Servicio DNS v1.0

Guillermo Pérez Trabado *©2016-2024*

Diseño de Infraestructuras de Redes

Depto. de Arquitectura de Computadores - Universidad de Málaga

Introducción al escenario

En todas las etapas previas hemos diseñado la red completa de una empresa formada por las redes internas (*LAN networks*) de varios sites unidas entre sí por una red corporativa (*backbone network*). Además hemos añadido conexiones a Internet de forma que además tenemos dos tipos de interacciones entre la empresa e Internet. Por un lado, los terminales de los departamentos pueden acceder a cualquier dirección de Internet usando traducción de direcciones (*outward dynamic NAT*) y por otro lado, los usuarios de Internet pueden acceder a los puertos TCP o UDP publicados expresamente como *servicios en Internet* de la empresa de nuevo mediante la traducción de direcciones (inward static NAT).

Todo el funcionamiento posible de la red hasta ahora se basa en usar direcciones IP numéricas para indicar el destino de una conexión. Esto no sería un problema para sistemas automaticos configurados por el personal IT ya que está acostumbrado a usar direcciones numéricas y dichas direcciones suelen encontrarse en fichero de configuración que no tienen que volver a modificase con frecuencia. Un ejemplo de este tipo de aplicaciones automáticas que se configuran con direcciones IP son sistemas de copia de seguridad (backup) o sistemas distribuidos (mirroring de información entre sites).

Sin embargo, los usuarios que no tienen formación IT no son demasiado buenos recordando direcciones IP, por lo que necesitan usar nombres para las dirección (mucho más fáciles de recordar). Para poder usar nombres, los sistemas operativos usan un servicio de traducción de nombres de sistemas a direcciones IP. Este servicio se denomina Domain Name System (DNS). El objetivo básico del servicio es traducir un nombre de sistema a una dirección IP.

En esta etapa de la práctica vamos a configurar el sistema DNS en nuestra empresa para que los usuarios (tanto de nuestra empresa como los clientes de Internet) puedan usar nombres en lugar de IPs.

1. Conceptos básicos de DNS

Hostname y Fully Qualified Domain Name (FQDN)

Los nombres de los sistemas deberían ser únicos en Internet para garantizar que no haya ambigüedad al obtener una IP. Por eso, el nombre de un sistema se compone de un **hostname** y de un **domain name**, separados por el carácter '.' (punto). A su vez, el *domain name* puede estar compuesto de varios *componentes de dominio* separados por puntos ya que, para poder recordarlos con facilidad, los dominios se organizan como una jerarquía (árbol) de nombres.

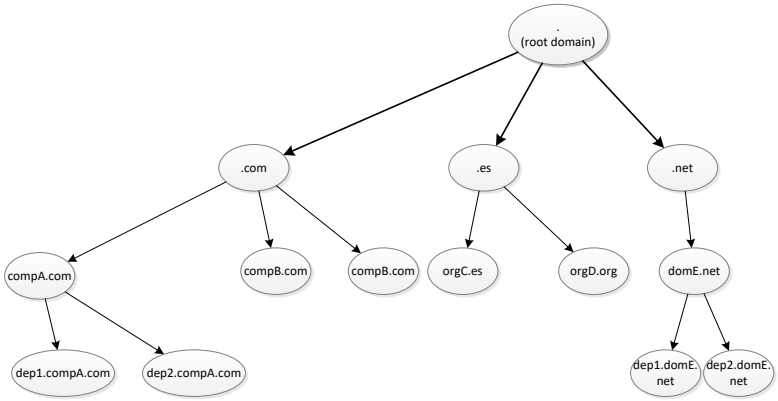
`www.gerencia.miempresa.com`

- Hostname*: www*
- *Domain name:* `gerencia.miempresa.com`
- Domain components*:* `gerencia, miempresa`
- *Top-Level Domain (TLD): com*

Se denomina *Fully Qualified Domain Name* (**FQDN**) a una cadena formada por el *hostname* y *domainname* completos de un sistema. El *FQDN* debe ser único globalmente en Internet. Cualquier otra forma del nombre (solo el *hostname* o solo una parte del *domainname)* no es fiable ya que no garantiza que sea único).

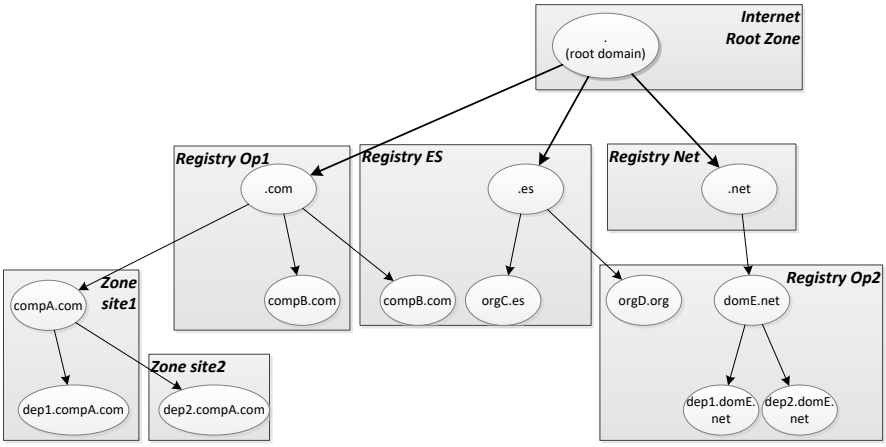
DNS Zone

Una empresa que posee un dominio de Internet puede tener subdominios debajo de su dominio. Los subdominios constituyen un árbol como en el ejemplo siguiente:



