

El Sistema de Nombres de Dominio DNS

Miguel Angel Astor Romero

9 de Noviembre de 2017

Agenda

- 1 Introducción
- 2 Antecedentes e Historia
- 3 Especificación, Implementación y Uso
- 4 Autoridades
- 5 Conclusiones

Nombres para máquinas

Las computadoras utilizan nombres de bajo nivel para referirse a los dispositivos y estructuras de datos que manipulan:

- Partición de disco real 4
- Archivo 4586
- Host 192.168.1.101
- Archivo 8785 en host 192.168.1.104 por puerto 80

Para nosotros es mejor usar nombres de más alto nivel:

- `/dev/sda4`
- `/home/miky/Documentos/holamundo.c`
- `diancie.cicore.ciens.ucv.ve`
- `http://www.example.com/index.html`

Espacios de nombre y sus tipos

El conjunto de todos los nombres conocidos para una categoría de datos se conoce como espacio de nombres.

Espacio de nombres plano

- Ana, Miguel, Antonio.
- Máquinas 1, 2, 3, 4.
- Puerto 80, 81, 82.

Espacio de nombres jerárquico

- C:/a/b/c/d.txt
- correo.ciens.ucv.ve
- +58-295-569-3028

El archivo HOSTS

- Usado en ARPANET.
- Un archivo central se copiaba en cada nodo.
- Cada nodo podía modificar su copia.
- Espacio de nombres plano.
- No es escalable.
- Hoy sigue en uso.

Contenido

```
127.0.0.1    localhost loopback
::1         localhost
```

Ubicaciones comunes

Linux /etc/hosts

Windows %SystemRoot%/System32/drivers/etc/hosts

macOS /private/etc/hosts

La Inventora de HOSTS



Nombre Elizabeth Jocelyn
"Jake" Feinler

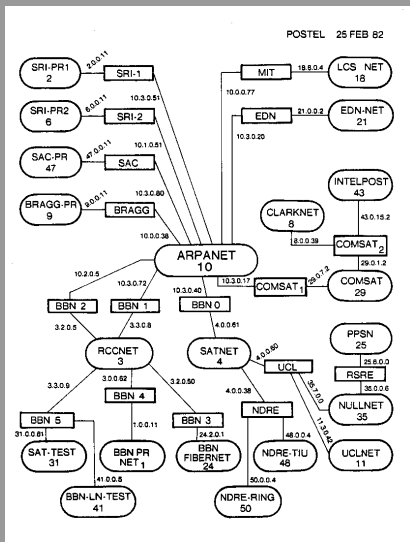
Nacimiento 22 de marzo de 1931

Afiliación SRI International.

Cargos Fundadora y primera
directora de NIC e
InterNIC.

Otros logros Directora del equipo
inventor del sistema
WHOIS.

Mapa de la Internet



J. Postel, febrero de 1982.

Invencción de DNS

El archivo HOSTS evidentemente escala muy mal. En 1983 Paul Mockapetris propone DNS como solución alternativa en el RFC 882.



Nombre Paul Mockapetris

Nacimiento 1948

Afiliación USC ISI.

El Sistema de Nombres de Dominio de Internet

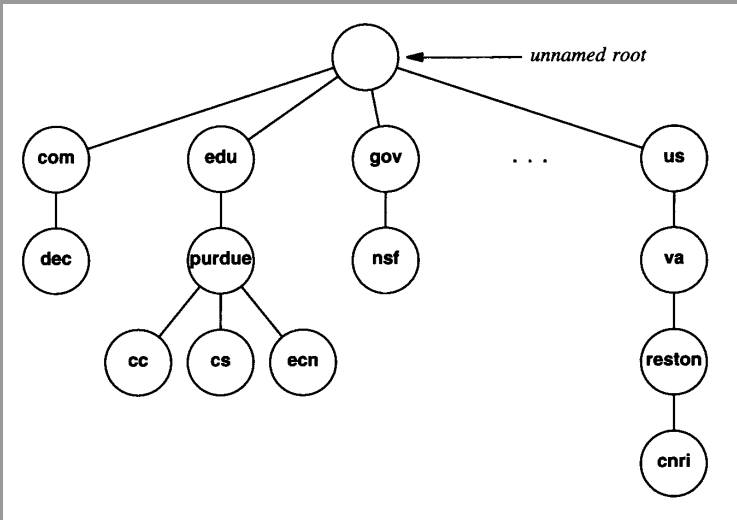
Es un sistema distribuido que permite asociar nombres a máquinas.
Posee dos funciones principales:

