

**FACULTAD DE INGENIERIA U.N.A.M.  
DIVISION DE EDUCACION CONTINUA**

**A LOS ASISTENTES A LOS CURSOS**

**L**as autoridades de la Facultad de Ingeniería, por conducto del jefe de la División de Educación Continua, otorgan una constancia de asistencia a quienes cumplan con los requisitos establecidos para cada curso.

El control de asistencia se llevará a cabo a través de la persona que le entregó las notas. Las inasistencias serán computadas por las autoridades de la División, con el fin de entregarle constancia solamente a los alumnos que tengan un mínimo de 80% de asistencias.

Pedimos a los asistentes recoger su constancia el día de la clausura. Estas se retendrán por el periodo de un año, pasado este tiempo la DECFI no se hará responsable de este documento.

Se recomienda a los asistentes participar activamente con sus ideas y experiencias, pues los cursos que ofrece la División están planeados para que los profesores expongan una tesis, pero sobre todo, para que coordinen las opiniones de todos los interesados, constituyendo verdaderos seminarios.

Es muy importante que todos los asistentes llenen y entreguen su hoja de inscripción al inicio del curso, información que servirá para integrar un directorio de asistentes, que se entregará oportunamente.

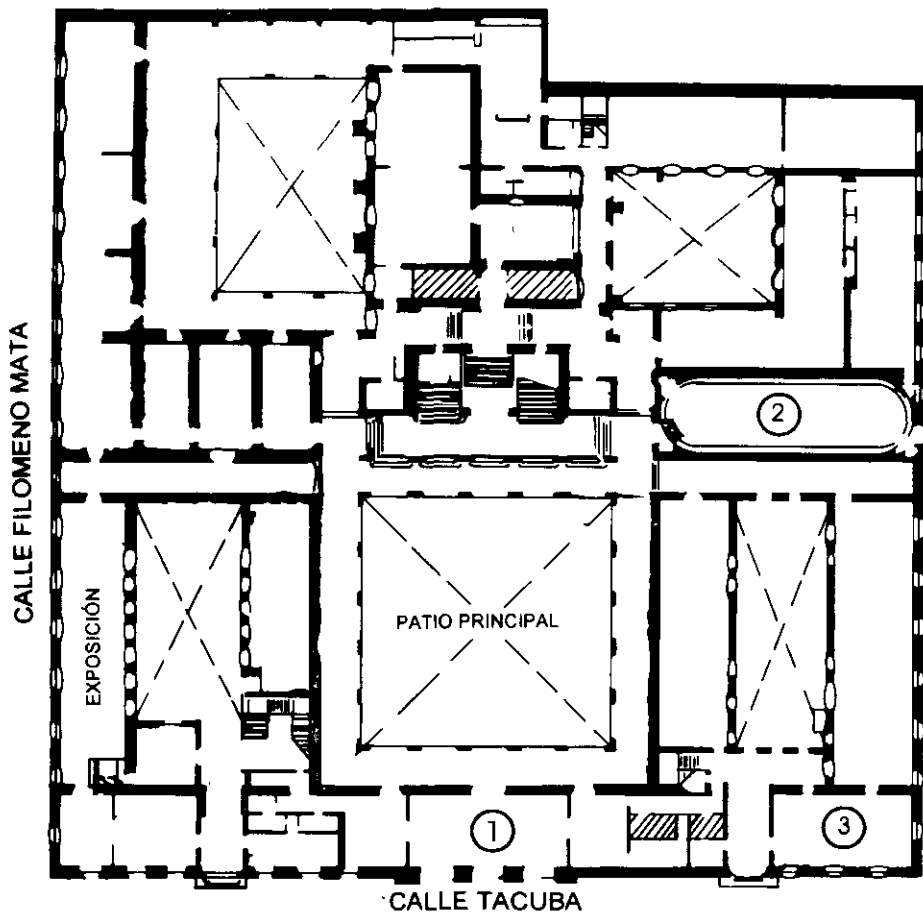
Con el objeto de mejorar los servicios que la División de Educación Continua ofrece, al final del curso deberán entregar la evaluación a través de un cuestionario diseñado para emitir juicios anónimos.

Se recomienda llenar dicha evaluación conforme los profesores impartan sus clases, a efecto de no llenar en la última sesión las evaluaciones y con esto sean más fehacientes sus apreciaciones.

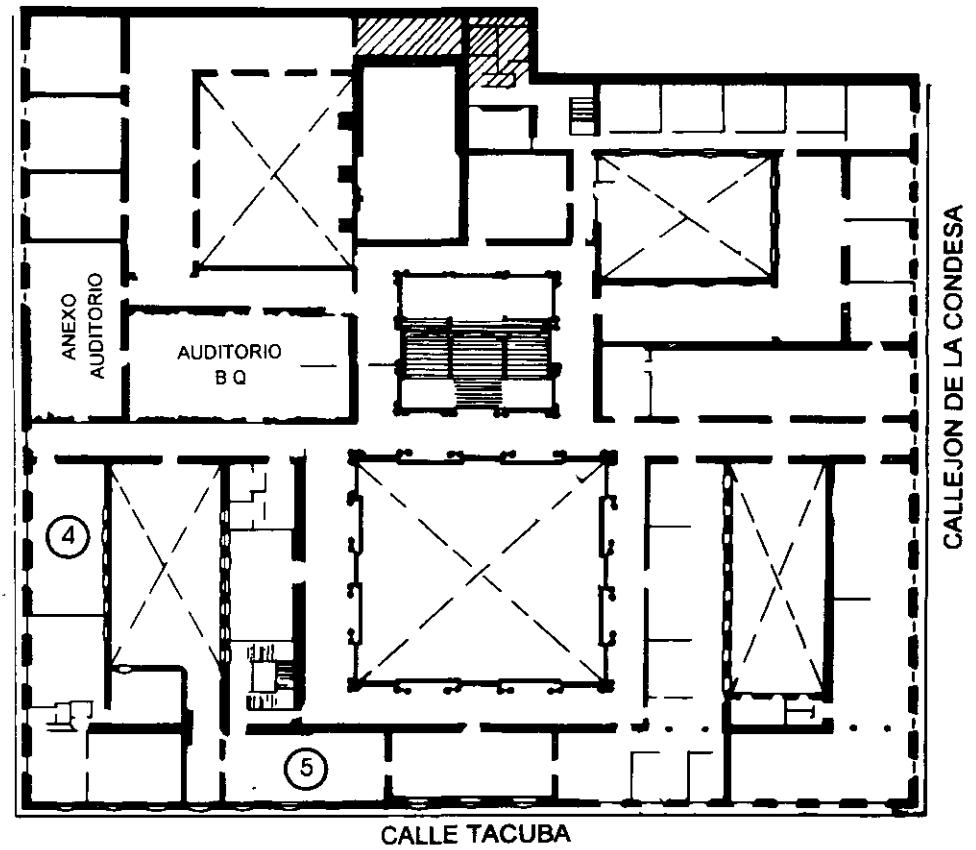
**Atentamente**

**División de Educación Continua.**

# PALACIO DE MINERIA

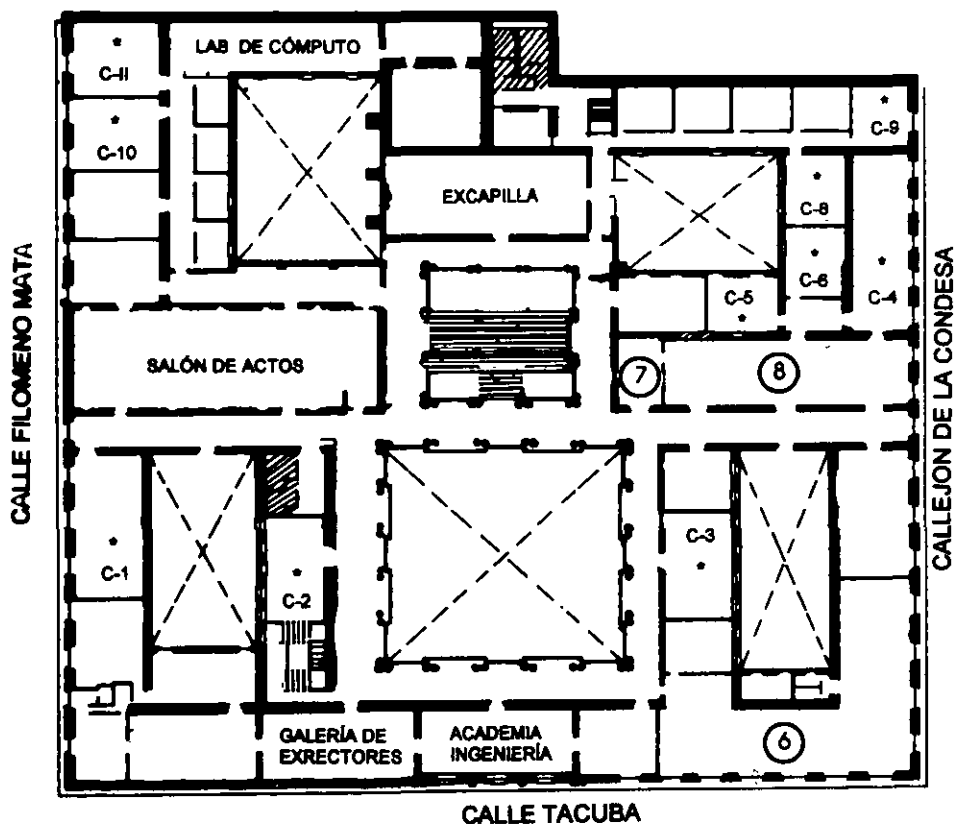


PLANTA BAJA



MEZZANINNE

# PALACIO DE MINERIA



1er. PISO

## GUÍA DE LOCALIZACIÓN

1. ACCESO
2. BIBLIOTECA HISTÓRICA
3. LIBRERÍA UNAM
4. CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN "ING. BRUNO MASCANZONI"
5. PROGRAMA DE APOYO A LA TITULACIÓN
6. OFICINAS GENERALES
7. ENTREGA DE MATERIAL Y CONTROL DE ASISTENCIA
8. SALA DE DESCANSO

SANITARIOS

\* AULAS



DIVISIÓN DE EDUCACIÓN CONTINUA  
FACULTAD DE INGENIERÍA U.N.A.M.  
CURSOS ABIERTOS

DIVISIÓN DE EDUCACIÓN CONTINUA





**FACULTAD DE INGENIERIA U.N.A.M.  
DIVISION DE EDUCACION CONTINUA**

**CURSOS A DISTANCIA**

**SERVIDORES  
DE NOMBRES DE  
DOMINIO (DNS)**

**PROFESORES: ING. CRISTOBAL CHAPITAL TIRADO  
ING. OSCAR ROBLES GARAY  
ING. FRANCISCO ARIAS CORREA  
ING. JAVIER CASTAÑON**

**Del 24 de octubre al 14 de Noviembre 1998  
(CA128)**



**Sistema de Nombres de Dominios ( DNS )**

# **Administración básica de un DNS con BIND 8**

*Ing. Cristóbal Chapital T.*

**NIC-México**



# Sistema de Nombres de Dominios ( DNS )

## Índice

### Introducción

1. Historia del Internet y el DNS.
2. Bosquejo del DNS y su estructura.
3. Características del BIND 8.
4. Archivo de configuración.
5. Tablas especiales.
6. Configuración de las tablas de un dominio.



## Sistema de Nombres de Dominios ( DNS )

### Índice

7. Configuración de un servidor secundario.
8. Resolución inversa.
9. Configuración del *resolver*.
10. Herramientas para verificar el DNS.

### Apéndices

- A. Organizaciones involucradas en Internet.
- B. Requisitos para registrar bajo *mx*.



**Sistema de Nombres de Dominios ( DNS )**

**Introducción al DNS**

# **Introducción**

*Ing. Cristóbal Chapital T.*

**NIC-México**





**Sistema de Nombres de Dominios ( DNS )**

**Historia del Internet y el DNS**

# **1. Historia del Internet y el DNS**

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Historia de Internet

- 1964 Surge un problema ¿Cómo comunicarse ?
- 1967 Planeación en Inglaterra sobre principios Operativos
- 1968 Primer resultado
- 1969 Desarrollan el proyecto RAND, MIT, UCLA
- 1971 15 nodos
- 1972 37 nodos comunicación, investigación, e-mail
- 1973 Primera Conferencia ARPANET (demostración 40 nodos)



## Sistema de Nombres de Dominios ( DNS )

### Historia de Internet

- 1974 Cambia NCP por TCP
- 1997 Acceso de diferentes grupos sociales
- 1981 Surge BITNET
- 1982 TCP/IP se vuelve un estándar, ARPANET se desliga de la parte militar
- 1984 Freenet acceso público de correo. Inicia organización de dominios (gov,mil,edu,com y net)
- 1989 Virus en red, se crea en CERT, IRC



## Sistema de Nombres de Dominios ( DNS )

### Historia de Internet

- 1989 ITESM, UNAM se conectan a Internet, guerra en el golfo y golpe de estado en la URSS
- 1990 ARPANET deja de existir, THE World primer ISP comercial, ARCHIE, WAIS, HYTELNET, GOPHER
- 1991 Se funda la ISOC, se desarrolla el WWW
- 1993 Se crea el InterNic



## Sistema de Nombres de Dominios ( DNS )

### Historia de Internet

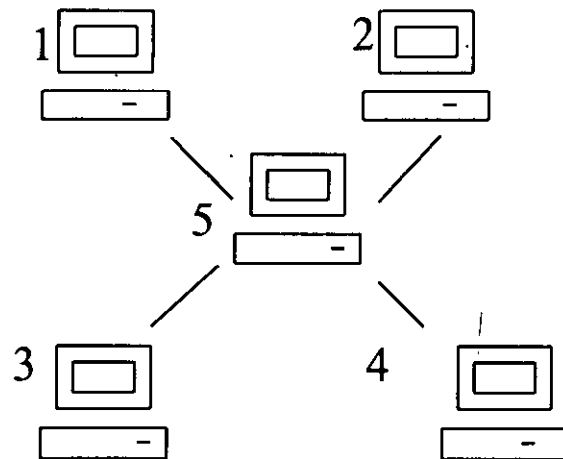
- ARPANET se convierte en el backbone de las redes locales y regionales basadas en TCP/IP, llamada Internet
- En 1988, DARPA decide acabar con el proyecto y la NSFNET reemplaza a la red ARPANET para convertirse en el nuevo backbone de Internet



# Sistema de Nombres de Dominios ( DNS )

## Historia de DNS

### Red ARPANET en los 70's

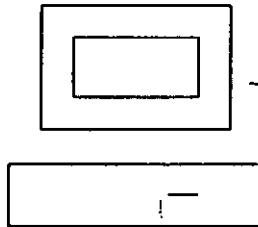
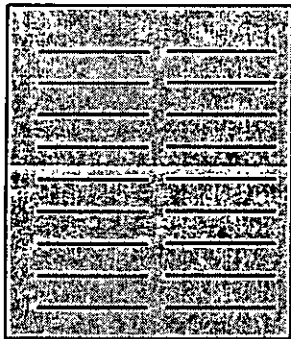


Nombre	Dirección IP:
ejemplo 1	192.0.2.1
ejemplo 2	192.0.2.2
ejemplo 3	192.0.2.3
ejemplo 4	192.0.2.4
ejemplo 5	192.0.2.5



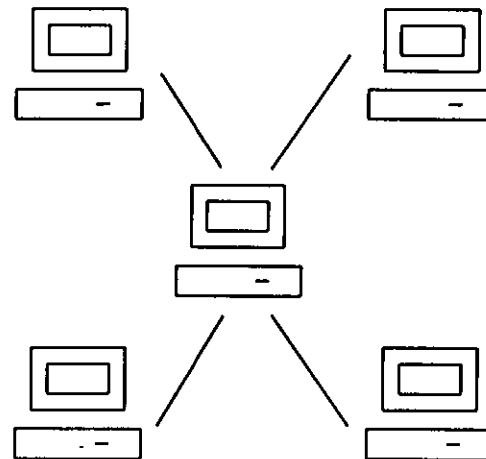
# Sistema de Nombres de Dominios ( DNS )

## Historia de DNS



SRI NIC

**Se distribuye el archivo  
HOSTS.TXT, y su  
actualización es  
periódica**



*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Historia de DNS

Paul Mockapetris se encarga de crear un nuevo concepto para utilizar el servicio de nombres de máquinas.