Una **zona DNS** es un conjunto de subdominios que son gestionados por el mismo administrador por razones comerciales, políticas o técnicas. Esto suele implicar que todos los subdominios de una zona están implementados en el mismo *servidor DNS* (ver más adelante). La administración de un dominio se puede delegar en una empresa especializada que se denomina un *operador de registro* (**registry operator**). Muchos ISP suelen ofrecer servicios de operador de registro.

Por ejemplo, el árbol de la figura anterior podría estar administrado por diversos operadores que gestionan zonas que no coinciden con subárboles, sino con el conjunto de dominios que han decidido delegarles la gestión de sus dominio. Por otro lado, una empresa muy grande puede gestionar su propia zona e incluso haber dividido su árbol de dominios en varias zonas independientes dentro de la empresa.



Resource Records

La información que gestiona DNS es parecida a una base de datos en la que a partir de una clave primaria se puede obtener la información asociada. La información sobre cada *hostname* está compuesta de varios *resource records*. Cada registro tiene una serie de campos fijos:

- **NAME**: Es el domain name al que pertenece la información del registro (es como la clave primaria).
- **CLASS**: Este campo estaba pensado para permitir almacenar información de distintos sistemas de información de gestión durante los años 80, pero la mayoría de los sistemas desaparecieron, por lo que la **el campo clase es siempre IN** (*Internet*).
- **TTL**: Este campo indica el tiempo máximo que se puede guardar este registro en una cache intermedia hasta que caduca y debe ser leído de nuevo.
- **RDATA**: Es el contenido del registro. Normalmente es una cadena de texto cuyo valor depende de la información asociada al tipo del registro.
- **TYPE**: Indica el tipo de información que contiene la cadena RDATA del registro. Hay bastantes tipos predefinidos, pero muchos de ellos fueron pensados en los años 80 para proporcionar información en un entorno donde no había hackers. Ahora no se usan por razones obvias. Por otro lado, se han añadido algunos tipos nuevos para soportar firmar la información electrónica entre servidores, aunque no son obligatorios. Entre los tipos más importantes destacan:
    - Name servers (**NS**): Contiene el nombre de un **servidor DNS de una zona** que se encarga del servicio DNS para el dominio indicado. Funciona como un puntero para referenciar a otro servidor al que se debe preguntar. El uso de este registro sirve para **delegar un subdominio en otro servidor.** El cliente DNS debe usar el servidor indicado por el registro NS para poder seguir traduciendo un *hostname* del subdominio indicado.

Existe un nombre de dominio especial que es '.' (punto). Por definición, todos los FQDN terminan en un punto. Si delegamos este dominio en un servidor mediante una entrada NS, estamos indicando cómo resolver todos los nombres de dominio para los que no hayamos especificado una entrada NS. Por hacer un símil, es el equivalente en DNS de la ruta 0.0.0.0/0 en una table de rutas IP.

    - Start of authority (**SOA**): Contiene información administrativa sobre la una zona DNS. La presencia de esta entrada indica que este es el *servidor oficial* (**authoritative server**) de una zona. Cuando un servidor tiene esta entrada, normalmente la encontramos junto con **una entrada NS que apunta a sí mismo**.
    - IP addresses (**A** and **AAAA**): Contiene la dirección IPv4 (A) o IPv6 (AAAA) que corresponde al nombre. Puede haber varios registros A y AAAA para un mismo nombre, lo que permite proporcionar a un cliente varias IP posibles para un mismo servicio (sistemas distribuidos).
    - Domain name aliases (**CNAME**): Contiene un **alias** del nombre indicado. El cliente debe continuar su búsqueda (resolución de nombre) volviendo a empezar con el nuevo nombre obtenido hasta que obtenga un registro de tipo **A**. Funciona como un puntero a un nuevo nombre. Sirve para definir

virtual hosts. Es decir, diferentes *hostnames* que son implementados por la misma máquina física.