- Definición del espacio de nombres:
 - El espacio de nombres es jerárquico.
 - Un nombre se divide en etiquetas separadas por punto.
 - La jerarquía es arbitraria.

icaro.cicore.ciens.ucv.ve

- Especificación del protocolo de resolución de nombres

El espacio de nombres forma un arbol



Almacenamiento de información

DNS es una base de datos distribuida. Algunas de sus entradas son:

Tipo	Significado	Contenido
A	Dirección de <i>host</i> IPv4	Dirección IP de 32 bits.
AAAA	Dirección de <i>host</i> IPv6	Dirección IP de 128 bits.
CNAME	Nombre canónico	Nombre canónico para un alias.
NS	Servidor de nombre	Servidor autorizado para un subdominio.
MX	Servidor de correo	Servidor que acepta correo en un dominio.
PTR	Apuntador	Nombre de dominio.
SOA	Inicio de autoridad	Información sobre autoridad para una zona.

Entre muchas otras...

Implementación de la base de datos

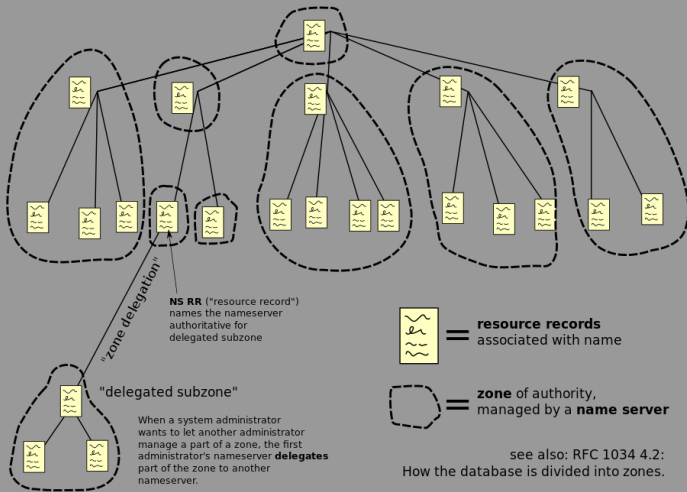
```
ve. 172800 IN NS ns1.nic.ve.  
ve. 172800 IN NS ns2.nic.ve.  
ve. 172800 IN NS ns3.nic.ve.  
ve. 172800 IN NS ns4.nic.ve.  
ns1.nic.ve. 172800 IN A 150.188.228.4  
ns1.nic.ve. 172800 IN AAAA 2001:1338:0:0:0:0:0:2  
ns2.nic.ve. 172800 IN A 150.188.228.5  
ns2.nic.ve. 172800 IN AAAA 2001:1338:0:0:0:0:0:3  
ns3.nic.ve. 172800 IN A 190.9.129.56  
ns4.nic.ve. 172800 IN A 190.202.128.43
```

Base de datos de la zona raiz:

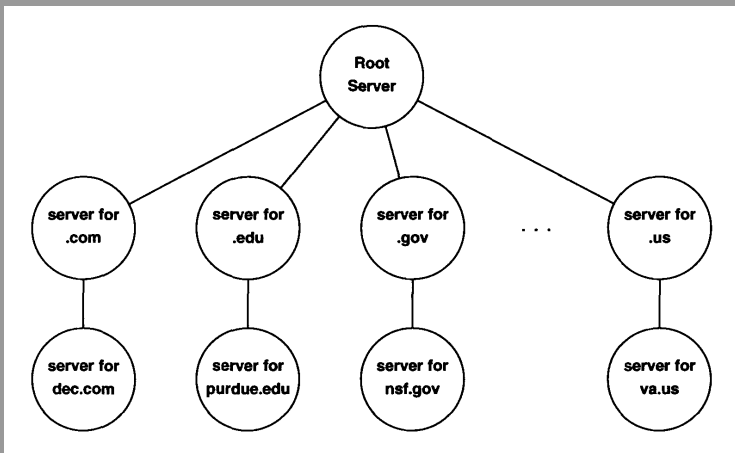
<https://www.internic.net/domain/root.zone>