Evolución de los RFC's relacionados con el DNS

882

—

883



1033

—

1034

—

1035





NIC-MÉXICO

## Sistema de Nombres de Dominios ( DNS )

Organizaciones involucradas en Internet

# Organizaciones involucradas en Internet

*Ing. Cristóbal Chapital T.*

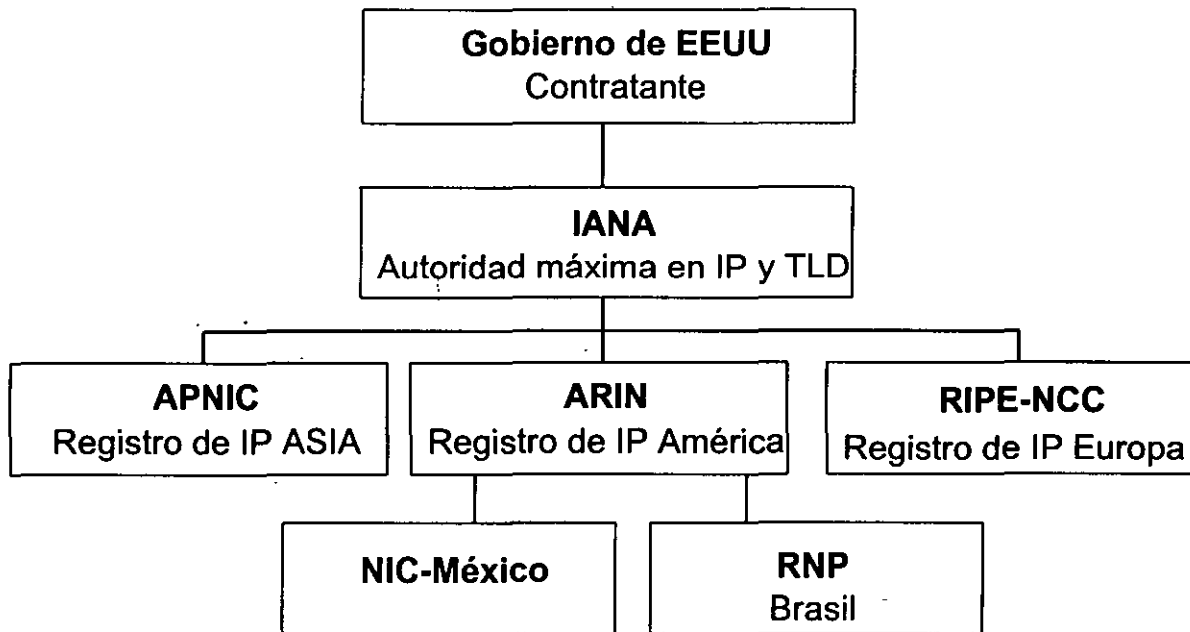
NIC-México



NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

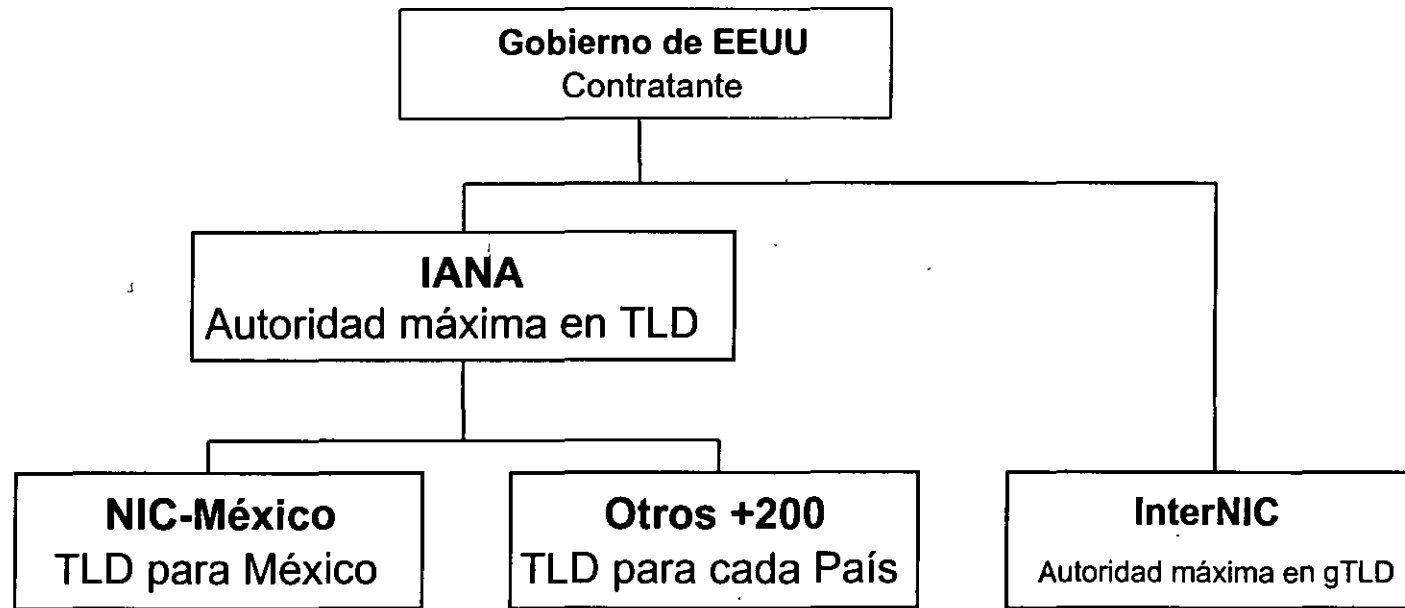
## Estructura de los registrantes de IP's





# Sistema de Nombres de Dominios ( DNS )

## Estructura de los registros de dominios



*Ing. Cristóbal Chapital T.*

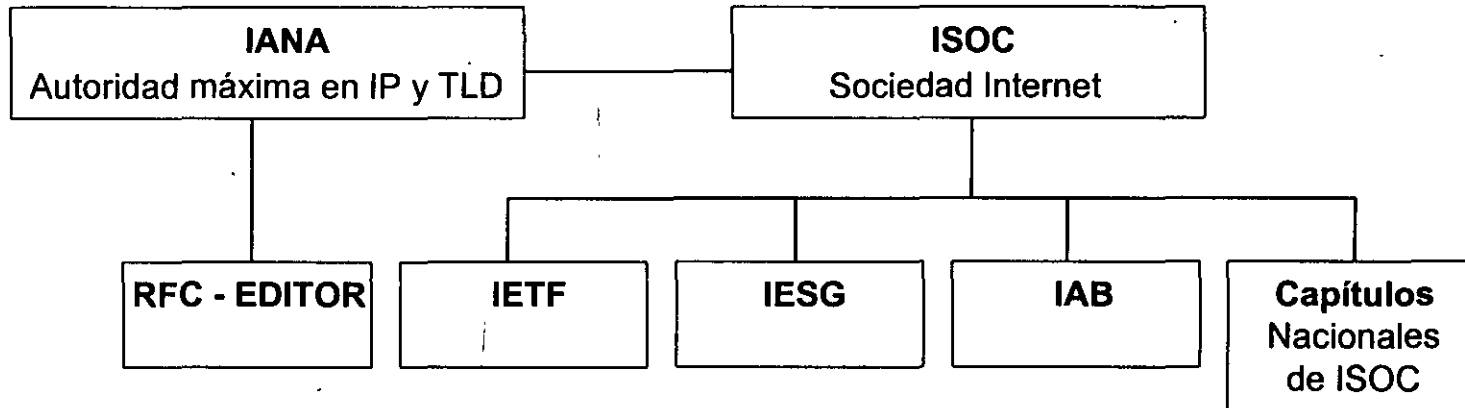
**NIC-México**



NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

Otras organizaciones involucradas en Internet



*Ing. Cristóbal Chapital T.*

**NIC-México**



NIC-MÉXICO

## Sistema de Nombres de Dominios ( DNS )

IANA

“The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols”.

The IANA is located at and operated by the Information Sciences Institute (ISI) of the University of Southern California (USC).

Jon Postel, Associate Director for Networking

*Ing. Cristóbal Chapital T.*

**NIC-México**



NIC-MÉXICO

## Sistema de Nombres de Dominios ( DNS )

ISOC



"To assure the beneficial, open evolution of the global Internet and its related internetworking technologies through leadership in standards, issues, and education."

- Miembros: 100 organizaciones y 7000 individuales en más de 150 países.
- Eventos: NDSS, INET.



## Sistema de Nombres de Dominios ( DNS )

**InterNic**



Es la organización encargada del registro y mantenimiento de los nombres de dominio ubicados bajo .com, .net, .org y .edu (gTLD). Realizan esta actividad desde 1993 por un convenio con la NSF.

- Más de 1.5 millones de registros bajo .com
- Se encuentra en Herndon, VA. Empresa Network Solutions, del grupo SAIC.

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

**ARIN**



Es la organización encargada de la asignación de direcciones IP para América y Norte de África. Realiza esta función desde el 22 de Diciembre de 1997. Anteriormente lo hacía InterNIC.

- Cobra la asignación de direcciones IP dependiendo del tamaño del bloque asignado.
- Se encuentra en Chantilly, VA.





NIC-MÉXICO

## Sistema de Nombres de Dominios ( DNS )

**NIC-México**



**ITESM**  
Campus Monterrey

Es la organización encargada del registro de nombres de dominio bajo el sufijo nacional .mx. Este centro se encuentra en el ITESM, Campus Monterrey, desde 1989.

- A partir de 1997 cobra el registro y mantenimiento de nombres de dominio. A partir del 1ro de Abril, las cuotas son:

70 USD por el registro y 35 USD por el mantenimiento

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

**ISOC-México**



“Extender el desarrollo y la disponibilidad de Internet, sus tecnologías asociadas y aplicaciones, como un fin en si mismo tanto como en términos de capacitar a organizaciones, profesiones e individuos de todo el mundo para colaborar e innovar más efectivamente en sus respectivos campos e intereses”

- Eventos: Seguridad en Internet, Internet, Foro de ISP's.
- Cambios en la mesa directiva.

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Referencias

<http://www.nic.mx>

<http://www.arin.net>

<http://www.isoc.org>

<http://www.internic.net>

<http://www.isocmex.org.mx>

<http://www.ietf.org>

<http://www.isi.edu/iana>

<http://www.dns.net/dnsrd>

*Ing. Cristóbal Chapital T.*

**NIC-México**



**Sistema de Nombres de Dominios ( DNS )**

**Bosquejo del DNS y su estructura**

# **2. Bosquejo del DNS y su estructura**



## Sistema de Nombres de Dominios ( DNS )

### Introducción

- Utilizamos DNS cada vez que hacemos uso de un servicio de Internet, como el correo electrónico, el http, telnet o ftp.
- La finalidad del DNS es facilitar la comunicación con los equipos ubicados en la red; referenciándolos por nombre en vez de direcciones numéricas.

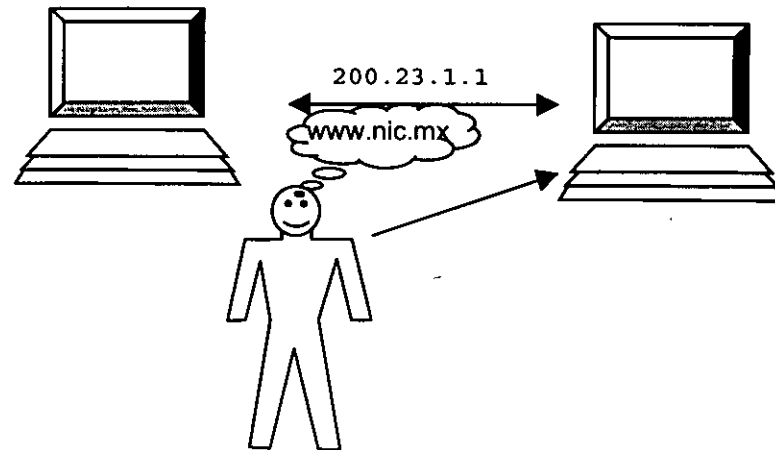


## Sistema de Nombres de Dominios ( DNS )

### Porqué utilizar DNS?

- Las máquinas se comunican con números (direcciones IP y MAC address ). Los humanos nos comunicamos con las máquinas a través de nombres.

Y esto nos resulta más fácil...





## Sistema de Nombres de Dominios ( DNS )

### Esquema General y Funcionamiento

Tres Conceptos Básicos:

- Arquitectura Cliente-Servidor.
- Base de Datos distribuida.
- Estructura jerárquica como los sistemas de archivos.



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un DNS?

- El Domain Name System es una colección de base de datos que contienen información relacionada con los nombres de dominio y direcciones de IP.
- Es un sistema que ayuda a los usuarios de Internet a recordar nombres y no números.





## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un DNS?

- El Domain Name System realiza la “traducción” de un nombre de dominios a una dirección IP (Internet Protocol). Los nombres de dominio y las direcciones de IP son únicas. Si más de una organización tienen el mismo nombre de dominio y/o IP's, ocurre una confusión en la identificación de equipos.



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un DNS?

- Por ejemplo, dos universidades una en México y la otra en Monterrey, no pueden usar el mismo nombre de dominio "escuela.edu.mx", porque el DNS no podría reconocer cual es cual y a su direcciones de IP asociadas con el nombre de dominio.



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un dominio?

- Una forma de identificar equipos (computadoras, routers, etc.)
- Jerárquico
- Único
- Correspondencia con IP's



## Sistema de Nombres de Dominios ( DNS )

¿Qué es un dominio?

Es un forma de localizar e identificar computadoras conectadas a Internet. Dos organizaciones diferentes NO pueden tener el mismo dominio.



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un dominio?

Un dominio siempre contiene dos o más componentes separados por puntos, llamados "dots".

Algunos ejemplos de dominios son:

- internic.net
- unam.mx
- nasa.gov
- utexas.edu
- mi-empresa.com.mx



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un dominio?

Bajo un dominio se puede delegar subdominios. Por ejemplo, el dominio puede ser tanto largo como la compañía desee "mi-empresa.com", se puede crear un subdominio para cada una de las oficinas regionales.

- mex. mi-empresa.com
- gdl. mi-empresa.com
- mty. mi-empresa.com



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un dominio?

Por Ejemplo, jefe.mex.mi-empresa.com describe a un host llamada jefe, en la oficina de la cd. de México de mi-empresa.

- Se pueden tener varios subdominios.
- No todos los dominios pueden contener subdominios.



## Sistema de Nombres de Dominios ( DNS )

¿Qué es un dominio?

hostname.second-level-domain.top-level-domain

hostname.third-level-domain.second-level-domain.top-level-domain

hostname.subdominio.second-level-domain.top-level-domain

hostname.subdominio.third-level-domain.second-level-domain.top-level-domain





## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un dominio?

- En Internet se usan las direcciones de IP para enviar datos. Se puede conectarse a un servidor World Wide Web por medio del nombre del dominio "www.internic.net", pero en realidad, se conecta al servidor Web con una dirección de IP la cual esta asociada al nombre del host.



## Sistema de Nombres de Dominios ( DNS )

### ¿Qué es un dominio?

- ~~Un Dominio no es:~~  
~~www.mi-empresa.com.mx~~  
~~www.mi-empresa.com~~

- Un Dominio si es:  
mi-empresa.com.mx  
mi-empresa.com ✓

- Registrar un dominio significa estar consciente de las implicaciones legales que trae consigo respecto a marcas registradas.
- Mantener el funcionamiento del nombre, así como los pagos de registro y mantenimiento del mismo.



## Sistema de Nombres de Dominios ( DNS )

IP's

- Las direcciones de IP están formadas por cuatro octetos de bits.

131.178.38.6 = 10000011 10110010 00100110  
00000110

- Los bloques de IP se clasifican por el número de bits que no cambian en el mismo.