- SMTP mail exchangers (**MX**): Contiene el nombre de un servidor de email SMTP para el nombre indicado (normalmente un dominio). Si hay varios servidores, se puede especificar la prioridad de cada uno de ellos añadiendo un número entero. Los servidores SMTP usan estos registros para entregar cada email a su servidor correspondiente.
- Pointers for reverse DNS lookups (**PTR**): Este registro contiene un nombre DNS asociado a una IP. Se usa para la traducción inversa de IP a nombre (**reverse lookup**).
- Contenido de texto libre (**TEXT**): Este registro contiene un texto sin propósito concreto. Es simplemente información en formato libre. Sin embargo, este registro ha sido usado extensivamente por algunas empresas para extender la información de DNS sin necesidad de agregar nuevos tipos al estándar. Por ejemplo:
  - Microsoft obliga a las empresas que tengan su propio dominio de email a crear en su DNS una entrada TXT que indique la dirección IP de su servidor de eMail SMTP como la siguiente. Si no existe la entrada, Hotmail rechaza cualquier como **spam** cualquier email enviado desde dicha IP.

```
miempresa.com IN TEXT "v=spf1 ip4:60.200.100.30 -all"
```

  - Google hace algo parecido a Microsoft con su GMail. Exige crear en su web una clave para nuestro dominio, y después hay que publicarla en una entrada de tipo MX como en este ejemplo. En caso contrario, rechaza todo el correo desde nuestro dominio como spam.

```
miempresa.com IN TEXT "google-site-verification=X82_bWARVymzyw2uRECs"
```

Ejemplo real

Como ejemplo de uso de estos registros puedes observar la respuesta del comando `dig` (más completo que `nslookup`) en Linux pidiendo todos los *DNS resource records* disponibles para el dominio *uma.es*.

```
$ dig -t ANY uma.es +multiline
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36881
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 13, ADDITIONAL: 8
;; QUESTION SECTION:
;uma.es.                        IN ANY
;; ANSWER SECTION:
uma.es.                 7200 IN SOA osiris.uma.es. hostmaster.uma.es. (
                                2018057895 ; serial
                                7200       ; refresh (2 hours)
                                7200       ; retry (2 hours)
                                7200       ; expire (2 hours)
                                7200       ; minimum (2 hours)
                                )
uma.es.                 3600 IN CAA 0 issue "digicert.com"
uma.es.                 3600 IN CAA 0 issue "sectigo.com"
```

```
uma.es.                                 3600  IN  TXT  "cisco-ci-domain-
verification=696128e15469f31f229f45c052b6fa5f1555428c7d4e4d0fab03fcd0ed1
2c43e"
uma.es.                         3600  IN  TXT  "v=spf1  ip4:150.214.47.64/27
ip4:150.214.47.224/28 ip6:2001:0720:0c20:cc05:b1ff::/80 -all"
uma.es.                 3600  IN  TXT  "MS=ms52767512"
uma.es.                                 3600  IN  TXT  "google-site-
verification=X82_bWARVymzyw2uRECsPRN0GR4AmKyHElQ_xEOmy0Q"
uma.es.                 7200  IN  MX  50 correo3.uma.es.
uma.es.                 7200  IN  MX  10 correoe2.uma.es.
uma.es.                 7200  IN  MX  5 correoe1.uma.es.
uma.es.                 7200  IN  NS  dns2.gssi.es.
uma.es.                 7200  IN  NS  chico.rediris.es.
uma.es.                 7200  IN  NS  sun.rediris.es.
uma.es.                 7200  IN  NS  dns1.gssi.es.
uma.es.                 7200  IN  NS  dns2.cica.es.
uma.es.                 7200  IN  NS  osiris.uma.es.
uma.es.                 7200  IN  NS  dns1.cica.es.
uma.es.                 7200  IN  NS  isis.uma.es.
;; Received 3477 bytes from 150.214.5.83#53(dns1.cica.es) in 5 ms
```

En este otro ejemplo puedes ver todos los *DNS resource records* disponibles para el host *www.uma.es*, que resulta ser un CNAME que apunta a *ccuma1i.sci.uma.es* cuyas direcciones IPv4 e IPv6 obtenemos con otra consulta:

```
$ dig -t ANY www.uma.es +multiline
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 56253
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 13, ADDITIONAL: 27
;; QUESTION SECTION:
;www.uma.es.             IN ANY
;; ANSWER SECTION:
www.uma.es.             6006 IN CNAME ccuma1i.sci.uma.es.

$ dig -t ANY ccuma1i.sci.uma.es +multiline
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 53642
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 13, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;ccuma1i.sci.uma.es.    IN ANY

;; ANSWER SECTION:
ccuma1i.sci.uma.es.     744 IN AAAA 2001:720:c20:cc05:beb:0:40:97
ccuma1i.sci.uma.es.     5074 IN A 150.214.40.97
```

Componentes del sistema DNS

El sistema DNS consta de dos **tipos de entidades**:

- **El DNS resolver**: El *resolver* es una librería cliente de DNS presente en **todo sistema** que necesite traducir (**resolver**) nombres a direcciones IP. Esta librería hace la traducción enviando una consulta del protocolo a un servidor DNS, de forma que la traducción realmente depende de un servidor externo. La librería suele estar integrada en el kernel de los sistemas operativos ya que los nombres son considerados una forma tan necesaria de direccionamiento como las direcciones IP, de manera que existen system calls POSIX que permiten traducir nombres a direcciones.
  El resolver suele tener la capacidad de guardar las respuestas (caching) para acelerar la traducción repetida del mismo nombre. La cache DNS debe respetar los parámetros especificados de ***TTL*** especificados por el servidor DNS.

