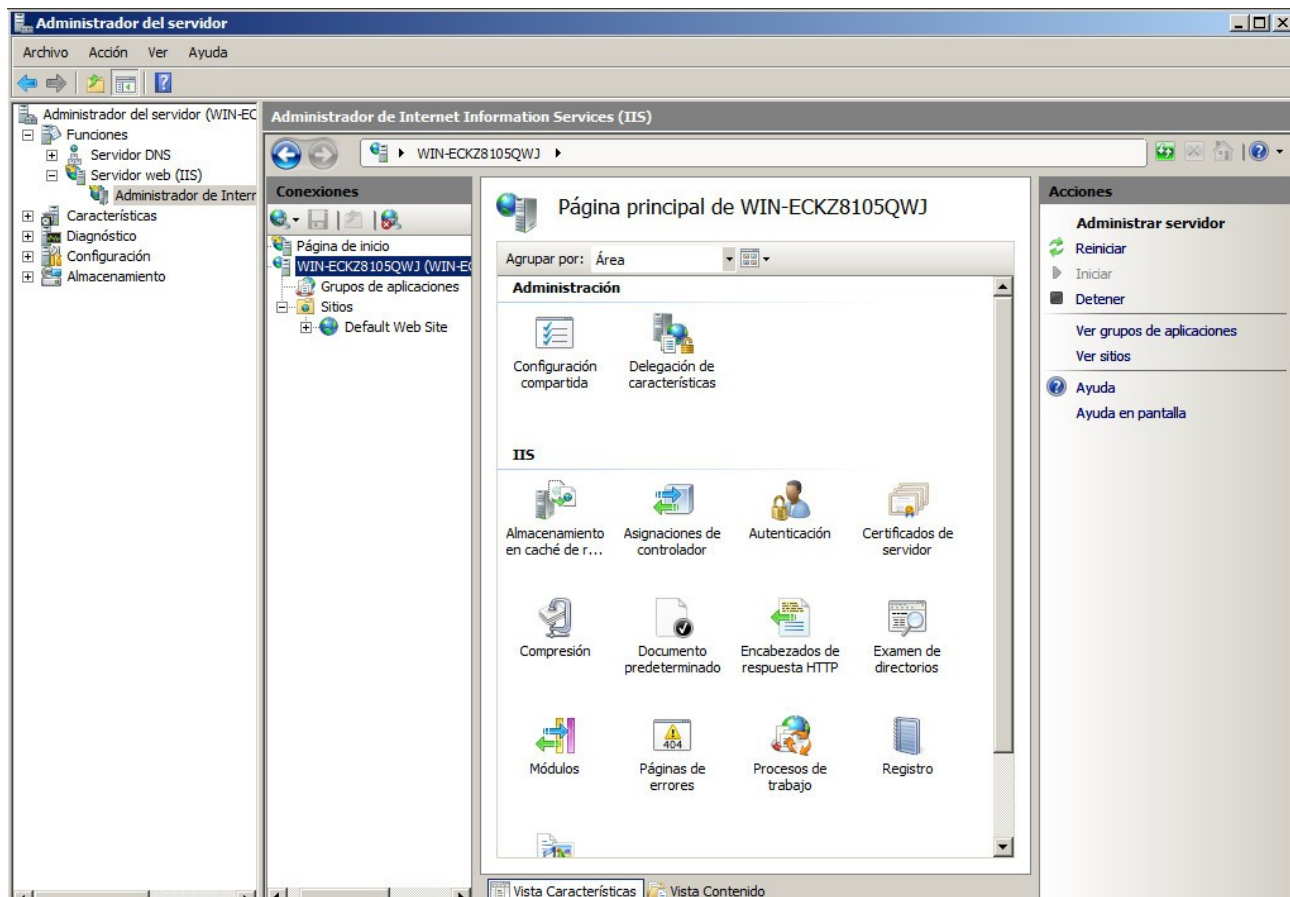
Servidor Virtual con ISS

Vamos a añadir el dominio www.dominio1.com al fichero hosts de windows (C:\Windows\System32\drivers\etc\hosts). También podríamos hacerlo añadiendo el dominio al servidor de DNS y poniendo como cliente el propio servidor.