División en zonas

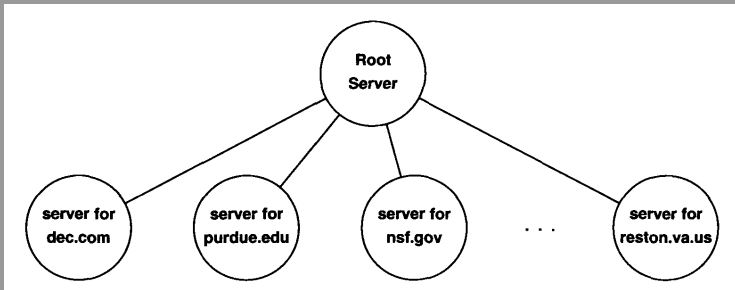
Domain Name Space



Servidores para resolución de nombres... en teoría

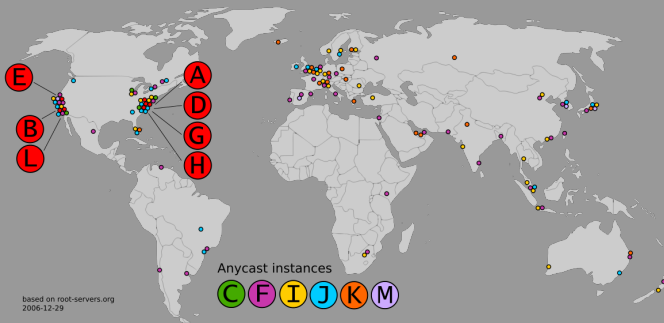


En la práctica



Servidores de la zona raíz

Identificados con letras de la A a la M. Son 13 canónicos, pero se reparten en más de 630 réplicas usando enrutamiento *anycast* y/o sistemas de balanceo de carga.



Servidores raiz alternativos

- AlterNIC (desaparecido).
- Emercoin.
- Namecoin.
- New Nations.
- ORSN.
- OpenNIC.

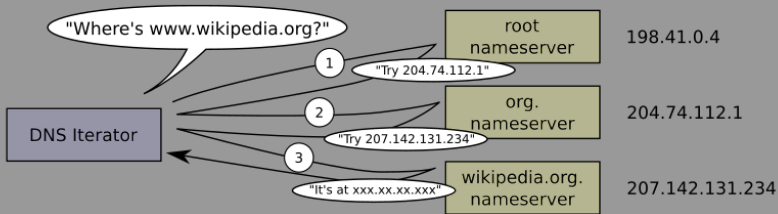
Resolvedores de nombres

Un cliente DNS se conoce como “resolvedor”. Existen dos tipos de resolvedores:

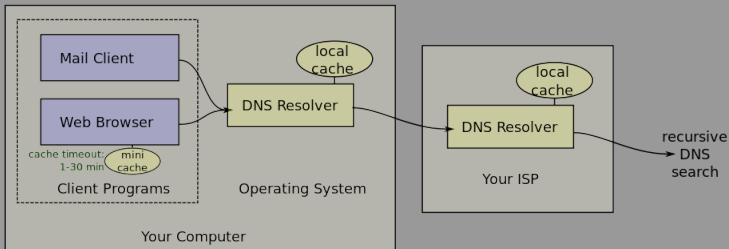
- Iterativos.
- Recursivos.

Un servidor de nombres puede rehusar o no implementar el mecanismo recursivo. Por lo tanto, los resolvedores deben implementar el mecanismo iterativo obligatoriamente.