La configuración de un cliente de DNS suele tener muy pocos parámetros. Estos parámetros suelen configurarse en el servidor DHCP para configurar todos los clientes de la empresa:

- **Dirección de uno o varios servidores DNS**: Es el parámetro fundamental que nunca puede faltar. El cliente contacta al servidor usando el puerto 53 UDP. Si el cliente tiene configurados varios servidores puede consultarlos bien en un orden fijo o bien usar todos mediante round-robin. El algoritmo concreto depende de la implementación del cliente en cada sistema operativo. El objetivo de usar varios servidores es conseguir alta disponibilidad.
- **Dominio por defecto del cliente (default domain)**: Este parámetro no es obligatorio, pero facilita la vida de los usuario. Si el usuario introduce un nombre sin dominio (sin ningún punto en la cadena) se asume que ha escrito un **unqualified name**, es decir, un nombre sin dominio. En este caso, antes de enviar el nombre a traducir al servidor, se añade a la cadena el *dominio por defecto* para formar un **FQDN**, ya que los servidores DNS solo son capaces de traducir nombres *FQDN* y nunca *unqualified names*.
- **Search domains**: Si el nombre no es encontrado en el **default domain** el administrador puede definir varios dominios alternativos donde busca el nombre. Por ejemplo, si no encontramos un nombre en nuestro *default domain site1.empresa1.com*, podríamos especificar que se buscara también en *site2.empresa1.com* y *site3.empresa1.com*. Sin embargo, esta práctica es considerada algo peligrosa, porque abre la puerta a la ambigüedad en la traducción de un nombre y eso puede ser usado por lo hackers. Por ejemplo, si un administrador configura un sistema usando un *unqualified name* como **backup**, podría pasar que si falla temporalmente el servidor DNS local, el sistema trate de resolver el nombre en otro dominio de búsqueda y haga que nuestro sistema termine contactado con el servidor equivocado. Un hacker podría hacer fallar nuestro servidor local mediante una *denegación de servicio* (*DoS attack*) para conseguir esto y declarar el nombre en otro servidor más vulnerable que esté en la lista de **search domains**.

- **El servidor DNS**: El servidor DNS es un servicio que recibe paquetes de tipo **query** desde los clientes y devuelve paquetes de tipo **reply**. El servidor almacena un base de datos compuesta de ***un conjunto de resource records*** para cada *subdominio* para el que tiene que resolver nombres.

  La petición del cliente es básicamente una tupla con un hostname, un tipo y una clase (siempre IN). Por ejemplo, *{hostname, A, IN}*.

  La respuesta del servidor es una lista con todos los *resource records* del tipo y la clase indicados encontrados para el *hostname*.

  Hay varios protocolos de transporte para los paquetes con preguntas y respuestas.
  - El más usado es DNS sobre UDP. El servidor DNS escucha en el puerto UDP:53, mientras que el cliente puede usar cualquier puerto UDP.
  - También puede usarse TCP con el servidor escuchando en el puerto TCP:53.
  - Existen muchas variantes como DNS sobre TLS (*DoT*), DNS sobre HTTPS (*DoH*) y otras menos estándares.

Recursive y caching name servers

El proceso de resolución de un nombre es un algoritmo sencillo donde el cliente va iterando recursivamente sobre toda la cadena de servidores hasta llegar al *authoritative server*:

- El cliente tiene definido un servidor DNS por defecto al que envía la petición inicial.
- Si el servidor DNS tiene la entrada buscada (*tiene la entrada SOA para ese dominio)*, la devuelve en la respuesta al cliente.
- Si el servidor DNS no tiene la entrada, retorna una entrada *NS* sugiriendo al cliente que pruebe con otro servidor DNS, que podría ser un servidor del dominio raíz ('.').
- El cliente prueba el servidor DNS indicado y sigue las sugerencias de los servidores hasta que por fin llega al servidor *authoritative* (SOA) del dominio y éste responde positiva o negativamente (nombre desconocido).

Como puede verse, aunque el cliente tenga capacidad de *caching* de los resultados, cada resolución de un cliente nuevo pasa por los servidores raíz de internet, lo que los obligaría a resolver muchos millones de peticiones por segundo. Para evitarlo esto se usa una técnica ligeramente distinta:

- Los servidores DNS **implementan recursividad**. Es decir, en lugar de redirigir al cliente a otro servidor, se encargan ellos mismos de buscar la respuesta a la petición del cliente ejecutando ellos el algoritmo recursivo descrito arriba.
- Además, cada servidor DNS es capaz de hacer **caching** de las respuestas obtenidas de otros servidores DNS durante el TTL definido en cada respuesta. De esta forma, evita repetir preguntas muy frecuentes. Los TTL con frecuencia son de un día (86400s) o incluso de varias semanas.

Con este algoritmo, la mayoría de las peticiones de los clientes son resueltas directamente por el servidor DNS de su propia zona DNS. Además, para cualquier empresa, los nombres de sus servicios estarán cacheados en miles de servidores DNS intermedios a lo largo y ancho de Internet. Por ejemplo, si un ISP tiene varios millones de clientes, su servidor DNS resolverá los nombres de miles de empresas durante días sin necesidad de volver a preguntar siquiera a los servidores raíz de Internet.