255	131.178.38.X	=
10000011	10110010 00100110	= /24
65536	131.178.X.X	=
10000011	10110010	= /16

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

IP's

- Las direcciones de IP deben asignarse de manera racional.
- Las políticas de asignación de direcciones está descritas en el RFC 2050.
  - Se debe aplicar algún esquema de *subnetting*.
  - Promueve el uso de DHCP.
  - Recomienda el uso de *Firewalls*.

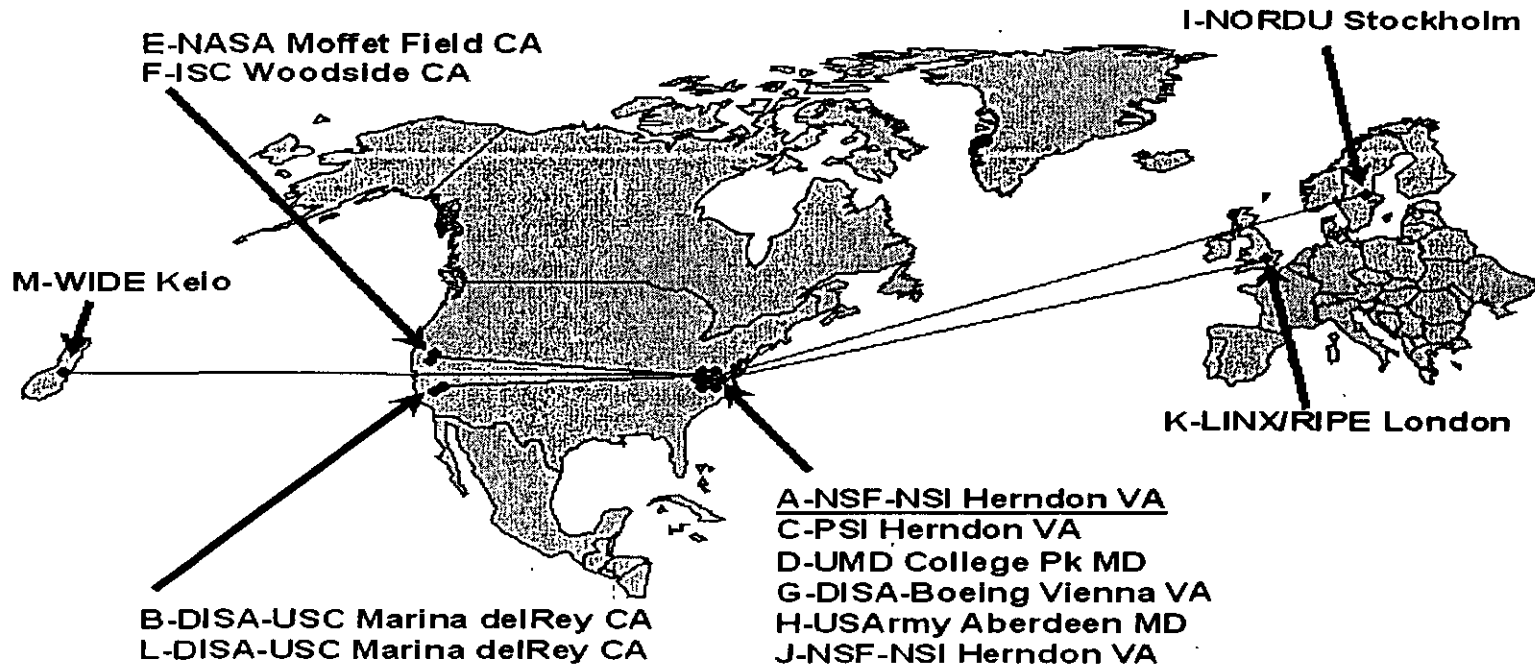


NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

## Root name servers

### DNS Root Servers



<http://www.wia.org/pub/rootsev.html>

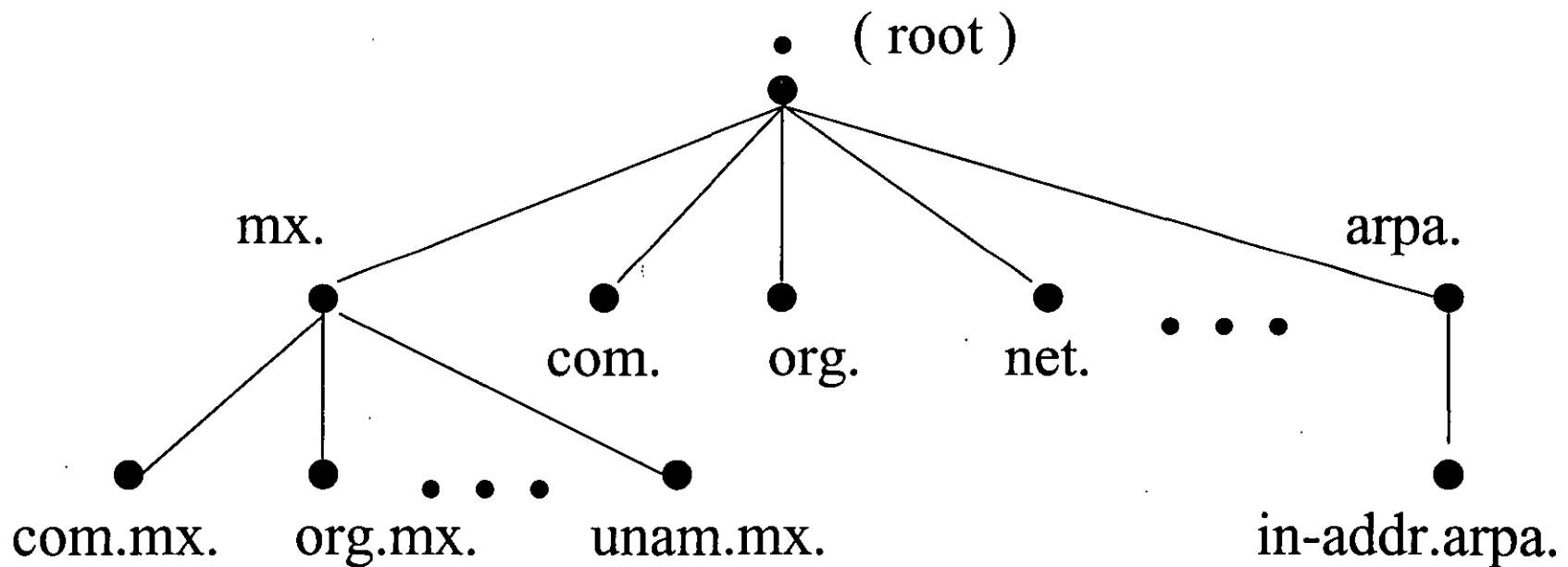
*Ing. Cristóbal Chapital T.*

**NIC-México**



# Sistema de Nombres de Dominios ( DNS )

## Estructura





## Sistema de Nombres de Dominios ( DNS )

### TLD's

<b>gTLD</b>	<b>nTLD cTLD</b>	<b>iTLD</b>	<b>sTLD</b>
.com	.de	.int	.arpa
.net	.fr		.edu
.org	.mx		.mil
	ISO3166		.gov

*Ing. Cristóbal Chapital T.*

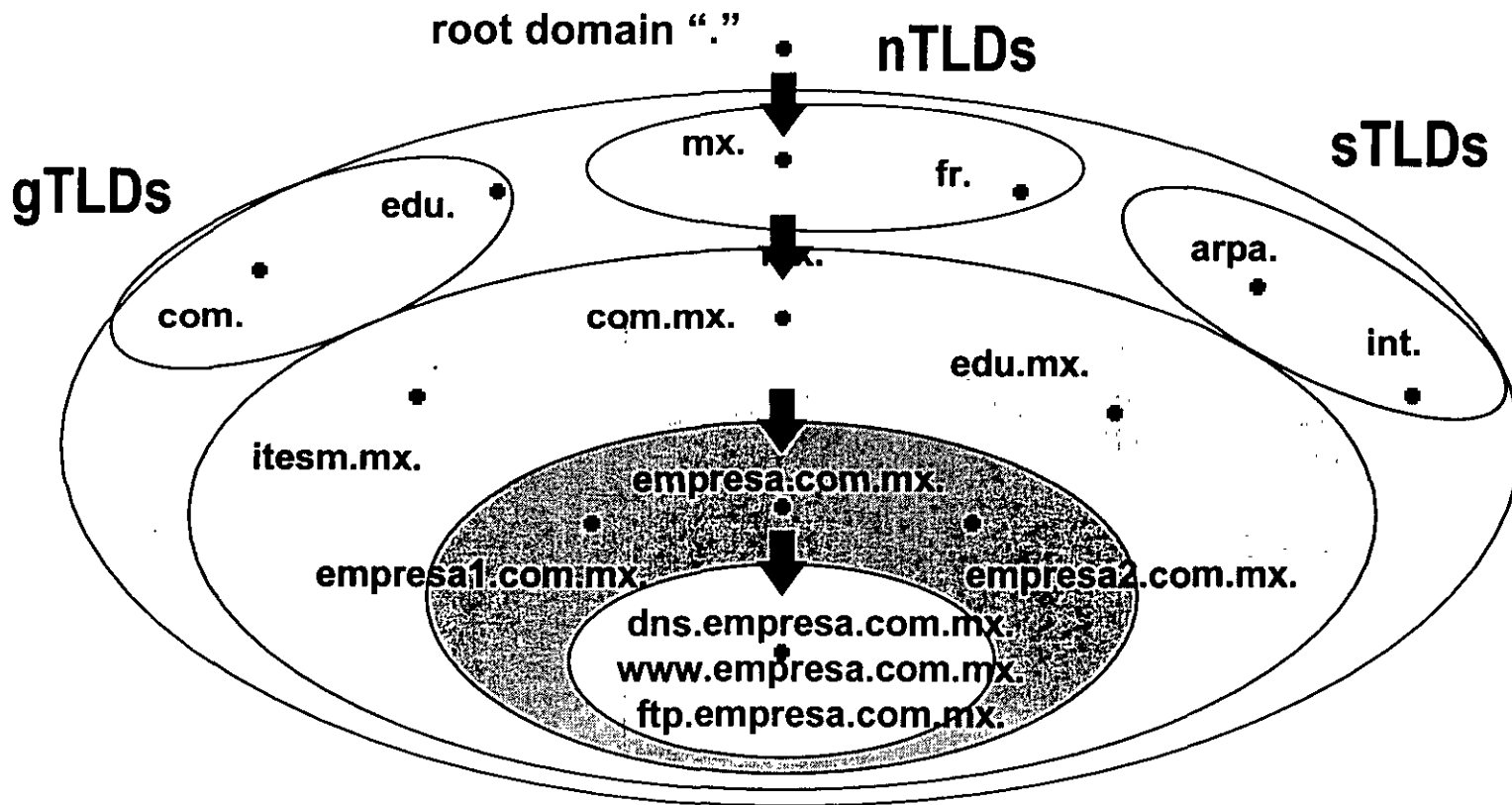
**NIC-México**



NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

## Delegación de dominios



Ing. Cristóbal Chapital T.

NIC-México



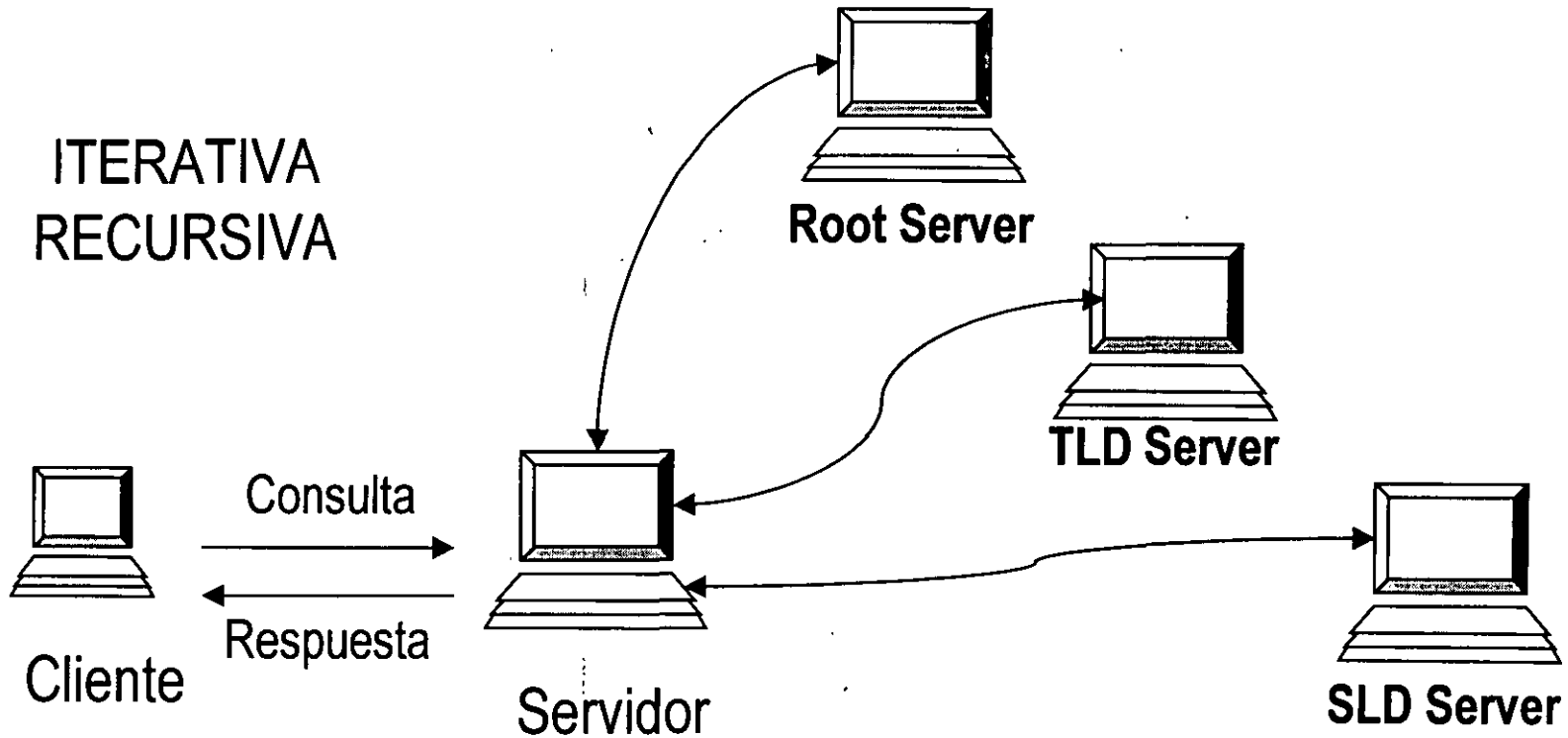


NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

## Resolución de nombres

ITERATIVA  
RECURSIVA



*Ing. Cristóbal Chapital T.*

**NIC-México**

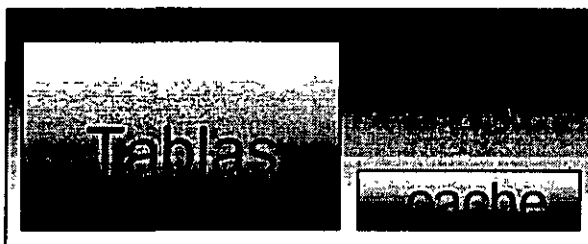


# Sistema de Nombres de Dominios ( DNS )

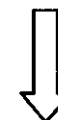
## Requerimientos para un DNS

- Capacidad mínima de procesamiento.

### *Estado Inicial*

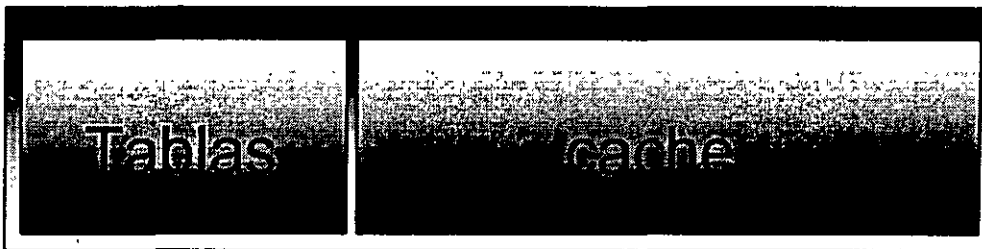


← RAM



### *... después de un tiempo*

- Memoria RAM suficiente.