Resolvedores iterativos



Resolvedores recursivos

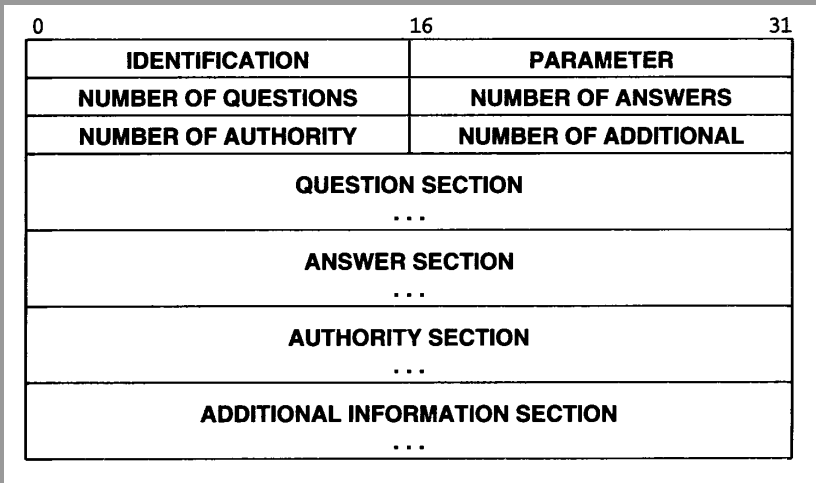


Caché de mensajes

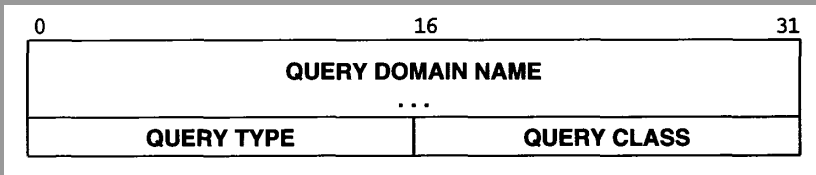
Los resolvedores pueden implementar memorias caché donde almacenan los nombres resueltos recientemente:

- Las entradas incluyen:
 - Nombre.
 - Dirección IP.
 - Servidor de nombres autorizado.
 - Dirección IP del servidor de nombres.
- Las entradas permanecen un tiempo limitado en caché.
 - Usualmente 30 minutos.
 - Internet Explorer mantenía entradas por 24 horas.
- Todas las respuestas que se generan desde caché se marcan como *no autorizadas*.
 - Compromiso entre eficiencia y seguridad.

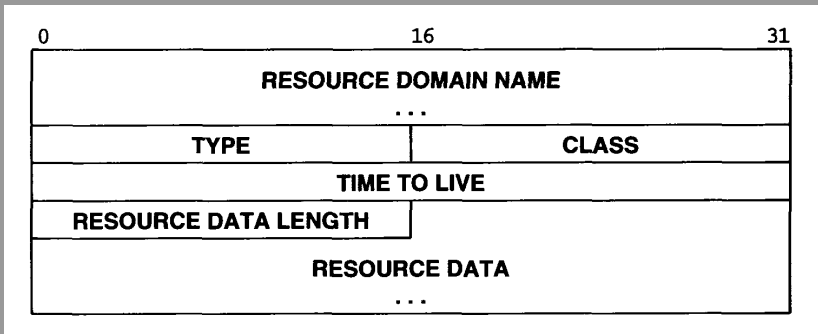
Formato de mensaje de DNS (solicitud y respuesta)



Campo de solicitud de nombre de dominio



Campo de recurso de nombre de dominio



Almacenamiento de nombres en los mensajes

- Un nombre de dominio se almacena como una secuencia de etiquetas:
 - Originalmente solo ASCII.
 - Se permite internacionalización con el sistema IDNA (RFC 3490).
 - Cada etiqueta está precedida por un octeto que indica su longitud:
 - ¿Longitud máxima de una etiqueta?
 - Una longitud de 0 indica el fin del nombre.

Búsqueda reversa

- Dada una dirección IP, retorna cuales nombres de dominio están asociados a esa dirección.
- Se implementa mediante los dominios in-addr.arpa para IPv4 y ip6.arpa:
 - Dirección: 190.169.94.200
 - Consulta: 200.94.169.190.in-addr.arpa
 - La base de datos de in-addr.arpa usa registros PTR.
- Usos:
 - Depuración de problemas de red.
 - Comando “traceroute”.
 - Autenticación de dominios (FCrDNS).
 - Logging de conexiones.

API de C para llamar al resolovedor del sistema

```
#include <netdb.h>

struct hostent {
    char *h_name;           /* official name of host */
    char **h_aliases;      /* alias list */
    int h_addrtype;        /* host address type */
    int h_length;          /* length of address */
    char **h_addr_list;    /* list of addresses */
}

#define h_addr h_addr_list[0]

struct host_ent * gethostbyname(const char * name);
```

Ejemplo

```
#include <stdio.h>
#include <netdb.h>      /* For gethostbyname() */
#include <sys/socket.h> /* For AF_INET */
#include <netinet/in.h> /* For struct in_addr */
#include <arpa/inet.h>  /* for inet_ntoa() */

typedef struct in_addr addr_t;

int main(int argc, char ** argv) {
    char * hostname = argc > 1 ? argv[1] : "google.com";
    struct hostent * host = gethostbyname(hostname);

    if (host == NULL) {
        perror(argv[0]);
        return 1;
    }
}
```