El lado negativo del caching es que si el administrador de una empresa quiere cambiar la IP de pública de un servicio de su empresa, el cambio puede tardar incluso semanas en propagarse, con el riesgo de que los clientes tengan problemas durante un intervalo de tiempo muy largo.

La solución para evitar esto consiste en que mucho antes de realizar un cambio (por ejemplo, días antes), el administrador reduce el TTL de la zona DNS a solo unos minutos. Pasados el tiempo del TTL anterior, cuando cree que todos los servidores DNS de Internet han refrescado la entrada y solo la guardan unos minutos, puede cambiar el TTL a unos segundos y al cabo de unos minutos podrá realizar el cambio en las entradas DNS, que se propagarán de forma casi instantánea.
Después de hacer los test correspondientes sobre el cambio y ver que todo funciona, puede volver a poner un TTL de varios días para que los clientes dejen de refrescar las entradas DNS cada pocos segundos.
Muchos operadores de registro permiten contratar servicios especiales para propagar inmediatamente los cambios de DNS de forma inmediata a una red de servidores DNS estratégicos repartidos por Internet que garantizan que los clientes refresquen en menos tiempo los cambios.

2. Split DNS horizon (horizonte DNS dividido)

Antes de pasar a implementar nuestro sistema DNS, debemos considerar otro aspecto de la traducción. Se trata de un problema lateral generado por el uso de NAT para ofrecer un servicio desde nuestra empresa a Internet.

- El servidor de nuestra empresa está ubicado dentro de la red del data center y tiene una dirección IP privada (por ejemplo 10.x.x.x).
- El router de Internet mapea mediante NAT dicha dirección privada a una dirección IP pública asignada por nuestro ISP (por ejemplo 80.3.15.21).

Sin embargo, todos los usuarios de nuestro servicio usan el mismo nombre (por ejemplo, www.site1.enterprise1.com), pero:

- El usuario que consulta desde un departamento de Intranet debe recibir la dirección IP privada para poder conectar.
- El usuario que consulta desde Internet debe recibir la dirección IP pública.

En este escenario, el servidor DNS debe dar una respuesta distinta **dependiendo de la dirección IP del cliente**. Esto se denomina *horizonte DNS dividido* (**split horizon**) y consiste en que el servidor tiene varias reglas que asocian una base de datos distinta para cada grupo de direcciones. En el **Split horizon** no solo se pueden dar direcciones IP distintas para el mismo nombre según la ubicación del cliente, sino que también se puede limitar el conjunto de *hostnames* que serán resueltos para evitar publicar demasiados detalles de la red de la empresa a Internet. Es decir, que desde Internet solo se puedan ver los nombres estrictamente necesarios.

Los servidores DNS reales implementan el horizonte dividido mediante directivas en las que se asocian **patrones de direcciones IP** (IP base y máscara) de los clientes que hacen consultas y tablas con *resource records*. Se pueden definir cualquier número de horizontes que necesitemos (no está limitado a dos) definiendo una lista de patrones y tablas.

En nuestro proyecto, Packet Tracer no implementa el horizonte dividido en el servidor DNS, por lo que usaremos dos servidores DNS separados. Uno para resolver las consultas de los sistemas del interior de nuestra empresa (DNS-Interno) y otro para resolver las consultas de los sistemas que estén fuera de nuestra empresa (DNS-Externo).

3. Servidor DNS interno

El primer paso de la configuración será configurar nuestro servidor DNS Interno para que resuelva las consultas de los clientes Internos. Los pasos a seguir serán:

- Renombrar el servidor a DNS-Int.
- Asegurarnos de que el servidor es accesible a nivel IP (ICMP ping) desde cualquier PC de departamento y también desde los servidores web.
- Configurar los parámetros (servidor DNS y default domain) del *resolver* de los PCs en el DHCP que ofrecer el router LAN.
- Configurar los parámetros del resolver de los servidores web de forma estática.
- Actualizar los *resource records* del servidor DNS para que sean coherente con el dominio de nuestra empresa y las IPs privadas de nuestros servicios web.

- Añadir las referencias necesarias para que nuestro DNS Interno pueda resolver recursivamente los nombres del resto de sites de nuestra empresa.
- Añadir las referencias necesarias para que nuestro DNS Interno pueda resolver recursivamente los nombres de Internet a través de nuestro ISP.
- Testear la configuración.

Obtener tu configuración DNS

Puedes encontrar todos los parámetros que necesitas en la tabla Excel con los datos de tu ISP. Hay una tabla que indica el nombre de dominio para tu site y otros datos que necesitarás.

Configurar los clientes DNS

El primer paso, además de renombrar el servidor interno, es comprobar que nuestro servidor DNS interno es accesible. Para eso, verifica que:

- Los PCs de todos los departamentos pueden llegar a él con ICMP ping.
- Lo mismo para los servidores web (layer de presentación) de nuestras aplicaciones.
- Verifica también que puede ver a los servidores DNS del resto de sites de tu empresa (o al menos el máximo número posible de ellos).
- Por último, verifica que tu servidor DNS puede ver **el servidor DNS de tu ISP** (puedes encontrar su dirección en la tabla Excel con los datos de tu ISP). Ojo que **el servidor DNS del ISP no es lo mismo** que el **router del ISP**.