Sistema de Nombres de Dominios ( DNS )

Nueva versión de BIND

# 3. Características del BIND 8

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Características del BIND 8

- Soporta DNS Dynamic Updates (RFC 2136).
- Soporta DNS Change Notification (RFC 1996).
- Control de acceso con base en direcciones de IP para *queries*, *dynamic updates* y transferencias de zona. Puede ser especificado zona por zona.
- Transferencias de zona más eficientes.



## Sistema de Nombres de Dominios ( DNS )

### Características del BIND 8

- Sintaxis del archivo de configuración más clara.
- Sistema de *logs* flexible y poderoso.
- Eficiencia mejorada especialmente para servidores que manejan miles de zonas.
- Las funciones *get\*by\*()* pueden ahora usar Sun NIS.



## Sistema de Nombres de Dominios ( DNS )

### Archivos y directorios

- Directorios
  - /var/named (el más común para las tablas)
  - /etc (para el archivo de configuración)
- Archivos relacionados
  - Ejecutables named, named-xfer, nslookup
  - Configuración /etc/named.conf
  - Tablas especiales named.local y named.root
  - Tablas tablas de zona



## Sistema de Nombres de Dominios ( DNS )

### Archivos ejecutables

- **named**
  - Nombre del programa ejecutable del servidor de nombres (/usr/sbin/named ). **UDP**
- **named-xfer**
  - Nombre del programa para transferir tablas completas de un servidor a otro (/usr/sbin/named-xfer ). **TCP**
- **nslookup**
  - Herramienta para realizar consultas a un servidor de DNS (/usr/bin/nslookup).



Sistema de Nombres de Dominios ( DNS )

/etc/named.conf

# 4. Archivo de configuración





## Sistema de Nombres de Dominios ( DNS )

### Comentarios en named.conf

/\* Comentarios como en C, continúan a través de cambios de línea, hasta encontrar \*/

// Comentarios como en C++ terminan con el fin de línea

# Comentarios como en *shell* terminan con el fin de línea



## Sistema de Nombres de Dominios ( DNS )

### Directiva *options*

```
options {
```

```
    directory "Directorio_del_named";
```

```
};
```

Path absoluto





## Sistema de Nombres de Dominios ( DNS )

### Options (ejemplo)

**named.boot** (versiones 4.\*)

directory /home/named

\*opciones de *named-xfer* y *pid-file* son fijas

**named.conf** (versiones 8.\*)

```
options {
```

```
    directory "/home/named";
```

```
};
```



## Sistema de Nombres de Dominios ( DNS )

### Directiva zone (tipo *master*)

```
zone "Nombre.de.dominio" {  
    type master;  
    file "Archivo_de_las_tablas";  
};
```

```
// Indica que el servidor tiene el original de la tabla  
// de la zona.
```



## Sistema de Nombres de Dominios ( DNS )

**Zone-master** (ejemplo)

**named.boot**

primary      mi-empresa.com.mx      mi-empresa.zone

**named.conf**

```
zone "mi-empresa.com.mx" {  
    type master;  
    file "mi-empresa.zone";  
};
```



## Sistema de Nombres de Dominios ( DNS )

### Directiva zone (tipo slave)

```
zone "Nombre.de.dominio" {  
    type slave;  
    file "Archivo_de_las_tablas";  
    masters { IP_addr; [IP_addr]; ...; };  
};  
# Indica que el servidor posee una copia de la  
# tabla original, la cual copia de alguno de los  
# servidores listados en masters.
```



## Sistema de Nombres de Dominios ( DNS )

**Zone-slave** (ejemplo)

**named. boot**

secondary    mi-empresa.com.mx    200.0.0.1    mi-empresa.zone

**named.conf**

```
zone "mi-empresa.com.mx" {  
    type slave;  
    file "mi-empresa.zone";  
    masters { 200.0.0.1; };  
};
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Directiva *zone* (tipo *hint*)

```
zone "." {
```

```
    type hint;
```

```
    file "Archivo de las tablas";
```

```
};
```

```
/* Indica las tablas que contienen las direcciones de  
IP y los nombres de los root servers.
```

```
*/
```





## Sistema de Nombres de Dominios ( DNS )

**Zone-hint** (ejemplo)

**named.boot**

cache .

named.cache

**named.conf**

zone "." {

type hint;

file "named.cache";

};



# Sistema de Nombres de Dominios ( DNS )

## named.conf (básico)

```
options {  
    directory "Directorio del named";  
};  
zone "." {  
    type hint;  
    file "Archivo root servers";  
};  
zone "0.0.127.in-addr.arpa" {  
    type master;  
    file "Archivo loopback address";  
};  
zone "nombre.del.dominio" {  
    type master;  
    file "Archivo tablas dominio";  
};
```

← Path absoluto



## Sistema de Nombres de Dominios ( DNS )

### Ejemplo de configuración

- `named.conf`

Suponer lo siguiente:

Una empresa que desea configurar el nombre de dominio **mi-empresa.com.mx**, que además acaba de adquirir un grupo de direcciones IP, un bloque /24, el **192.0.2.X** y que va a configurar algunas máquinas como la de **FTP**, la de **WWW** y ruteadores.



## Sistema de Nombres de Dominios ( DNS )

### named.conf (ejemplo)

```
options {  
    directory "/var/named";  
};  
zone "." {  
    type hint;  
    file "named.root";  
};  
zone "0.0.127.in-addr.arpa" {  
    type master;  
    file "named.local";  
};  
zone "mi-empresa.com.mx" {  
    type master;  
    file "mi-empresa.zone";  
};
```



## Sistema de Nombres de Dominios ( DNS )

### Resumiendo...

- El archivo **named.conf** contiene la información de aquellos nombres de dominio a los que va a servir nuestro DNS, tanto primarios como secundarios.
- En el archivo **named.conf** se especifican las opciones de configuración del BIND, tales como *directory*.
- Representan comentarios en el archivo de configuración: “//”, “#” y “/\*...\*/”.



**Sistema de Nombres de Dominios ( DNS )**

**named.root y named.local**

# **5. Tablas especiales**



## Sistema de Nombres de Dominios ( DNS )

named.local

```
; BIND data file for local loopback interface.
```

```
;
```

```
@          IN          SOA      Nuestro nameserver  
                                Nuestra cuenta (real)   
                                1          ; Serial  
                                360         ; Refresh  
                                300         ; Retry  
                                3600000      ; Expire  
                                3600        ; TTL  
  
1          IN          NS       Nuestro nameserver  
  
1          IN          PTR      localhost.
```

Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

**named.local** (ejemplo)

```
; BIND data file for local loopback interface.
;
@      IN      SOA      dns.mi-empresa.com.mx.
                               mi-cuenta.mi-empresa.com.mx. (
                               1          ; Serial
                               360        ; Refresh
                               300        ; Retry
                               3600000    ; Expire
                               3600 )    ; TTL
                               IN      NS      dns.mi-empresa.com.mx.
1      IN      PTR     localhost.
```





## Sistema de Nombres de Dominios ( DNS )

### Archivos vía FTP

- Ejemplos
  - `ftp://ftp.nic.mx/pub/named.conf`
- Originales
  - `ftp://ftp.nic.mx/pub/named.root`
  - `ftp://ftp.nic.mx/pub/named.local`



## Sistema de Nombres de Dominios ( DNS )

Configuración de tablas de zona

# 6. Configuración de las tablas de un dominio



## Sistema de Nombres de Dominios ( DNS )

### Configuración de tablas de zona

- Estructura de los archivos de zona:

SOA

Name Servers

Máquinas



## Sistema de Nombres de Dominios ( DNS )

**mi-empresa.zone** (ejemplo 1)

```
mi-empresa.com.mx.  IN   SOA   dns.mi-empresa.com.mx.
                    mi-cuenta.mi-empresa.com.mx. (
                    1      ; serial
                    10800 ; refresh
                    3600  ; retry
                    604800 ; expire
                    86400 ) ; ttl

; ns
mi-empresa.com.mx.  IN   NS     dns.mi-empresa.com.mx.
; máquinas
dns                 IN   A      192.0.2.1
```



NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

...y delegación de dominios

SOA

Name Servers

Subdominios

Máquinas

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

**mi-empresa.zone** (ejemplo 2)

```
mi-empresa.com.mx.  IN   SOA   dns.mi-empresa.com.mx.
                    1      ; serial
                    10800  ; refresh
                    3600   ; retry
                    604800 ; expire
                    86400 ) ; ttl

mi-empresa.com.mx.  IN   NS    dns.mi-empresa.com.mx.
;
conta               IN   NS    odns.conta.mi-empresa.com.mx.
;
odns.conta         IN   A     192.0.2.10
dns                IN   A     192.0.2.1
```



## Sistema de Nombres de Dominios ( DNS )

### Registro SOA

- **Serial.** Número de versión de la tabla.
- **Refresh.** Intervalo de tiempo al final del cual el NS secundario debe copiar la tabla del NS primario.
- **Retry.** Tiempo que el secundario debe esperar para reintentar la conexión al primario en caso de que falle la conexión al hacer el *refresh*.
- **Expire.** Tiempo después del cual, sino se ha logrado hacer el *refresh*, se desecha la tabla; el NS secundario deja de responder a requisiciones sobre el dominio al que se refiere la tabla.
- **TTL.** Intervalo de tiempo asociado a cada uno de los RR's de la tabla. Cuando un NS responde un *query*, envía en la respuesta este TTL para indicarle al NS que “pregunta”, por cuanto tiempo debe guardar en memoria este registro.



# Sistema de Nombres de Dominios ( DNS )

## Registro SOA

@

IN

SOA

*Nuestro nameserver*

*E-mail del responsable*

1 ; Serial  
360 ; Refresh  
300 ; Retry  
3600000 ; Expire  
3600 ■ ; TTL





## Sistema de Nombres de Dominios ( DNS )

### Registro SOA (ejemplo)

```
@      IN      SOA      dns.mi-empresa.com.mx.  
                        mi-cuenta.mi-empresa.com.mx. (  
                            1          ; Serial  
                            360        ; Refresh  
                            300        ; Retry  
                            3600000    ; Expire  
                            3600 )     ; TTL
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Elemento \$ORIGIN

Define el nombre de dominio relativo que se agregará al final de los nombres que no terminen en "." (punto).

Formato:

\$ORIGIN sub.dominio.

\$ORIGIN com.mx.



## Sistema de Nombres de Dominios ( DNS )

### Registro NS

Define el servidor de nombres para un dominio.

Formato del registro:

[dominio] [ttl] IN NS servidor-de-nombres

```
mi-empresa.com.mx.  IN   NS   dns.mi-empresa.com.mx.  
                    IN   NS   dns2.mi-empresa.com.mx.
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Registro A

Define una dirección para un nombre de máquina.

Formato del registro:

[nombre-máquina] [ttl] IN A dirección.de.IP

dns.mi-empresa.com.mx.	IN	A	192.0.2.1
dns2	IN	A	192.0.2.2
	IN	A	192.0.2.6



## Sistema de Nombres de Dominios ( DNS )

### Registro CNAME

Define un ALIAS para una máquina.

Formato del registro:

[alias-máquina] [ttl] IN CNAME nombre-máquina

ftp.mi-empresa.com.mx. IN CNAME www

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Registro HINFO

Define el tipo de CPU y sistema operativo de una máquina.

Formato del registro:

```
[nombre-máquina] [ttl] IN HINFO "CPU" "OS"
```

```
dns2 IN HINFO "Sparc Station 10" "Solaris 2.5.1"
```



## Sistema de Nombres de Dominios ( DNS )

### Registro TXT

Define información para una máquina.

Formato del registro:

```
[nombre-máquina] [ttl] IN TXT "Información"
```

```
dns2                IN TXT    "Site, piso 2"
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

**mi-empresa.zone** (ejemplo 3)

```
mi-empresa.com.mx.  IN  SOA  dns
                                mi-cuenta.mi-empresa.com.mx. (
                                1          ; serial
                                10800      ; refresh
                                3600       ; retry
                                604800     ; expire
                                86400 )    ; ttl

mi-empresa.com.mx.  IN  NS   dns.mi-empresa.com.mx.
dns.mi-empresa.com.mx.  IN  A   192.0.2.1
www                  IN  A   192.0.2.3
                    IN  HINFO "Sparc 10", "Solaris 2.6"
ftp.mi-empresa.com.mx. IN  CNAME www
```

*Ing. Cristóbal Chapital T.*

**NIC-México**





## Sistema de Nombres de Dominios ( DNS )

**mi-empresa.zone** (ejemplo 4)

```
mi-empresa.com.mx.  IN  SOA  dns  mi-cuenta.mi-empresa.com.mx. (
                    1      ; serial
                    10800  ; refresh
                    3600   ; retry
                    604800 ; expire
                    86400 ) ; ttl

mi-empresa.com.mx.  IN  NS  dns.mi-empresa.com.mx.
conta               IN  NS  odns.conta.mi-empresa.com.mx.
odns.conta         IN  A   192.0.2.10
dns                 IN  A   192.0.2.1
www                 IN  A   192.0.2.3
                   IN  HINFO "Sparc Station 10", "Solaris 2.6"
ftp.mi-empresa.com.mx. IN CNAME www
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Resumiendo...

- El elemento \$ORIGIN define el subdominio con el cual se trabaja en cada archivo. Este se “arrastra” desde el archivo **named.conf**.
- La estructura definida para las tablas es una recomendación, el orden de los segmentos es irrelevante.
- En los archivos de tablas (excepto en el registro **SOA**), todos los nombres de dominios o máquinas que **no** finalicen con punto “.” el BIND les concatena el \$ORIGIN.



## Sistema de Nombres de Dominios ( DNS )

### Resumiendo...

- La versión de las tablas está indicada por el *serial* del **SOA record**, si la tabla sufre alguna modificación es necesario actualizar también el *serial*.
- Existen elementos relativos a dominios y elementos relativos a máquinas.
- El elemento NS es relativo a dominios.
- Los relativos a máquinas son A, TXT, HINFO, CNAME.
- HINFO y TXT dan elegancia y claridad a las tablas pero inseguridad en las máquinas de este dominio.



## Sistema de Nombres de Dominios ( DNS )

### Herramientas

- NIC-México
  - <http://www.nic.mx/cgi/dns>
- Otros
  - <http://simmons.starkville.ms.us/dns/tools.html>



## Sistema de Nombres de Dominios ( DNS )

Configuración de *nameservers* secundarios

# 7. Configuración de un servidor secundario

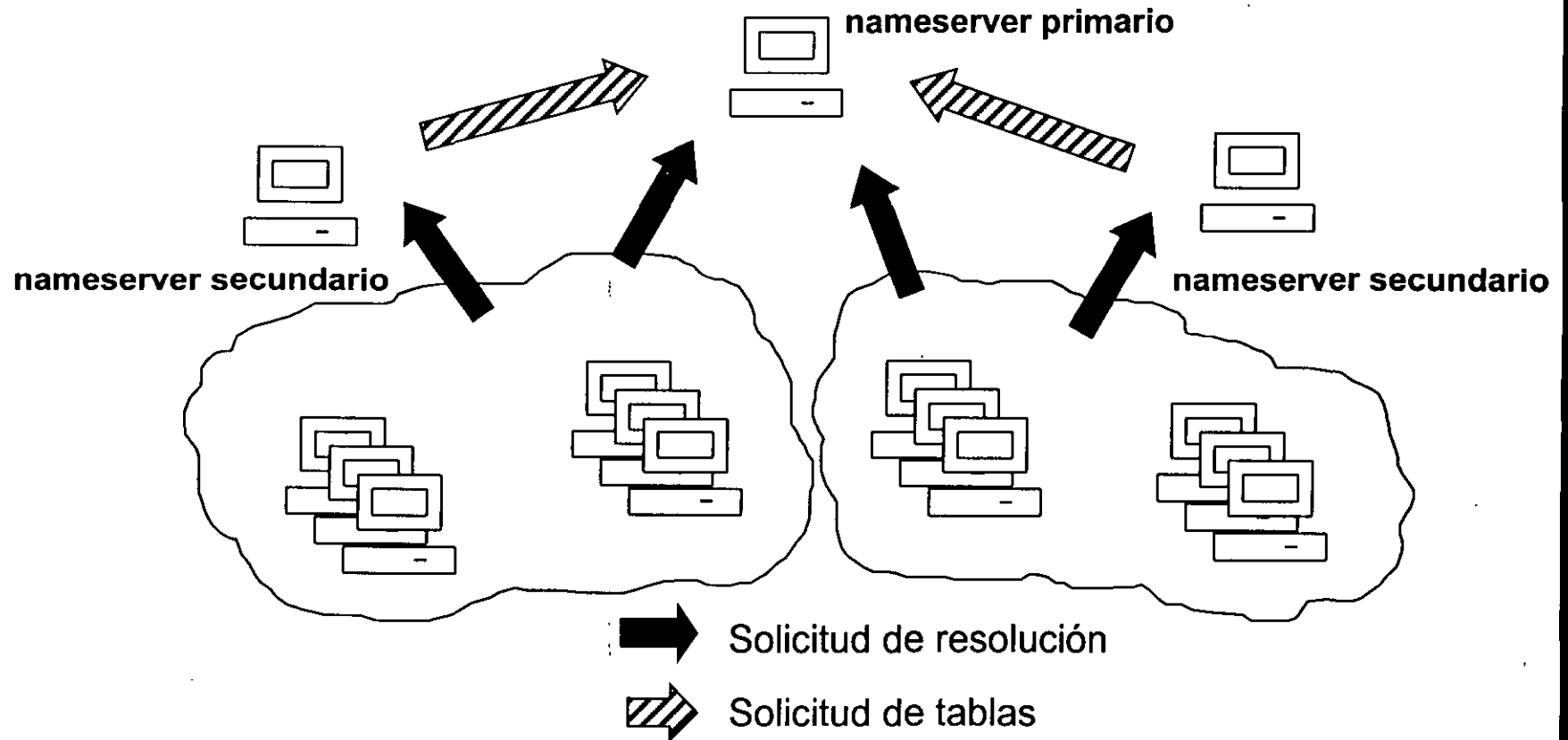
Ing. Cristóbal Chapital T.

NIC-México



# Sistema de Nombres de Dominios ( DNS )

Configuración de *nameservers* secundarios

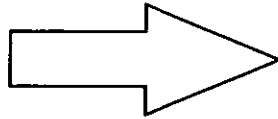




# Sistema de Nombres de Dominios ( DNS )

## Registro SOA ( NS primario )