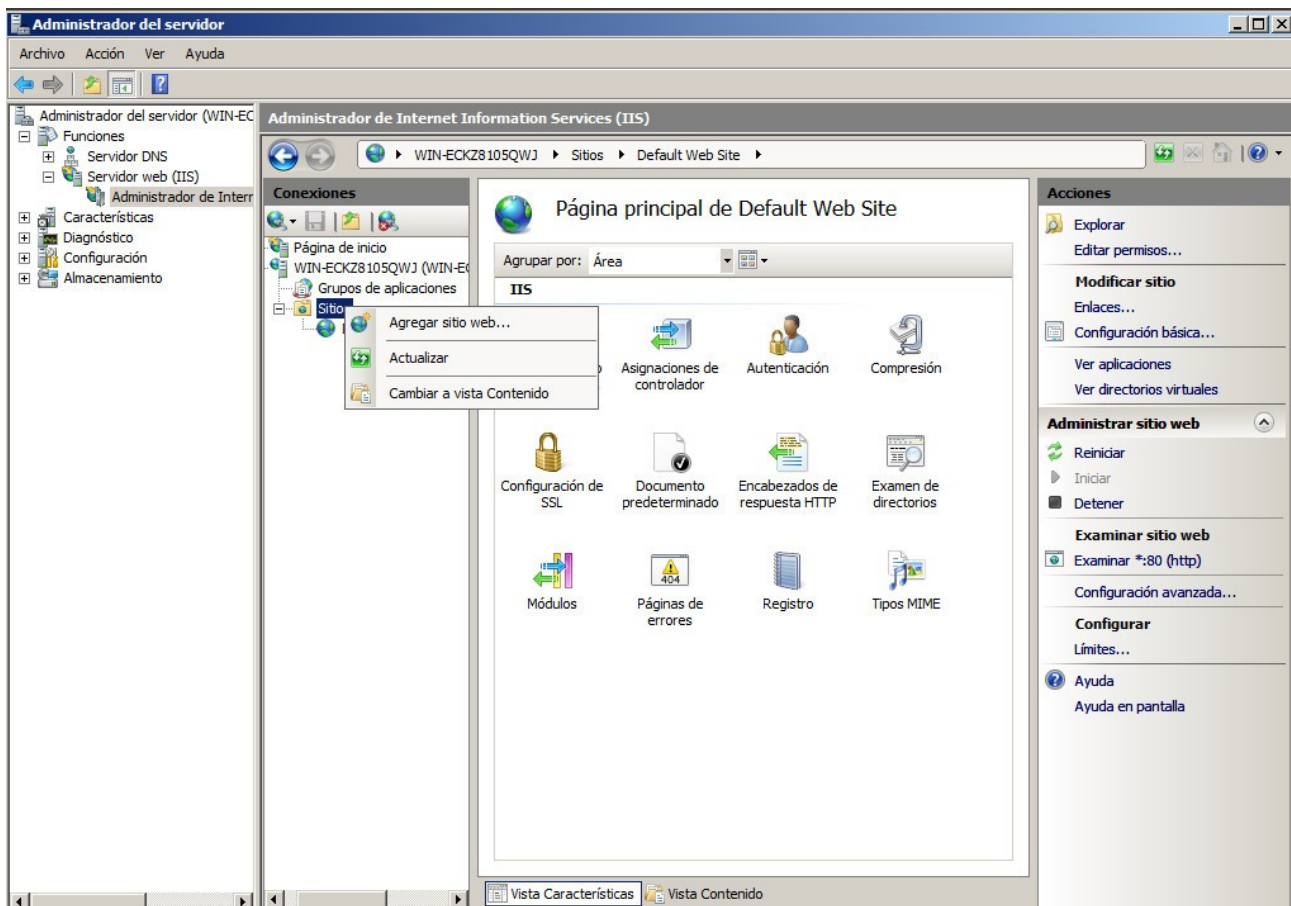


```
Administrador: C:\Windows\system32\cmd.exe - edit hosts
File Edit Search View Options Help
C:\Windows\System32\drivers\etc\hosts
# Copyright (c) 1993-2006 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com       # source server
#       38.25.63.10       x.acme.com           # x client host
127.0.0.1       localhost www.dominio1.com www.dominio2.com www.dominio3.com
::1            localhost
```