Después de esto, ve al router LAN y para cada pool de DHCP, añade dos líneas para definir el servidor DNS y el dominio por defecto en los PCs (el dominio por defecto es el nombre de dominio para tu site). Por ejemplo:

```
ip dhcp pool 100
 dns-server 10.1.0.16
 domain-name site1.enterprise6.com
ip dhcp pool 101
 dns-server 10.1.0.16
 domain-name site1.enterprise6.com
ip dhcp pool 102
 dns-server 10.1.0.16
 domain-name site1.enterprise6.com
```

Una vez actualizado el router LAN recarga la configuración en los clientes DHCP y verifica que aparece el servidor DNS.

Haz lo mismo en los servidores de forma estática. Para definir el servidor DNS tienes que ir al apartado Config/Settings y cambiar el servidor DNS (no se puede cambiar el default domain). Ojo, que en los servidores con dos interfaces, solo tiene sentido poner el servidor DNS en uno de ellos, en el que tiene conectividad con el exterior.

Actualizar resource records en el servidor DNS Interno

Para convertir nuestro servidor DNS en un verdadero *authoritative server* de nuestro dominio necesitamos lo siguiente:

- Borrar las entradas que no tengan sentido. Por ejemplo los servidores inaccesibles como web-logic y db.
- Actualizar las IPs de los servidores accesibles a las actuales.

- Añadir una entrada **SOA** para el dominio de tu propio site. Los campos necesarios son:
  - **Primary Server Name**: Hay que poner el *hostname* del servidor DNS de esta zona (no la IP).
  - **Minimum TTL**: Es el TTL para forzar la expiración de las caches DNS. Mientras estés haciendo cambios debes poner un valor bajo, como por ejemplo 30s o 60s para no tener que esperar demasiado a ver los cambios. Si pones un valor demasiado bajo, es posible que no te funcione porque a la simulación no le dé tiempo de resolver nombres cuando hay recursividad por varios servidores. Cuando todo funcione bien, puedes poner un valor más alto (86400) para que la simulación sea muy eficiente.
  - **Refresh Time (900), Retry Time (120), Expiry Time (40000)**: Estos parámetros son usados por un servidor secundario para sincronizarse con el primario. No se usan aquí para nada, pero puedes dejar los valores indicados.
  - **MailBox**: No se usa en esta simulación.
- Añadir una entrada **NS** para el dominio de tu propio site. El argumento *Server Name* es el *hostname* del servidor DNS (no la IP).
- Añadir una entrada **A** para traducir el *hostname* del servidor DNS de tu dominio a su IP.

Una vez hechos los cambios anteriores, verifica en los PCs que puedes resolver nombres con tu servidor DNS usando el comando `nslookup`. Por ejemplo, prueba a resolver la dirección de tu servidor DNS (obviamente cambia el nombre del ejemplo por el de tu dominio):

```
C:\>nslookup dns.site1.enterprise6.com

Server: [10.1.0.16]
Address:  10.1.0.16

Non-authoritative answer:
Name:   dns.site1.enterprise6.com
Address:   10.1.0.16
```

Una vez que resuelvas los nombres de los servidores web desde la consola, puedes probar a usar el navegador con nombres y verificar que puedes acceder.

No olvides documentar todo añadiendo una nueva tabla para DNS a tu documentación.

Activar la resolución recursiva de otros sites

Una vez que nuestro servidor DNS ya está funcionando, queremos que se encargue de resolver los nombres del resto de sites de la empresa. Para ello solo necesitamos añadir una entrada NS para el dominio de cada uno del resto de los sites (y la correspondiente entrada A que traduce el nombre del servidor DNS a IP).

Para ello, introduce 5 entradas NS para el resto de sites. Por ejemplo:

```
NS: site2.enterprise6.com → dns.site2.enterprise6.com
NS: site3.enterprise6.com → dns.site3.enterprise6.com
NS: site4.enterprise6.com → dns.site4.enterprise6.com
NS: site5.enterprise6.com → dns.site5.enterprise6.com
NS: site6.enterprise6.com → dns.site6.enterprise6.com
```

Y luego las entradas A necesarias. Tendrás que pedirle a tus compañeros que publiquen sus direcciones IP de sus servidores DNS internos (mejor en el foro para que todos puedan

obtenerlas). Recuerda que son direcciones privadas ya que la red entre sites comunica directamente sin pasar por Internet.

```
A: dns.site2.enterprise6.com → <ip 10.x.x.x>
A: dns.site3.enterprise6.com → <ip 10.x.x.x>
A: dns.site4.enterprise6.com → <ip ...>
A: dns.site5.enterprise6.com → <ip ...>
A: dns.site6.enterprise6.com → <ip ...>
```

Una vez configurado el servidor DNS, primero prueba a ver si tu servidor DNS puede ver los servidores DNS del resto de sites. Para ello, en la consola de comandos del servidor DNS, prueba a resolver alguna entrada del otro site que exista directamente sin pasar por tu servidor DNS. Para ello, en el comando `nslookup` se puede especificar un servidor a usar distinto al *default* mediante su IP. Por ejemplo:

```
nslookup a-web-1.site2.enterprise6.com 10.16.8.1
```

Si te funciona la resolución desde la consola de tu servidor DNS, es muy probable que tu servidor DNS ya pueda resolver direcciones del otro site recursivamente. Ahora prueba a resolver el mismo nombre desde un PC de tu red usando tu propio servidor NFS. Si funciona correctamente, ya puedes navegar por nombre desde el browser. Además, puedes mirar la cache de tu servidor DNS y ver los nombres que está coleccionando de otros servidores al usar la recursividad.