```
;  
; Start of authority for mi-empresa.com.zone  
;  
@      IN          SOA          dns.mi-empresa.com.mx.  
                                mi-cuenta.mi-empresa.com.mx. (  
                                1          ; Serial  
                                360        ; Refresh  
                                300        ; Retry  
                                3600000    ; Expire  
                                3600 )     ; TTL  
  
      IN          NS          dns.mi-empresa.com.mx.  
      IN          NS          dns.otra-empresa.com.mx.
```



◀ Servidor secundario

Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

### named.conf ( NS secundario )

```
options {  
    directory "/var/named";  
};  
zone "." {  
    type hint;  
    file "named.root";  
};  
zone "0.0.127.in-addr.arpa" {  
    type master;  
    file "named.local";  
};  
zone "mi-empresa.com.mx" {  
    type slave;  
    file "mi-empresa.zone";  
    masters { 192.0.2.1; };  
};
```

Dirección de IP  
del NS primario

Ing. Cristóbal Chapital T.

NIC-México





## Sistema de Nombres de Dominios ( DNS )

### Resumiendo...

- Es posible mantener más de un servidor de nombres para cada dominio. A éstos se les conoce como *nameservers* secundarios y pueden ser “oficiales” o “no oficiales”, dependiendo de si están definidos o no con un registro *NS* en la tabla de zona del dominio en el servidor primario.
- La directiva que indica que un servidor es secundario para un dominio es “**type slave;**” y ésta se indica en el **named.conf**.
- El servidor de nombres secundarios solamente copia las tablas cuando la versión ( *serial* ) de las tablas en el primario es mayor que la que él tiene.



NIC-MÉXICO

**Sistema de Nombres de Dominios ( DNS )**

**Resolución inversa**

# **8. Resolución inversa**

*Ing. Cristóbal Chapital T.*

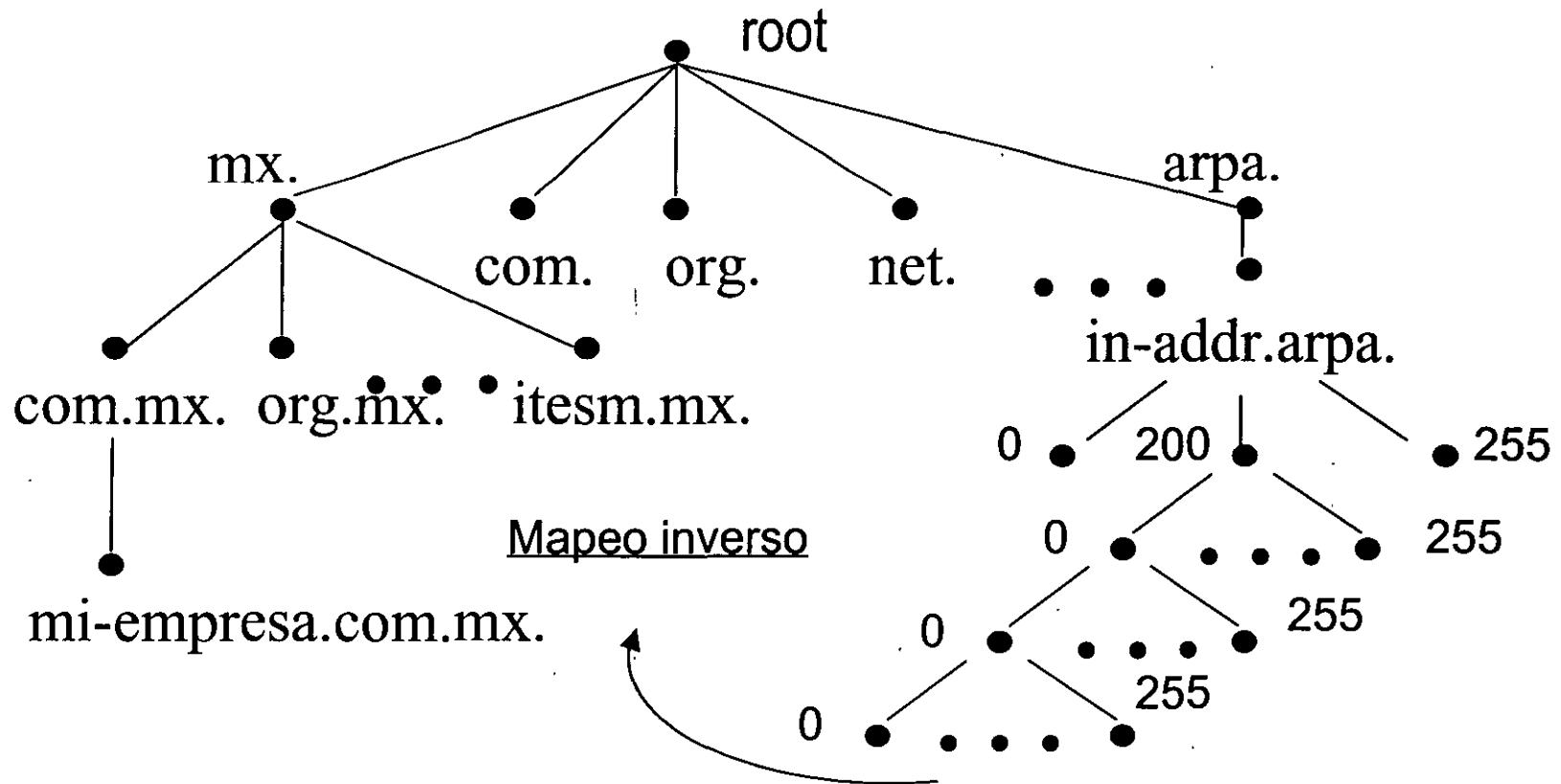
**NIC-México**



NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

## Resolución inversa



Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

### named.conf (bloques de in-addr)

```
options {  
    directory "Directorio_del_named"; ← Path absoluto  
};  
zone "." {  
    type hint;  
    file "Archivo_root_servers";  
};  
zone "0.0.127.in-addr.arpa" {  
    type master;  
    file "Archivo_loopback_address";  
};  
zone "IP.invertida.in-addr.arpa" {  
    type master;  
    file "Archivo_tablas_IP-hosts";  
};
```



# Sistema de Nombres de Dominios ( DNS )

## Resolución inversa y named.conf

```
options {  
    directory "/var/named";  
};  
zone "." {  
    type hint;  
    file "named.root";  
};  
zone "0.0.127.in-addr.arpa" {  
    type master;  
    file "named.local";  
};  
zone "2.0.192.in-addr.arpa" {  
    type master;  
    file "2.0.192.zone";  
};
```

←  
Dominio que indica  
resolución inversa

Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

### Registro PTR

Define el nombre de máquina que le corresponde a una dirección de IP.

Formato del registro:

`dirección.IP.invertida [ttl] IN PTR nombre-máquina`

`1.2.0.192.in-addr.arpa. IN PTR dns.mi-empresa.com.mx.`



NIC-MÉXICO

# Sistema de Nombres de Dominios ( DNS )

z.y.x.zone

;  
; BIND data file for in-addr.arpa zone.

@

IN

SOA

**Nuestro nameserver**

**Nuestra cuenta ( real )**

1 ; Serial  
360 ; Refresh  
300 ; Retry  
3600000 ; Expire  
3600 ; TTL

Dirección de  
IP invertida



IN

NS

**Nuestro nameserver**

IN

PTR

**NOMBRE DE MAQUINA**

IN

PTR

**NOMBRE DE MAQUINA**

Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

2.0.192.zone

```
;  
; BIND data file for in-addr.arpa zone.  
;  
@      IN      SOA      dns.mi-empresa.com.mx.  
                          mi-cuenta.mi-empresa.com.mx. (  
                          1      ; Serial  
                          360    ; Refresh  
                          300    ; Retry  
                          3600000 ; Expire  
                          3600 ) ; TTL  
↓  
      IN      NS       dns.mi-empresa.com.mx.  
1     IN      PTR      dns.mi-empresa.com.mx.  
3     IN      PTR      www.mi-empresa.com.mx.
```





## Sistema de Nombres de Dominios ( DNS )

named.local

```
;
; BIND data file for local loopback interface.
;
@      IN      SOA      dns.mi-empresa.com.mx.
                          mi-cuenta.mi-empresa.com.mx. (
                              1          ; Serial
                              360        ; Refresh
                              300        ; Retry
                              3600000    ; Expire
                              3600 )    ; TTL
                          IN      NS      dns.mi-empresa.com.mx.
1      IN      PTR      localhost.
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



Sistema de Nombres de Dominios ( DNS )

Configuración de `/etc/resolv.conf`

# 9. Configuración del resolver

*Ing. Cristóbal Chapital T.*

NIC-México



## Sistema de Nombres de Dominios ( DNS )

### Directiva *search*

```
search dominio-default [ dominio2 ... dominio6 ]
```

```
# Define el orden de búsqueda  
# en los dominios en que el  
# resolver tratará de ubicar  
# una máquina.
```



## Sistema de Nombres de Dominios ( DNS )

### Directiva *nameserver*

**nameserver**

**servidor-de-nombres**

```
# Define el servidor de  
# nombres a utilizar por  
# omisión. Se pueden usar  
# hasta 3 directivas.
```



## Sistema de Nombres de Dominios ( DNS )

**/etc/resolv.conf** (ejemplo)

```
#  
# Archivo de configuración del resolver  
#  
search mi-empresa.com.mx conta.mi-empresa.com.mx  
nameserver dns.mi-empresa.com.mx  
nameserver dns2.mi-empresa.com.mx  
nameserver odns.otra-empresa.com.mx
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



**Sistema de Nombres de Dominios ( DNS )**

**Herramientas para el DNS**

# **10. Herramientas para verificar el DNS**

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

*nslookup*

### Opciones de *nslookup*

- > **server** servidor.de.nombres
- > **ls** [-d] nombre.de.dominio
- > **set type=tipo\_de\_registro**  
tipo\_de\_registro=A, ANY, NS, HINFO,  
TXT, PTR, SOA,  
CNAME.



## Sistema de Nombres de Dominios ( DNS )

### *nslookup* (ejemplo)

```
% nslookup
```

```
Default server: dns.mi-  
empresa.com.mx
```

```
Address: 192.0.2.1
```

```
> www.mi-empresa.com.mx
```

```
Server: dns.mi-empresa.com.mx
```

```
Address: 192.0.2.1
```

```
Name: www.mi-empresa.com.mx
```

```
Address: 192.0.2.3
```

```
> set type=ns
```

```
> mi-empresa.com.mx.
```

```
Server: dns.mi-empresa.com.mx
```

```
Address: 192.0.2.1
```

```
mi-empresa.com.mx. nameserver = dns.mi-  
empresa.com.mx.
```

```
> set type=ptr
```

```
> 3.2.0.192.in-addr.arpa.
```

```
Server: dns.mi-empresa.com.mx
```

```
Address: 200.0.0.1
```

```
3.2.0.192.in-addr-arpa. name = www.mi-  
empresa.com.mx
```

Ing. Cristóbal Chapital T.

NIC-México





## Sistema de Nombres de Dominios ( DNS )

*dig*

### Opciones de *dig*

```
% dig [@servidor] [dominio] [q-type]
servidor= servidor.de.nombres
dominio = nombre.de.dominio.
q-type   = A, ANY, PTR, TXT, NS,
           HINFO, SOA, CNAME
```



## Sistema de Nombres de Dominios ( DNS )

*dig* (ejemplo)

```
% dig @dns.mi-empresa.com.mx mi-empresa.com.mx. ns

; <<>> DiG 8.1 <<>> @dns.mi-empresa.com.mx mi-empresa.com.mx. ns
; (1 server found)
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 10
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; QUERY SECTION:
;;      mi-empresa.com.mx, type = NS, class = IN

;; ANSWER SECTION:
mi-empresa.com.mx.          1D IN NS   dns.mi-empresa.com.mx.