Ejemplo (cont.)

```
} else {  
    printf("NAME: %s\n", host->h_name);  
  
    int i;  
    for (i = 0; host->h_aliases[i]; i++)  
        printf("ALIAS %d: %s\n", i, host->h_aliases[i]);  
  
    int ipv4 = host->h_addrtype == AF_INET;  
    printf("TYPE: %s\n", ipv4 ? "IPv4" : "IPv6");  
  
    printf("LENGTH: %d\n", host->h_length);
```

Ejemplo (cont.)

```
for (i = 0; host->h_addr_list[i]; i++) {
    addr_t * addr = (addr_t *)host->h_addr_list[i];
    printf("ADDR: %s\n", inet_ntoa(*addr));
}

return 0;
}
```

Resultado

```

miky@LIS: -
Archivo Editar Ver Buscar Terminal Ayuda
miky@LIS:~$ gcc -o hosts hosts.c
miky@LIS:~$ ./hosts seed.bitcoin.sipa.be
NAME: seed.bitcoin.sipa.be
TYPE: IPv4
LENGTH: 4
ADDR: 2.30.83.16
ADDR: 5.9.0.236
ADDR: 68.111.10.219
ADDR: 5.22.250.183
ADDR: 109.66.164.124
ADDR: 5.135.176.208
ADDR: 23.238.132.188
ADDR: 5.186.53.67
ADDR: 37.1.202.134
ADDR: 5.189.134.4
ADDR: 211.57.201.54
ADDR: 216.126.212.157
ADDR: 13.80.67.162
ADDR: 23.95.117.72
ADDR: 213.113.46.169
ADDR: 5.149.251.69
ADDR: 83.220.190.27
ADDR: 217.182.200.54
ADDR: 144.76.84.53
```

El dios de Internet



Nombre Jonathan Bruce
"Jon" Postel.

Nacimiento 6 de agosto de 1943

Muerte 16 de octubre de 1998

Afiliación UCLA y USC ISI.

Cargos Editor de la serie RFC,
Administrador de
IANA.

RFC's Relevantes a DNS:
819, 881 y 1591. En
total más de 200(!).

ICANN e IANA

- Establecida en 1998.
- Empresa sin fines de lucro.
- Supervisada por el DOC de los Estados Unidos hasta 2016.
- Responsable de:
 - Asignación de bloques de direcciones IP.
 - Mantenimiento de los servidores *Top-Level* de DNS.
 - Creación o eliminación de dominios *Root*.



Internet Assigned Numbers Authority

Resumen

- DNS es una base de datos distribuida que permite asociar nombres con direcciones IP.
- El espacio de nombres de DNS es completamente genérico.
- Se implementa por zonas para aumentar la eficiencia y alcanzar redundancia.
- La resolución de nombres puede hacerse de forma iterativa o recursiva, e incluso en reversa.

RFC's históricos

- Z. Su y J. Postel, *The Domain Naming Convention for Internet User Applications*, RFC 819, IETF, 1982.
- J. Postel, *The Domain Names Plan and Schedule*, RFC 881, IETF, 1983.
- P. Mockapetris, *DOMAIN NAMES - CONCEPTS AND FACILITIES*, RFC 882, IETF, 1983.
- P. Mockapetris, *DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION*, RFC 883, IETF, 1983.
- J. Postel y J. Reynolds, *Domain Requirements*, RFC 920, IETF, 1984.

RFC's importantes

- P. Mockapetris, *DOMAIN NAMES - CONCEPTS AND FACILITIES*, RFC 1034, IETF Internet Standard, 1987.
- P. Mockapetris, *DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION*, RFC 1035, IETF Internet Standard, 1987.
- J. Postel, *Domain Name System Structure and Delegation*, RFC 1591, IETF Informational, 1994.

Referencias

- A. Tanenbaum, *Redes de Computadoras*, 4ª Edición, Pearson, 2003.
- D. Comer, *Redes Globales de Información con Internet y TCP/IP: Principios Básicos, Protocolos y Arquitectura*, 3ª Edición, Prentice Hall, 1996.