Configurar la resolución de nombres de Internet

El proceso de configuración de la recursividad es siempre idéntico: Entrada NS + entrada A.

En el caso de la resolución de los nombres de Internet necesitas:

- Añadir una entrada NS para el dominio '.' (punto) que apunte al servidor DNS de tu ISP.
- Añadir una entrada que traduzca el nombre del servidor DNS del ISP a IP. Los datos vienen con la configuración de tu ISP.

Una vez configurado, debes tener la seguridad de que tu servidor DNS puede hacer NAT de salida (usa el pool dinámico de NAT igual que el resto de los PCs). Si no lo hiciste, puedes añadir una segunda regla a la ACL de la configuración NAT que incluya la dirección de tu servidor DNS interno. Cuando estés seguro de que el servidor DNS interno puede contactar con el DNS del ISP, prueba desde la consola de tu servidor DNS a resolver un nombre de Internet manualmente.

Prueba a resolver estos nombres con nslookup usando directamente el DNS del ISP (puede tardar bastante tiempo e incluso fallar la primera vez), por lo que debes reintentarlo.

- www.potafone.com
- www.ripe.org
- www.icann.org

Si funciona desde tu servidor DNS interno, entonces ya puedes probar a resolver dichos nombres desde un PC y a navegar con un browser usando esos nombres.

No olvides añadir todas las entradas añadidas a tu documentación.

4. Servidor DNS externo

El último apartado de la práctica es implementar el servicio DNS para el **horizonte DNS externo de nuestra empresa**. Esto quiere decir implementar el servicio que proporciona los nombres DNS a los clientes que están por el lado de fuera del router de Internet y que usan direcciones públicas para acceder a los servicios que publicamos con NAT estático en el apartado anterior.

Los puntos más importantes a tener en cuenta para describir la tarea son los siguientes:

- Estamos definiendo un servicio para las consultas DNS de clientes que están **fuera de nuestra empresa** (en Internet).
- Dichos clientes usan obligatoriamente la **dirección pública** del servicio que nuestro router NAT se encarga de traducir a direcciones locales internas de forma transparente para los clientes.
- Los clientes externos **no necesitan conocer** otras máquinas y servicios internos a nuestra empresa **que no sean los servicios publicados a internet.**
- Es posible que el NAT estático haya asociado distintos puertos de la misma dirección pública a distintos servidores internos. Por tanto es posible que los nombres DNS de distintos servicios apunten a la misma IP pública.

De lo anterior se deduce que necesitamos un nuevo servidor de nombres, diferente al servidor interno y que tiene una base de datos totalmente distinta:

- El servidor DNS externo solo tiene relación con el interno en el hecho de que traduce nombres del mismo subdominio DNS (con la misma cadena de dominio DNS). Como las traducciones son distintas, ambos servidores no se referencian mutuamente para nada.
- Las direcciones IP que devuelve son la dirección NAT pública del servicio.
- Solo definimos aquellos servicios que hemos publicado. Por tanto, la lista de nombres es muy breve.

Preparación del servidor DNS externo

Antes de empezar, hay que resaltar que el ISP que te ha proporcionado el acceso a Internet y el acceso a su servidor DNS también te ha indicado cual será la dirección pública donde tienes que configurar el NAT de tu DNS externo. Mira la documentación para saber la IP para el NAT de entrada antes de configurar el NAT.

Los siguientes pasos tratan la preparación del nuevo servidor:

- Creación del servidor: Necesitas un nuevo servidor para el DNS externo. Puedes poner uno nuevo, o puedes copiar el servidor DNS interno. Copiarlo puede ahorrar trabajo ya que los nombres del externo son un subconjunto del servidor Interno, pero tendrás que tener mucho cuidado de repasarlo meticulosamente y que no se queden entradas sin traducir o olvidar borrar entradas que no tienen sentido aquí.
- Puedes conectar el servidor DNS externo en la misma VLAN que el interno. Acuérdate de asignar una dirección en la documentación, ponerlo en el rack correspondiente y de configurar el puerto del ToR Switch. Testea con ICMP (ping) que puede alcanzar la IP privada del router de Internet (no del router del ISP).
- Configuración NAT. El servidor Externo **nunca inicia conexiones hacia Internet**. Solo las recibe y contesta. Por tanto, no debería poder acceder a Internet desde su consola. Para

que reciba conexiones debemos crear una asociación de NAT estático de entrada tal como se hizo en la etapa anterior. De hecho, si asociaste el puerto UDP:53 de entrada al servidor DNS interno, borra dicha asociación porque es incorrecta y defínela para el servidor DNS externo.