;; ADDITIONAL SECTION:
dns.mi-empresa.com.mx.    1D IN A           192.0.2.1

;; Total query time: 3 msec
;; FROM: dns to SERVER: dns.mi-empresa.com.mx 192.0.2.1
;; WHEN: Thu Jun 11 11:35:06 1998
;; MSG SIZE  sent: 31  rcvd: 71
```

Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

*whois*

### Opciones de *whois*

```
% whois [ -h servidor ] nombre
```

```
servidor = servidor.de.whois v.g.,  
          rs.internic.net,  
          whois.arin.net,  
          whois.nic.mx
```

```
nombre   = nombre.de.dominio o  
          dirección.de.IP
```



## Sistema de Nombres de Dominios ( DNS )

*whois* (ejemplo)

```
% whois -h whois.nic.mx nic.mx
```

```
Nombre del Dominio: nic.mx
```

```
Organizacion:
```

```
[nicmx] Nic-México
```

```
Av. Eugenio Garza Sada 2501 Sur Monterrey N.L., C.P. 64849
```

```
Fecha de creacion: 1997 Ene 15
```

```
Administrative Contact:
```

```
[oscar] Oscar Alejandro Robles Garay
```

```
orobles@nic.mx 52(8) 328 4373
```

```
Technical Contact:
```

```
[oscar] Oscar Alejandro Robles Garay
```

```
orobles@nic.mx 52(8) 328 4373
```

```
Billing Contact:
```

```
[oscar] Oscar Alejandro Robles Garay
```

```
orobles@nic.mx 52(8) 328 4373
```

```
Primary Nameserver: ns.nic.mx
```

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

Requisitos para registrar bajo "mx"

# Requisitos para registrar bajo mx

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Requisitos para registrar bajo "mx"

- Nombres de dominio
  - Caracteres válidos: números, letras, y (-).
  - No debe empezar ni finalizar con (-) .
  - 26 caracteres en total
- Configurar previamente el nameserver.
- Mantener siempre un nameserver resolviendo para el nombre del dominio.



NIC-MÉXICO

## Sistema de Nombres de Dominios ( DNS )

### Requisitos para registrar bajo "mx"

- Identificar el "dominio padre" adecuado al giro que va a tener el dominio solicitado, de acuerdo a lo establecido en:

<http://www.nic.mx/dom/polgral.html#uno>



## Sistema de Nombres de Dominios ( DNS )

URL's y bibliografía

# Referencias

*Ing. Cristóbal Chapital T.*

**NIC-México**





## Sistema de Nombres de Dominios ( DNS )

Documentación referente a nombres de dominio

- ***RFC's***

- <http://www.dns.net/dnsrd/docs/rfc.html>

**RFC's** 805 811 819 882 883 920 973 974 1032 1033 1034 1035 1101 1122 1123 1178 1183 1348 1464 1480 1535  
1536 1537 1591 1611 1612 1637 1664 1706 1712 1713 1794 1811 1816 1876 1884 1886 1912 1956 1982 1995  
1996 2010 2052 2053 2065 2136 2137 2146 2163 2168 2181 2182 2219 2230 2240 2247

- ***URL's y otros doctos.***

- <http://www.dns.net/dnsrd/>

- <http://www.digidem.com/legal/domain.html>

- <http://www.iahc.org/draft-iahc-recommend-00.html>

- [http://www.yahoo.com/Computers\\_and\\_Internet/Internet/Domain\\_Regist](http://www.yahoo.com/Computers_and_Internet/Internet/Domain_Regist)  
ration

Ing. Cristóbal Chapital T.

NIC-México



## Sistema de Nombres de Dominios ( DNS )

Documentación referente a nombres de dominio

- *URL's y otros doctos.*

- [http://www.yahoo.com/Computers\\_and\\_Internet/Internet/Domain\\_Registration/Domain\\_Name\\_Controversies](http://www.yahoo.com/Computers_and_Internet/Internet/Domain_Registration/Domain_Name_Controversies)
- [http://www.yahoo.com/Computers\\_and\\_Internet/Internet/Domain\\_Registration/Network\\_Information\\_Centers](http://www.yahoo.com/Computers_and_Internet/Internet/Domain_Registration/Network_Information_Centers)
- <http://www.uninett.no/navn/domreg.html>
- <http://www.alternic.net>
- [http://www.isc.uci.edu/pub/websoft/wwwstat/country\\_codes.txt](http://www.isc.uci.edu/pub/websoft/wwwstat/country_codes.txt)
- <http://www.nic.mx/dom>
- <http://www.isc.org>



## Sistema de Nombres de Dominios ( DNS )

### Bibliografía:

*DNS and BIND*  
*Paul Albitz & Cricket Liu*  
*O'Reilly & Associates, Inc.*

*<http://www.isc.org>*

*<http://www.dns.net/dnsrd>*

*Ing. Cristóbal Chapital T.*

**NIC-México**



## Sistema de Nombres de Dominios ( DNS )

### Créditos

**Oscar Robles, NIC-México**

Coordinador Administrativo del NIC-México

**orobles@nic.mx**

**Cristóbal Chapital, NIC-México**

Coordinador Operativo del NIC-México

**chapital@nic.mx**

**Francisco Arias, NIC-México**

Soporte Técnico del NIC-México

**farias@nic.mx**

*Ing. Cristóbal Chapital T.*

**NIC-México**



**FACULTAD DE INGENIERIA U.N.A.M.  
DIVISION DE EDUCACION CONTINUA**

**CURSOS A DISTANTANCIA**

**SERVIDORES DE  
NOMBRES DE  
DOMINIO (DNS)**

**DEL 24 DE OCTUBRE AL 14 DE NOVIEMBRE DE 1998  
CLAVE (CA 128)**

**PROFESOR:**

**ING. JAVIER CASTAÑON DE LA VEGA**

# Contenido

---

<b>Introducción.....</b>	<b>1</b>
¿DNS o /etc/hosts?.....	1
Implementación de DNS.....	2
<b>El espacio de nombres DNS.....</b>	<b>3</b>
Dominios.....	3
Dominio raíz.....	3
Dominios de alto nivel.....	3
Dominios de segundo nivel.....	4
Nombre de dominio completamente calificado.....	5
<b>Funcionamiento de DNS.....</b>	<b>6</b>
Resolvedores y servidores de nombres.....	6
Resolución de nombres.....	6
Consultas no recursivas.....	6
Consultas recursivas.....	7
“Caching”.....	7
Consultas inversas.....	8
Zonas de autoridad.....	8
Servidores primarios, secundarios y de sólo caché.....	9
Servidores primarios.....	9
Servidores secundarios.....	9
Transferencia de zona.....	10
Servidores de caché.....	10
Respuestas con y sin autoridad.....	10
<b>La base de datos DNS.....</b>	<b>11</b>
Archivos de zona.....	11
Estructura de los registros.....	11
Nombre.....	11
TTL.....	11
Clase.....	12
Tipo.....	12
Caracteres especiales.....	12
Descripción de los registros.....	13
SOA.....	13
NS.....	15
A.....	15
PTR.....	15
MX.....	16
CNAME.....	16
HINFO.....	16

TXT .....	16
<b>El Berkeley Internet Name Daemon (BIND) .....</b>	<b>17</b>
Configuración del cliente: el archivo /etc/resolv.conf .....	17
Domain .....	17
Search .....	17
Nameserver .....	18
Configuración del servidor: el demonio named .....	18
El archivo de inicio: /etc/named.boot .....	18
directory .....	19
cache .....	19
primary .....	19
secondary .....	19
include .....	20
domain .....	20
forwarders .....	20
El archivo de caché: named.ca .....	20
El archivo de zona: named.hosts .....	21
El archivo de resolución inversa: named.rev .....	21
Ejemplos de implementación de BIND .....	21
Un servidor de sólo caché .....	21
Un servidor de una red pequeña .....	23
<b>Bibliografía .....</b>	<b>26</b>

## **Introducción**

---

Utilizar nombres simbólicos (una cadena de caracteres) para identificar máquinas en una red es más sencillo que utilizar sus direcciones IP. En una red pequeña, el uso del archivo `/etc/hosts` resuelve de una manera aceptable el problema de relacionar nombres simbólicos con sus respectivas direcciones IP.

Sin embargo, cuando la red crece en tamaño, mantener actualizado el archivo `hosts` puede resultar difícil, sobre todo cuando se piensa en replicar dicho archivo y asegurarse que todas las máquinas utilicen una versión actualizada del mismo. Además, debido a su estructura plana, es indispensable que los nombres de cada uno de los anfitriones sea único en toda la estructura de la red, lo cual no resulta difícil en un ambiente conformado por decenas o tal vez centenas de equipos, pero no es así en uno donde conviven miles de anfitriones.

Además, entran en consideración cuestiones de rendimiento y tolerancia a fallos. El ancho de banda que debe utilizarse para replicar el archivo `hosts` depende también de su tamaño; la carga administrativa que implica tener que mantener dicho archivo de una manera centralizada; así como la dependencia de un punto central de administración.

La solución a estos problemas se encontró desarrollando un sistema enfocado en dos puntos clave: eliminación del punto central de administración y creación de un espacio de nombres que no fuera plano sino jerárquico. En este sistema, la administración se distribuye a lo largo de la red participante, lo cual disminuye la carga sobre la red. En segundo lugar, al cambiar el espacio plano de nombramiento por uno jerárquico, se le otorga flexibilidad a la red.

Esto es lo que hace DNS o Servicio de Nombres de Dominio/Domain Name System. La implementación actual de DNS está basada en los RFC's 1034 y 1035, aunque existen varios RFC's más que complementan a los dos mencionados y los extienden, como los 1101 y 1183.

### **¿DNS o /etc/hosts?**

Los dos sistemas están basados en una tabla de anfitriones y sus direcciones IP, sin embargo, en el caso de DNS, la implementación de dicha tabla se realiza bajo la forma de una base de datos distribuida. Considerando lo



anterior, el tamaño y complejidad de una red es una situación importante a considerar antes de optar por alguna de las dos soluciones.

Cuando la red es pequeña, con pocos anfitriones en un solo segmento, el archivo hosts provee facilidad de configuración y uso. Cuando la red es más grande, por las razones mencionadas párrafos arriba, DNS puede ser la alternativa más adecuada. Pero aquí entra en consideración una situación adicional: si la red está conectada a Internet o no. Si no está conectada a Internet y se desea usar DNS no hay mucho que decidir: es necesario mantener los propios servidores DNS. En caso de que la red esté conectada a Internet, cabe la alternativa de que el proveedor de acceso proporcione el servicio de DNS.

Ahora bien, si la red está segmentada en múltiples segmentos a lo largo de una red de área amplia, será preferible instalar servidores DNS en cada segmento, con el propósito de reducir en lo posible el tráfico de consulta de nombres sobre los enlaces lentos de la WAN. Si la red además de encontrarse en la situación anterior, cuenta con una gran número de subredes y anfitriones dentro de cada una de ellas, con DNS aún es posible crear subdominios (concepto que se explicará más adelante) y dividir todavía más las cargas sobre la red y de administración.

## **Implementación de DNS**

En realidad la implementación de DNS se lleva a cabo a través de BIND o Berkeley Internet Name Daemon. En algunos textos, ambos términos se emplean como sinónimos. Para efectos prácticos la diferencia estriba en que DNS es una especificación (descrita por medio de RFC's) y BIND es su implementación más popular.

Esa es la razón por la que en muchos textos de UNIX, dentro del capítulo dedicado a DNS, más que explicar DNS en sí mismo, se explica BIND. En este documento dichos términos no se usarán como sinónimos y se explicarán por separado, en primer lugar explicando la estructura de DNS y luego describiendo su implementación por medio de BIND.

# **El espacio de nombres DNS**

---

## **Dominios**

El espacio de nombres DNS está conformado por un árbol de *dominios*. Un dominio representa un nivel de autoridad jerárquica dentro de la estructura del árbol. Cada dominio puede ser administrado de manera independiente de los demás.

## **Dominio raíz**

En la parte más alta del árbol de autoridad se encuentra el *dominio raíz*, representado por un punto (.).

## **Dominios de alto nivel**

En el siguiente nivel se hallan los *dominios de alto nivel* (top level domains), llamados en algunos textos *dominios raíz* (root level domains). Podrían dividirse en dos tipos: los utilizados en los Estados Unidos y los que emplean el código de país ISO de dos letras. En el primer caso, describen la naturaleza de la organización a la cual pertenecen. Dichos dominios se describen en el RFC 1591 y se muestra un resumen en la Tabla 1.

**Tabla 1**

<b>Dominio</b>	<b>Uso</b>
COM	Entidades comerciales
EDU	Instituciones educativas que ofrecen carreras de al menos 4 años de duración y universidades
GOV	Agencias del Gobierno Federal de los Estados Unidos
INT	Organizaciones establecidas por tratados internacionales
MIL	Organizaciones gubernamentales militares
NET	Proveedores de acceso a Internet
ORG	Organizaciones no lucrativas

Los dominios correspondientes a cada país coexisten con los dominios listados arriba, en la Tabla 2 se listan algunos ejemplos. Los códigos están basados en la norma ISO-3166.

**Tabla 2**

<b>Código</b>	<b>País</b>
CA	Canadá
DE	Alemania
FR	Francia
JP	Japón
MX	México
UK	Reino Unido
US	Estados Unidos

El último caso es especial, ya que cuando es utilizado generalmente se refiere a dominios que hacen referencia a una localidad dentro de Estados Unidos, y no a una organización.

## **Dominios de segundo nivel**

A partir de los dominios de segundo nivel, se pueden incluir máquinas anfitrionas u otros dominio, llamados *subdominios*. En la mayoría de los dominios que usan el código de dos letras para cada país, se emplean los subdominios para contruir una estructura jerárquica que refleja también la naturaleza de la organización que posee el nombre del dominio. En el caso del Reino Unido por ejemplo, se utiliza el sufijo CO en lugar de COM, o en México se emplea GOB en lugar de GOV. Los dominios de segundo nivel son administrados por NIC o Network Information Center. Es a través de dichos centros que se solicitan los nombres de dominio y se garantiza que no existan repeticiones en los nombres de subdominios, dentro de un dominio determinado o *zona de autoridad*.

En México, el organismo encargado de este registro es NIC México. Los subdominios bajo .mx son los mostrados en la Tabla 3.

**Tabla 3**

<b>Subdominio</b>	<b>Uso</b>
.edu.mx	Para instituciones de educación o investigación
.org.mx	Para asociaciones no lucrativas
.net.mx	Para proveedores de servicios de red
.gob.mx	Para instituciones gubernamentales
.com.mx	Para entidades comerciales y aquellas que no se incluyan en las clasificaciones anteriores.

Por lo que respecta a las máquinas anfitrionas, éstas pueden existir en más de un dominio a la vez.

### **Nombre de dominio completamente calificado**

Éste nombre se forma agregando al nombre de la máquina anfitriona el nombre del dominio. Por ejemplo, serpiente.dgsca.unam.mx es el nombre completamente calificado (fully qualified domain name) para la anfitriona serpiente en el dominio dgsca.unam.mx. De esta manera, otras anfitrionas pueden llamarse serpiente, siempre que sea fuera del dominio dgsca.unam.mx.

Los nombres de dominio no son sensitivos a mayúsculas o minúsculas, sin embargo, algunos servidores se confunden al encontrar mezcladas letras mayúsculas y minúsculas. Se ha acostumbrado en los últimos años emplear solamente letras minúsculas.

# Funcionamiento de DNS

---

## Resolvedores y servidores de nombres

DNS es un sistema basado en clientes y servidores. Los clientes hacen solicitudes a los servidores, para que éstos traduzcan un nombre de computadora o de dominio en una dirección IP. En la terminología de DNS, los clientes reciben el nombre de *resolvedores* y los servidores se conocen como *servidores de nombres*.

## Resolución de nombres

Es el nombre que recibe el proceso de traducir un nombre de computadora o de dominio a una dirección IP. En DNS los resolvedores realizan consultas en los servidores de nombres. Son tres los tipos de consultas que los servidores de nombres pueden atender: recursivas, no recursivas (o iterativas) e inversas.

### Consultas no recursivas

Las consultas no recursivas son llevadas a cabo por *servidores no recursivos*. En este caso, cuando un servidor no recursivo recibe una consulta, revisa si tiene autoridad sobre el dominio acerca del cual se le está haciendo la consulta, o busca en su *archivo de caché* para ver si tiene la respuesta. Si no es así, no da una respuesta real, simplemente se limita a dar una *referencia* a un servidor de nombres con autoridad sobre el dominio acerca del cual se hace la consulta.

Las referencias se hacen siguiendo el orden jerárquico dentro de la estructura de DNS. Si por ejemplo, un servidor no conoce acerca de *serpiente.dgsca.unam.mx*, entonces enviará una referencia a los servidores de *dgsca.unam.mx*, *unam.mx*, *mx* o el dominio raíz. Siempre se devolverá la referencia más completa posible, pero esto depende de las direcciones que conozca el servidor. Tal vez no pueda dar una referencia al servidor *dgsca.unam.mx*, pero sí al servidor de *unam.mx*. Siempre se dará la respuesta más completa posible.

Los archivos de caché generalmente vienen precargados con las direcciones de los servidores del dominio raíz, de tal manera que siempre sea posible dar una referencia, inclusive tan arriba como en el dominio raíz.

## **Consultas recursivas**

Son llevadas a cabo por *servidores recursivos*. Este tipo de servidores, sólo pueden dar respuestas reales o mensajes de error, no pueden referir a otro servidor. Más bien siguen las referencias que reciben de un servidor no recursivo, hasta que encuentran la respuesta real y la regresan al cliente, o en caso contrario envían al cliente un mensaje de error. Es decir, los servidores recursivos relevan a los clientes o resolvidores de la tarea de seguir las referencias por ellos mismos. Aunque pudiese parecer una carga de trabajo excesiva para un servidor dar seguimiento a todas las peticiones que le hacen los resolvidores y hacer las consultas por sí mismos, en realidad esto ayuda a mejorar el rendimiento del sistema en general como se explica a continuación.

## **“Caching”**

Conforme un servidor recursivo sigue referencias, adquiere información acerca de otros dominios. Esto da por resultado, que en las búsquedas subsecuentes dará respuesta mucho más rápidamente a todos sus clientes. Al realizar una petición, cada uno de los clientes contribuye a aumentar el “conocimiento” que posee el servidor de nombres, o se beneficia del trabajo hecho anteriormente por el servidor al momento de responder las solicitudes de otros clientes.

Los servidores de nombres de dominio de alto nivel no hacen “caching”. Si lo hicieran, tendrían que guardar una enorme cantidad de respuestas y perderían una gran cantidad de tiempo procesando consultas recursivas. Por ello, los servidores de alto nivel y muchos de segundo nivel son servidores no recursivos, en tanto que los la mayoría de los servidores de segundo nivel o de subdominios sí son recursivos.

Generalmente el proceso de “caching” se aplica sólo a las respuestas correctas. Este “caching” no tiene que ver con el archivo de caché que se mencionó anteriormente. En la sección que explica la implementación de DNS se aborda el tema del archivo de caché.

El tiempo que una respuesta correcta permanece en el caché, depende de la configuración de un parámetro conocido como *tiempo de vida* o TTL (Time To Live). Mientras más grande sea el tiempo de vida para un registro que se ha resuelto, se aumenta la eficiencia, pero disminuye la precisión y viceversa. Ésto se debe a que si el TTL es muy grande, cuando se lleve a cabo otra consulta, se utilizará el dato que está en el caché, y si los datos correspondientes a dicho registro han sido modificados en el servidor de nombres del cual provienen, éstos cambios no serán consultados.

El servidor de nombres va disminuyendo el valor del TTL, y cuando el resolovedor recibe los datos respeta en su propio caché el valor del tiempo de vida especificado, hasta que los datos expiran.

### **Consultas inversas**

En este caso no se trata de traducir un nombre a una dirección IP, sino al contrario, lo que se conoce es la dirección IP y lo que se desea saber es el nombre de dominio que le corresponde. Con el propósito de resolver este tipo de situaciones, existe un dominio especial de alto nivel llamado IN-ADDR.ARPA. El funcionamiento de las consultas inversas y el dominio IN-ADDR.ARPA se basa en las consideraciones discutidas en los siguientes párrafos.

Un nombre completamente calificado, va de lo general a lo particular en una dirección de derecha a izquierda. En el ejemplo serpiente.dgsca.unam.mx, serpiente está dentro de dgsca, y dgsca está dentro de unam, y unam está dentro de mx. Pero las direcciones IP van de lo general a lo particular de izquierda a derecha. En el ejemplo 132.248.104.10, el anfitrión número 10 está en la subred 104, que se encuentra en la red 132.248.

Por lo tanto, cuando se hace una consulta para la dirección 132.248.104.10, ésta se convierte en una consulta para encontrar 10.104.248.132.IN-ADDR.ARPA. Para que la consulta inversa pueda ser exitosa, el administrador de un servidor DNS debe asegurarse de crear una base de datos cuyos registros tenga sus direcciones IP invertidas. Esto es lo que hacen los registros de tipo PTR dentro de DNS. Más adelante se describen este tipo de registros.

### **Zonas de autoridad**

Con frecuencia se emplea este término como sinónimo de dominio, lo cual no es completamente exacto. Una *zona de autoridad* es un fragmento del espacio de nombres DNS sobre el cual un servidor tiene *autoridad*, es decir, sobre el cual es responsable. En sentido estricto, los servidores de nombres trabajan con *zonas*, no con *dominios*. La zona de autoridad puede ser definida dentro de uno o varios archivos. Un servidor de nombres puede tener autoridad sobre varias zonas.

La zona de autoridad de un servidor no necesariamente incluye todos los subdominios bajo un dominio. Esto puede ser por cuestión de eficiencia al

distribuir la carga administrativa o disminuir la carga sobre la red al momento de realizar replicaciones, según se verá a continuación.

## **Servidores primarios, secundarios y de sólo caché**

Un servidor es primario o secundario dependiendo de si tiene autoridad sobre una zona. Si tiene autoridad sobre una zona es porque posee una copia del archivo de zona correspondiente. Un servidor de sólo caché no tiene autoridad alguna, todas las respuestas que otorga las obtiene de otros servidores que sí tienen autoridad sobre otras zonas.

Es frecuente que un servidor sea primario para algunas zonas y secundario para otras.

### **Servidores primarios**

Cada dominio o subdominio cuenta con un solo servidor primario. El servidor primario posee la copia maestra del archivo que contiene la información acerca del dominio. Cuando se realizan cambios en los archivos de una zona, estos cambios se realizan en el servidor primario. Esta información es replicada hacia al menos un servidor secundario.

### **Servidores secundarios**

Este servidor obtiene los datos de una zona en particular mediante la copia del archivo de zona correspondiente desde el servidor primario. Los servidores secundarios se emplean para proveer de redundancia, para optimizar la respuesta o para distribuir la carga de trabajo.

Para cada zona, debería existir al menos un servidor secundario, aunque pueden existir varios servidores secundarios para una misma zona.

### **Redundancia**

Si el servidor primario falla y no existe soporte de parte de uno o más servidores secundarios, los anfitriones de la red podrían perder su habilidad de contactar otros anfitriones (dependiendo de cómo se haya implementado DNS en cada uno de los clientes).

Los servidores secundarios podrían ubicarse en el mismo edificio que el primario o en otra ubicación. Si se encuentran en el mismo edificio, deberían estar conectados a diferentes segmentos de red que el servidor primario. También a diferentes circuitos eléctricos en caso de presentarse una falla en el suministro de fuerza. En otras palabras, un servidor secundario debe ser tan independiente del primario como se pueda.



## **Respuesta**

Si la red es demasiado extensa, cuenta con enlaces lentos tipo WAN o tiene un tráfico muy pesado, no es mala idea colocar un servidor secundario en cada subred o segmento. Este enfoque provocará que las peticiones para resolver nombres no tengan que cruzar por enlaces lentos o atravesar toda la red. Por supuesto, esto sólo resultará de utilidad cuando el tráfico de consultas hechas por los resolvers es menor al tráfico generado por la replicación de los archivos de zona.

## **Carga de trabajo**

Puede ocurrir que la carga de trabajo sobre un servidor sea demasiado alta. En este caso, al configurar otra máquina como servidor secundario se puede aliviar la carga de trabajo de los servidores existentes. Sólo será necesario cuidar cómo es que se configuran los clientes al momento de indicarles cuáles son los servidores de nombres que atenderán sus peticiones.

## **Transferencia de zona**

Es el nombre que recibe el copiado del archivo de zona que hace un servidor secundario. No se llevan a cabo actualizaciones parciales, siempre se transfiere la totalidad del archivo de zona cuando se detectan cambios en el servidor primario.

## **Servidores de caché**

Estos servidores no tienen autoridad sobre zona alguna. Sólo cargan en memoria las direcciones de los dominios raíz y después ejecutan consultas y guardan en su caché las respuestas que obtienen.

Los servidores de caché son útiles para disminuir el tráfico DNS sobre la red, debido a que no realizan transferencias de zona. Esto es particularmente útil cuando se tienen que cruzar enlaces lentos WAN.

## **Respuestas con y sin autoridad**

Los únicos servidores que pueden dar respuestas con autoridad son los servidores primarios y secundarios, dado que ellos sí consultan directamente un archivo de zona. Está "casi" garantizado que las respuestas con autoridad son precisas. Los servidores de caché no proporcionan respuestas con autoridad. Las respuestas sin autoridad son casi siempre precisas, pero corren el riesgo de estar desactualizadas. Aún las respuestas con autoridad no siempre pueden ser precisas, debido a que si se realizó algún cambio en el archivo de zona en el servidor primario y este cambio no ha sido replicado a los servidores secundarios, éstos no tendrán la respuesta correcta debido a la inconsistencia entre los archivos de zona que maneja cada uno de ellos.

# La base de datos DNS

---

## Archivos de zona

La base de datos se guarda en archivos conocidos como *archivos de zona*. Los elementos guardados en dichos archivos reciben el nombre genérico de *registros de recursos* o RR's (Resource Records). Los tipos y formatos de los registros se definen en los RFC's 882, 1035 y 1183.

## Estructura de los registros

La sintaxis general de un registro de recurso es la que sigue:

```
[nombre] [ttl] [clase] tipo datos
```

Los elementos encerrados entre corchetes son opcionales. Los campos se separan por medio de espacios en blanco, ya sean espacios o tabuladores.

### Nombre

Identifica la entidad que describe el registro: anfitrión o dominio. Si el nombre es el mismo en varios registros consecutivos, puede omitirse después del primer registro. Puede ser escrito como un nombre relativo o como un nombre completamente calificado.

Por omisión, se trabaja con nombres completamente calificados, y se agrega el nombre del dominio a todo nombre que no finalice con un punto. Por ejemplo, en el dominio `dgsca.unam.mx` el nombre "serpiente" sería interpretado como "serpiente.dgsca.unam.mx.". Pero si el nombre es introducido como "serpiente.dgsca.unam.mx", como no termina en un punto, se interpreta como un nombre relativo y el resultado final es "serpiente.dgsca.unam.mx.dgsca.unam.mx."

### TTL

Es el tiempo de vida. Especifica cuántos segundos debe permanecer el registro en el caché del servidor antes de ser considerado obsoleto. Con frecuencia se omite, con excepción del archivo de inicio de caché que contiene los nombres y las direcciones de los servidores nombres de alto nivel.

## Clase

Identifica el tipo de red. Existen tres valores: IN para Internet, CH para ChaosNet y HS para Hesiod. CH se considera obsoleto y HS es poco utilizado. El valor por defecto es IN.

## Tipo

Identifica dominios y sus servidores, anfitriones, apuntadores para traducir direcciones a nombres, enrutado de correo y alias principalmente.

Los principales tipos de registros se listan en la Tabla 4

**Tabla 4**

<b>Tipo</b>	<b>Nombre</b>	<b>Uso</b>
SOA	Inicio de autoridad (Start of Authority)	Define una zona de autoridad DNS
NS	Servidor de nombres (Name Server)	Identifica a los servidores en una zona
A	Dirección (Address)	Traduce un nombre en su dirección
PTR	Apuntador (Pointer)	Traduce una dirección en su nombre
MX	Intercambiador de correo (Mail Exchanger)	Controla ruteo de correo.
CNAME	Nombre canónico (Canonical Name)	Alias para una anfitrión
HINFO	Información del anfitrión (Host Info)	Identifica hardware y sistema operativo
RP	Persona responsable (Responsible Person)	Persona que sirve como punto de contacto técnico para un anfitrión determinado.
WKS	Servicios conocidos (Well Known Services)	Servicios que provee un anfitrión
TXT	Texto	Comentarios

## Caracteres especiales

Se pueden utilizar ciertos caracteres especiales, listados en la Tabla 5

**Tabla 5**

<b>Carácter</b>	<b>Uso</b>
:	Comentario
#	Comentario
@	El nombre de dominio actual
()	Permite que el registro ocupe más de una línea
*	Comodín (sólo en el campo nombre)

## Descripción de los registros

A continuación se describen los principales tipos de registros utilizados en una base de datos DNS.

### SOA

Marca el inicio de una zona. Sólo existe un registro SOA para cada zona y debe ser el primer registro en ella. Todos los registros que le siguen son registros de recursos autorizados dentro de dicha zona. Este es un registro típicamente multilínea, por lo que hace uso de los paréntesis. La colocación de los paréntesis no es arbitraria, pero el siguiente arreglo funciona generalmente bien, el cual sirve para ejemplificar la sintaxis de este registro, donde nnnnn son números enteros.

```
@      IN      SOA  nombre      contacto (
                                nnnnn      ; Número de serie
                                nnnnn      ; Refresco
                                nnnnn      ; Reintento
                                nnnnn      ; Expiración
                                nnnnn)    ; Mínimo
```

No es extraño encontrarse con que el campo nombre esté ocupado por el carácter especial “@”, que simboliza el nombre de dominio actual, según se indicó en la Tabla 5. Cabe hacer notar que el nombre de dominio actual es definido en otro archivo. Tampoco es extraño que se omita el campo TTL. La clase es generalmente IN y el tipo es SOA por supuesto.

El campo de datos de este registro incluye los valores descritos en los siguientes párrafos. Los últimos cuatro valores son valores de vencimiento

de tiempo, y representan un compromiso entre exactitud y rendimiento. Para proporcionar más exactitud cuando se presenten cambios en el archivo de zona, deberían ser más pequeños, pero esto penaliza el tráfico en la red, puesto que lo incrementa. Para mejorar el rendimiento, pueden seleccionarse valores más grandes, pero esto penaliza la exactitud, porque las actualizaciones de datos son menos frecuentes.

### **Nombre**

El nombre del servidor de nombres primario del dominio. Usualmente se escribe como un nombre absoluto, es decir, con un punto al final.

### **Contacto**

La dirección de correo electrónico del contacto técnico. Esta dirección no se escribe en el formato habitual "nombre@anfitrión", sino "nombre.anfitrión.", por dos razones: el símbolo @ está reservado por tener un significado especial dentro de DNS y por lo cual se reemplaza por un punto y porque se escribe en formato de dirección absoluta, por lo tanto lleva un punto al final.

### **No de serie**

Puede ser cualquier número entero. Cada vez que se lleva a cabo un cambio en el servidor primario, este número se incrementa. El número de serie es consultado por los servidores secundarios para determinar si han existido cambios en el servidor primario.

Si se cambian los datos en el servidor de nombres sin incrementar el valor del número de serie, dichos cambios serán ignorados.

### **Refresco**

El tiempo medido en segundos que debe transcurrir para que los servidores secundarios contacten al servidor primario en busca de posibles cambios. Algunos textos recomiendan valores desde una hora hasta un día.

### **Reintento**

Si cuando el servidor secundario trata de consultar el número de serie del servidor primario, éste no responde, entonces ocurrirá una espera igual a la establecida en segundos en este parámetro. Los valores recomendados oscilan entre 20 y 60 minutos, es decir, relativamente pequeños.

### **Expiración**

Continuando con el caso anterior, si el servidor primario no se restablece a un estado funcional después de un tiempo razonable, los servidores secundarios siempre fallarán al intentar renovar sus datos, los cuales

deberán ser desechados una vez transcurrido el tiempo marcado por el valor expiración. Para preservar el mayor tiempo posible los datos de los servidores secundarios, este valor debería ser relativamente grande, por ejemplo entre una semana y un hasta un mes.

### **Tiempo de vida mínimo**

Este es el valor de tiempo de vida (TTL) predeterminado para los registros de recursos que omiten el campo TTL. Este es el valor que indica durante cuánto tiempo debe el dato ser conservado en el caché de los servidores recursivos. Los valores típicos oscilan desde unas horas hasta varios días, siendo preferidos los de varios días y hasta unas dos semanas, debido a que se disminuye el tráfico.

### **NS**

Identifica los servidores de nombre de otra zona. Generalmente los registros NS son los que siguen al registro SOA. El formato es:

```
zona [ttl] [clase] NS nombre_de_anfitrión
```

Es recomendable listar todos los servidores que tienen autoridad sobre alguna zona dentro de la red. Son los registros NS los que definen subdominios y delegan autoridad en ellos:

### **A**

Dado que son los registros que asocian un nombre de anfitrión en una dirección IP, son los más comunes en una base de datos DNS. Si un anfitrión posee más de una interfase, debe tener un registro para cada interfase adicional. La sintaxis del registro es como sigue:

```
nombre_de_anfitrión [ttl] [clase] A dirección_ip
```

Es responsabilidad del administrador que los registros A coincidan con su correspondiente registro PTR.

### **PTR**

Asocia una dirección IP con un nombre canónico de anfitrión, es decir, la resolución inversa. La sintaxis es la que sigue:

```
dirección_ip [ttl] [clase] PTR nombre_de_anfitrión
```

En el caso de la dirección IP, no termina en un punto y por lo tanto es una dirección relativa, pero no relativa al nombre de dominio actual, sino al dominio IN-ADDR.ARPA.

## **MX**

Es el registro intercambiador de correo. Ayuda a enrutar el correo dando preferencia a determinados anfitriones (generalmente servidores de correo) sobre otros, por medio de un número entero que se guarda en el área correspondiente al campo de datos, según se ve en la sintaxis del comando.