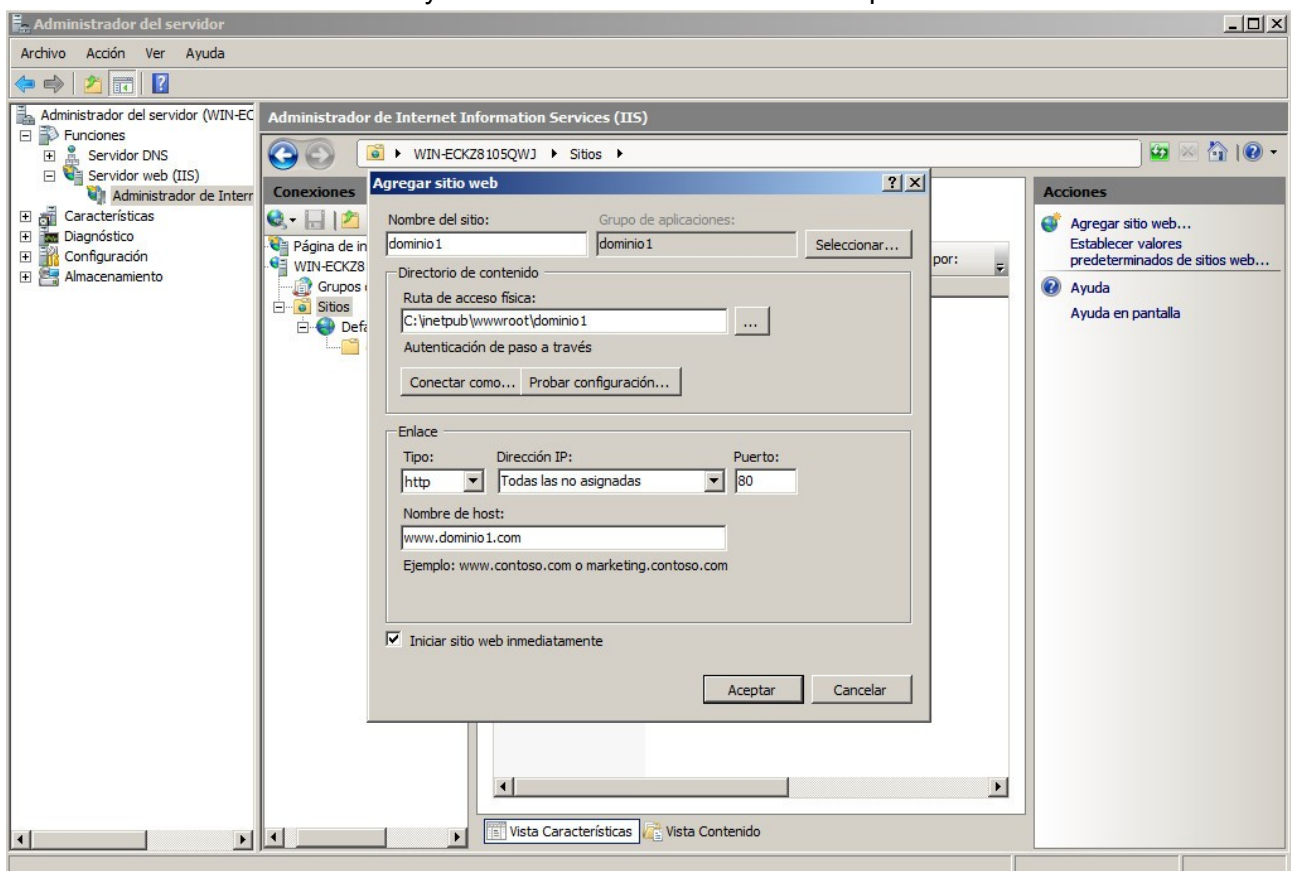
Accedemos al administrador de IIS:



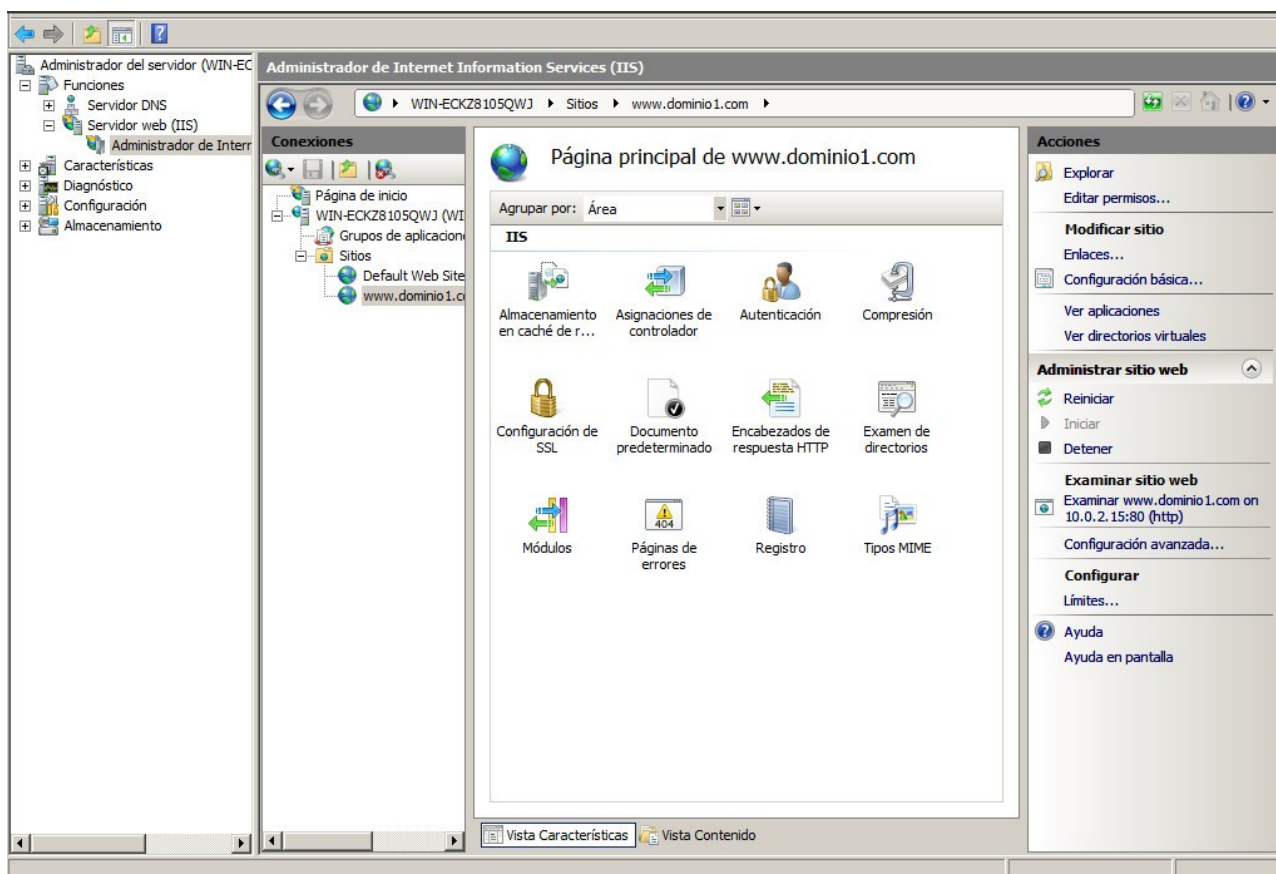
Agregamos un sitio web:



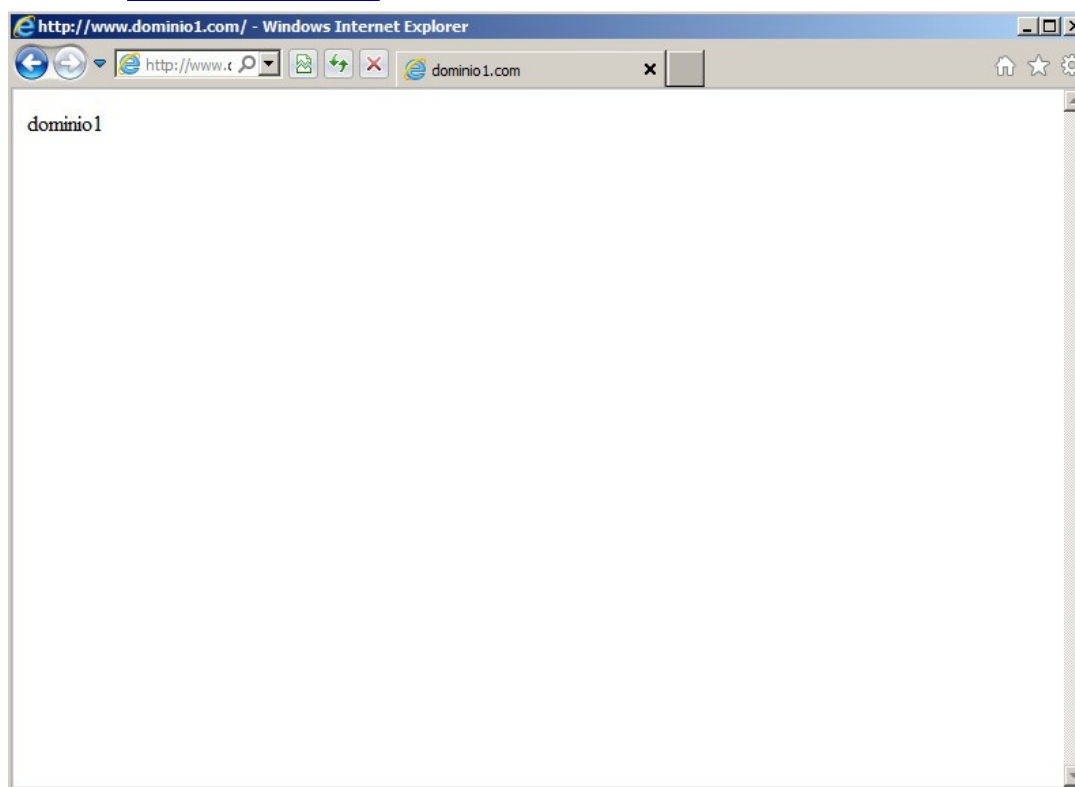
Le indicamos el dominio y donde va a estar ubicado C:\inetpub\wwwroot\dominio1:



Podemos configurar otros elementos del dominio.



Creamos la página. En Windows la página por defecto es la Default.htm y vemos el resultado en www.dominio1.com:



Ejercicios Resueltos

Ejercicio 1

Instala lynx y navega por internet.

```
apt-get install lynx  
lynx www.google.com
```

Ejercicio 2

Instala y prueba diferentes navegadores. Así como diferentes pluguins o complementos.

Ejercicio 3

Prueba el correcto funcionamiento del servidor con el comando telnet.
telnet servidor 80