```
nombre [ttl] [clase] MX preferencia anfitrión ...
```

Donde preferencia es un número entero entre 0 y 65,535. El anfitrión con el número de preferencia más pequeño es el de más alta prioridad. Cuando se listan diversos anfitriones, es conveniente dejar un margen entre los números de preferencia, para el caso de que haya que intercalar un anfitrión adicional.

## **CNAME**

Asocia un alias a un *nombre canónico* de anfitrión. El nombre canónico es el designado en el registro A correspondiente al anfitrión. Un alias es útil para asignar nombres descriptivos a los anfitriones. La sintaxis es como sigue:

```
alias [ttl] [clase] CNAME nombre_de_anfitrión
```

Si un anfitrión tiene más de un alias, cada registro CNAME debe apuntar al nombre original o nombre canónico, no a un alias.

## **HINFO**

Guarda información acerca del hardware y sistema operativo de un anfitrión. Aunque el RFC 1340 recomienda algunos valores a utilizar, se puede usar cualquier cadena de caracteres en el campo de datos, según se muestra en la sintaxis del registro a continuación:

```
nombre [ttl] [clase] HINFO tipo_de_cpu  
sistema_operativo
```

## **TXT**

Sirve para agregar cualquier cadena de texto a un registro DNS. Su sintaxis es como sigue:

```
nombre [ttl] [clase] TXT información
```

## **El Berkeley Internet Name Daemon (BIND)**

Un sistema basado en BIND se divide en dos partes principales:

- El agente de resolución que realiza consultas por medio de DNS.
- El demonio **named** que contesta consultas.

Los servidores de nombres DNS corren el demonio **named**. Este demonio es quien contesta las consultas, pregunta a otros servidores en caso de que no pueda resolver una consulta, hace el “caching” de las respuestas obtenidas y ejecuta las transferencias de zonas entre los servidores primarios y los secundarios.

### **Configuración del cliente: el archivo `/etc/resolv.conf`**

Cada cliente debe contar con un archivo llamado `/etc/resolv.conf` que lista los servidores de nombres que se van a consultar y en qué orden deben consultarse. Además guarda el nombre de dominio local.

En algunos sistemas, es suficiente con agregar una línea `nameserver` dentro del archivo `/etc/resolv.conf` conforme se explica más adelante, para que el cliente o resolovedor comience a utilizar DNS en lugar del archivo `/etc/hosts`. Sin embargo, en otros hay que modificar algún otro archivo para que se comience a utilizar DNS.

En dicho archivo se cuenta con las sentencias descritas en los siguientes párrafos.

#### **Domain**

Define el nombre de dominio local. El RFC 1535 recomienda fuertemente no usar esta sentencia, y emplear en lugar de ella la sentencia `search`. La sintaxis es la siguiente:

```
domain nombre_del_dominio
```

#### **Search**

Define una lista de nombres de dominio a probar con los nombres de anfitriones a resolver. Si no se tiene éxito con el primer dominio especificado, se intenta con el segundo y así sucesivamente. La sintaxis es la siguiente:



```
search nombre_de_dominio ...
```

Se pueden especificar hasta seis dominios.

### **Nameserver**

Establece las direcciones IP de los servidores de nombres de dominio. Pueden listarse de uno hasta tres servidores de nombres, repitiendo las líneas `nameserver`. El primer servidor listado es el que primero se contacta siempre. Si éste falla, se realizan hasta 4 reintentos, en cada uno de ellos se incrementa el tiempo entre cada reintento.

La sintaxis es como sigue:

```
nameserver dirección_ip
```

Si un anfitrión es además servidor de nombres, no debe usarse la dirección 127.0.0.1 sino la dirección IP real, ya que pueden surgir problemas. Los servidores de nombres deben ser recursivos, ya que el resolvidor o cliente no sabe cómo manejar las referencias.

### **Configuración del servidor: el demonio named**

La configuración de `named` requiere tener listas tres cosas:

- El archivo de inicio
- El archivo de caché
- El archivo o archivos de datos que contienen los mapeos de direcciones para cada zona en el caso de servidores primarios, junto con el archivo para resolver consultas inversas.

### **El archivo de inicio: `/etc/named.boot`**

`named` se inicia durante el arranque del sistema, lee su configuración desde el archivo `/etc/named.boot`, escribe su PID en el archivo `/etc/named.pid`, y después comienza a escuchar peticiones DNS en el puerto determinado en `/etc/services`.

`named` corre continuamente, pero puede enviársele una señal para que termine:

```
kill -señal `cat /etc/named.pid`
```

El archivo `named.boot` especifica el rol del servidor relativo a cada zona (primario, secundario o de caché), puede apuntar a otros archivos de configuración y a otros servidores de nombres.

De manera general, el formato del archivo `named.boot` es como sigue:

```
directory nombre_de_directorio
cache      . nombre_de_archivo
primary    zona nombre_de_archivo
secondary  zona dirección_ip ... nombre_de_archivo
```

;Las siguientes entradas son opcionales

```
include nombre_de_archivo
domain nombre_del_dominio
forwarders dirección_ip
```

Los comentarios se colocan con un punto y coma. Las entradas se separan por espacios en blanco o tabuladores. En los siguientes párrafos se explican cada una de las sentencias anteriores.

### **directory**

Indica que todos los archivos que se mencionen posteriormente se encuentran en el directorio especificado.

### **cache**

Esta sentencia activa la memoria caché, lee la información acerca de los servidores raíz desde el archivo `/etc/named.ca` o `/var/named.ca`. Esta información también la carga en memoria.

### **primary**

Especifica que este anfitrión tiene autoridad para la zona señalada, es decir que es el servidor primario para dicha zona. Además indica el nombre del archivo donde se encuentran guardados los registros de recurso. Pueden existir múltiples líneas `primary` para múltiples zonas.

### **secondary**

Especifica que este anfitrión es un servidor secundario para la zona señalada. En este caso, se escribe la dirección IP del servidor de nombres primario desde el cual se obtienen los datos de la zona y el nombre del archivo en el cual se van a guardar localmente dichos datos para hacer el "caching". Cuando el servidor secundario arranca, carga la información de la zona desde este archivo, y luego consulta el número de serie del servidor primario para comprobar si es que existieron cambios.

Pueden especificarse diferentes direcciones IP, pero ello no significa que se señale a diferentes máquinas, sino a diferentes interfases en una misma máquina en caso de que una de ellas falle. No es una buena idea copiar la información desde otro servidor secundario. Pueden existir múltiples líneas secondary.

### **include**

Si la cantidad de directivas secondary crece demasiado, el archivo named.boot puede crecer demasiado y ser difícil de mantener. Para evitar lo anterior, puede incluirse el contenido de otros archivos desde el directorio especificado al principio del archivo de inicio.

### **domain**

Especifica un dominio a ser agregado a las consultas incompletas que no incluyan un nombre de dominio. Sin embargo, su uso no es recomendado actualmente, ya que se considera que es responsabilidad del resolvidor especificar un nombre completo.

### **forwarders**

Especifica la dirección IP de un servidor al cual se van a reenviar las consultas que no puedan ser resueltas por los demás servidores de nombre. Este servidor, normalmente muy potente, consultará otros servidores de nombres fuera de la red y regresará las respuestas que obtenga, creando rápidamente un caché muy completo, puesto que hace "caching" con las solicitudes de todos los clientes de la red. De otra manera, el caché de un servidor de nombres sólo beneficiaría a los usuarios cuyas peticiones se dirigieran directamente a dicho servidor. El uso de un "forwarder" es una manera de hacer como si todos los servidores de nombres de una red pudiesen compartir sus cachés.

## **El archivo de caché: named.ca**

Aunque es llamado un archivo de caché, en realidad sirve para inicializar los nombres y las direcciones de los servidores de nivel superior o servidores raíz. Si se consultan varios libros y se comparan tanto los nombres como las direcciones de los servidores de nivel raíz, se podrán encontrar discrepancias entre un texto y otro. Esto es debido a que de tiempo en tiempo, cambian los servidores de nivel raíz. Un archivo de caché actualizado puede obtenerse de <ftp://rs.internic.net/domain/named.cache>.

En este archivo es importante notar el uso que se le da al punto. En el campo correspondiente al nombre se utiliza para simbolizar la raíz. Del mismo

modo, cada uno de los nombres de servidores termina en un punto, para evitar ser tratados como nombres relativos.

## **El archivo de zona: named.hosts**

En realidad pueden existir varios archivos de zona, y pueden recibir cualquier otro nombre que describa de mejor manera su contenido. Ello se especifica en el archivo `named.boot`, según se explicó en la sección correspondiente. La manera de construir este archivo se detalla en la sección relativa a la base de datos DNS.

## **El archivo de resolución inversa: named.rev**

Traduce direcciones IP en nombres de anfitrión, exactamente al revés que el archivo `named.hosts`. El formato que utiliza es el mismo que el archivo `named.hosts`, salvo porque se invierten las posiciones que ocupan las direcciones IP y el nombre de los anfitriones. Respecto a las direcciones IP, éstas no se escriben igual que en el archivo `named.hosts`, sino que debe seguirse la convención establecida por `IN-ADDR.ARPA`.

Debe definirse una zona `IN-ADDR.ARPA` para cada zona que maneje un servidor. Cada zona debe tener su propio registro `SOA` y su propio conjunto de registros `RR`.

Cuando la red es muy grande y está dividida en varias subredes, resulta conveniente crear un archivo `IN-ADDR.ARPA` por cada subred. En los archivos de resolución inversa, el tipo de registro utilizado es el `PTR`. Es importante resaltar la importancia que tiene que cada registro `PTR` esté correctamente asociado con el registro `A` correspondiente.

## **Ejemplos de implementación de BIND**

### **Un servidor de sólo caché**

Este es el primer ejemplo porque es el servidor más sencillo de configurar, dado que un servidor de sólo caché no maneja archivos de zona, según se explicó en la sección correspondiente. Así que lo que hay que hacer, es configurar adecuadamente el archivo `named.boot` para que inicialice el caché de servidores de dominio raíz. Este archivo debe estar actualizado. También se incluye una muestra del mismo.

## named.boot

```
; named.boot para empresa.com
; Este el archivo de inicio de un servidor de sólo caché.

directory /var/named
cache . root

; Fin de archivo
```

## named.ca

```
; This file holds the information on root name servers needed to
; initialize cache of Internet domain name servers
; (e.g. reference this file in the "cache ."
; configuration file of BIND domain name servers).
;
; This file is made available by InterNIC registration services
; under anonymous FTP as
; file /domain/named.root ; on server FTP.RS.INTERNIC.NET
; -OR- under Gopher at RS.INTERNIC.NET
; under menu InterNIC Registration Services (NSI)
; submenu InterNIC Registration Archives ; file named.root
;
; last update: Aug 22, 1997
; related version of root zone: 1997082200
;
;
; formerly NS.INTERNIC.NET
;
.          3600000 IN NS      A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000 A      198.41.0.4
;
; formerly NS1.ISI.EDU
;
.          3600000 NS       B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000 A      128.9.0.107
;
; formerly C.PSI.NET
;
.          3600000 NS       C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000 A      192.33.4.12
;
; formerly TERP.UMD.EDU
;
.          3600000 NS       D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET. 3600000 A      128.8.10.90
;
; formerly NS.NASA.GOV
;
.          3600000 NS       E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET. 3600000 A      192.203.230.10
;
; formerly NS.ISC.ORG
;
```

```

.           3600000   NS   F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET. 3600000   A   192.5.5.241
;
; formerly NS.NIC.DDN.MIL
;
.           3600000   NS   G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET. 3600000   A   192.112.36.4
;
; formerly AOS.ARL.ARMY.MIL
;
.           3600000   NS   H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET. 3600000   A   128.63.2.53
;
; formerly NIC.NORDU.NET
;
.           3600000   NS   I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET. 3600000   A   192.36.148.17
;
; temporarily housed at NSI (InterNIC)
;
.           3600000   NS   J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET. 3600000   A   198.41.0.10
;
; housed in LINX, operated by RIPE NCC
;
.           3600000   NS   K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET. 3600000   A   193.0.14.129
;
; temporarily housed at ISI (IANA)
;
.           3600000   NS   L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET. 3600000   A   198.32.64.12
;
; housed in Japan, operated by WIDE
;
.           3600000   NS   M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET. 3600000   A   202.12.27.33
; End of File

```

## Un servidor de una red pequeña

### named.boot

```

; named.boot para empresa.com
;
directory /var/named
cache .
primary empresa.com      named.ca
primary 194.40.27.IN-ADDR.ARPA  empresa.hosts
primary 0.0.127.IN-ADDR.ARPA  empresa.rev
primary                                local.rev
;
; Fin de archivo

```

## named.ca

Es el mismo que el empleado en el caso anterior de un servidor de sólo caché.

## empresa.hosts (named.hosts)

Este es el archivo de zona. No se siguió el nombre convencional que hubiese podido ser named.hosts.

```
;
; empresa.hosts
; Archivo de zona para empresa.com
;
@           IN      SOA     empresa.com.      admin.empresa.com. (
                    132          ; Número de serie
                    21600         ; Refresco
                    1200          ; Reintento
                    3600000        ; Expiración
                    432000 )       ; Tiempo de vida mínimo

                    IN      NS      empresa.com.
                    IN      NS      servidor.isp.com.
                    IN      A      132.40.27.1
                    IN      MX     10 empresa.com.
                    IN      MX     20 mail.isp.com.
localhost    IN      A      127.0.0.1
unixsrv      IN      A      194.40.27.2
www          IN      CNAME   unixsrv.
ftp          IN      CNAME   unixsrv.
ntsrv       IN      A      194.40.27.3
            IN      MX     10 empresa.com.
unixsrv2     IN      A      194.40.27.15
            IN      MX     10 empresa.com.
oracle       IN      A      194.40.27.21
```

## empresa.rev (named.rev)

Estos son los mapeos inversos correspondientes.

```
;
; empresa.rev
; Archivo de resolución inversa para empresa.com
; 27.40.194.in-addr.arpa
;
@           IN      SOA     empresa.com.      admin.empresa.com. (
                    48           ; Número de serie
                    7200         ; Refresco           2 horas
                    1800         ; Reintento           30 min
                    1209600       ; Expiración         2 sem
                    7200 )       ; TTL mínimo         2 horas

                    IN      NS      empresa.com
                    IN      NS      servidor.isp.com
```

## **Bibliografía**

---

Parker, Tim. *Teach yourself TCP/IP in 14 days*. 2ª ed., Ed. Sams Publishing, Estados Unidos, 1996.

Nemeth, Evi; Snyder, Garth; Seebas, Scott y Hein, Trent. *UNIX System Administration Handbook*, 2ª ed., Ed. Prentice Hall, Estados Unidos, 1994.

Tackett Jr, Jack y Gunter, David. *Special Edition Using Linux*, 2ª ed, Ed. Que, Estados Unidos, 1996.

RFC 1035 -- Domain names - implementation and specification. P.V. Mockapetris. Nov-01-1987.

RFC 1034 -- Domain names - concepts and facilities. P.V. Mockapetris. Nov-01-1987.

RFC 1033 -- Domain administrators operations guide. M. Lottor. Nov-01-1987.

RFC 1032 -- Domain administrators guide. M.K. Stahl. Nov-01-1987.

RFC 1101 -- DNS encoding of network names and other types. P.V. Mockapetris. Apr-01-1989.

RFC 1183 -- New DNS RR Definitions. C.F. Everhart, L.A. Mamakos, R. Ullmann, P.V. Mockapetris. Oct-01-1990.



2	IN	PTR	unixsrv.empresa.com.
3	IN	PTR	ntsrv.empresa.com.
15	IN	PTR	unixsrv2.empresa.com.
21	IN	PTR	oracle.empresa.com.