- Testea que tu servidor DNS externo es alcanzable desde cualquier host de Internet. Para ello vamos a usar nuestra conexión doméstica:
  - Primero, define una entrada **A** para un FQDN cualquiera con una dirección cualquiera. El objetivo no es acceder a un servicio sino verificar que traduce. Por ejemplo puedes definir el resource record `dns.test.com A` ➔ `1.1.1.1`.
  - Luego conecta tu PC doméstico (HOME) de prueba. Verifica antes con ICMP que puede llegar hasta la dirección pública de tu site en Internet.
  - Desde la consola de comandos verifica que puedes obtener la traducción del registro que acabas de crear indicando que use la dirección pública de tu servidor DNS externo. Tu router de Internet traducirá el acceso al puerto UDP:53 y debería poder llegar hasta tu servidor DNS externo.

```
C:\>nslookup dns.test.com 80.1.5.32
Server: [80.0.5.32]
Address:  80.0.5.32

Non-authoritative answer:
Name:   dns.test.com
Address:   1.1.1.1
```

  - La respuesta anterior nos indica que nuestro servidor puede ser contactado desde Internet.

Contenido del servidor DNS externo

El nuevo servidor debe ser el **oficial** (*authoritative*) para el dominio de nuestro site en Internet. Eso quiere decir que otros servidores tratarán de contactar con el nuestro cuando se les pida traducir una entrada de nuestro dominio. Por tanto, debemos prepararlo como servidor oficial con una entrada SOA para nuestro dominio y las entradas auxiliares de tipo **A** necesarias (el servidor DNS responsable del dominio).

Una vez preparado el dominio, solo tendremos que crear las entradas de los servicios web de las aplicaciones A y B, y el alias *www*.

- Crea las entradas SOA y NS para el dominio de tu site. Recuerda que el TTL de la entrada SOA no debe ser muy alto porque si cometes un error, las caches de Internet retendrán el nombre erróneo sin actualizarlo durante demasiado tiempo.
- Las entradas anteriores usan el nombre del servidor dns de tu dominio. Crea una entrada A con la **IP pública** del NAT de entrada de tu servidor DNS externo.
- Da de alta en tu dominio las entradas de tipo **A** para los host *a-web-1* y *b-web-1* con las IPs públicas del NAT de entrada de ambos servicios.
- Por último, crea un alias CNAME para la entrada *www* que apunte a *a-web-1*.

Una vez creado este contenido, comprueba de nuevo desde el PC doméstico que puedes obtener correctamente la información de tu servidor DNS externo.

Delegando el dominio DNS en tu servidor

El servidor ya está listo para operar, pero el paso más importante es que tiene que estar conectado a la jerarquía DNS de forma que los servidores que están por encima de él en la resolución de dominios le envíen las peticiones cuando se encuentren con una que pregunte por el dominio de tu site. Esto se llama ***delegar el dominio DNS en un área DNS***.

Recuerda que la búsqueda es un proceso recursivo, por lo que la delegación sigue una jerarquía:

- La delegación comienza en el servidor del TLD (*top level domain*). Si un cliente busca el nombre *host.site1.enterprise6.com* el **servidor del root level domain** (".") delegará la petición en el servidor del TLD ".com" mediante una entrada **NS** que nos permite encontrar al servidor de dicho TLD.
- Como hay millones de empresas que cuelgan directamente de este TLD, el servidor debe enviarnos al servidor correspondiente a la empresa. Para ello, el servidor del dominio ".com" contiene millones de entradas **NS** que apuntan a los servidores delegados. En muchos casos, el destinatario no es el servidor de la empresa, sino del operador ISP que la empresa ha contratado como ***registry operator***. En este caso, el NS de nuestra entrada ".enterprise6.com" apunta a *dns.potafone.com*.
- En el servidor DNS de nuestro **registry operator** (Potafone) existe una entrada **NS** por cada site que delega cada subdominio a una de las IPs públicas que ha contratado la empresa para sus sites. Aunque cada site ha contratado un bloque de IPs, el *registry operator* ha elegido una de ellas expresamente para el DNS de cada site.

Es decir, que nosotros no podemos *registrar* nuestro dominio en Internet, sino que debemos acudir a un operador de registros que es el que se encarga de pedirle al resto de operadores de los dominios que den de alta el dominio en los distintos niveles y lo deleguen en la IP correcta.

La buena noticia es que este paso ya está hecho en esta práctica porque el operador ISP al que hemos encargado el bloque de IPs y el dominio ha cumplido su trabajo y lo ha documentando en la información que nos ha pasado.

Para verificar que nuestro dominio está ubicado en la IP correcta y que funciona, basta con repetir el comando de prueba, pero usando el DNS del ISP. Para eso basta repetir el comando de prueba desde el PC doméstico preguntando, pero retirando la dirección de nuestro DNS externo. El PC usará el DNS del operador y seguirá toda la jerarquía de Internet para encontrar nuestro servidor DNS.

```
C:\>nslookup b-web-1.site1.enterprise6.com

Server: [81.0.255.253]
Address:  81.0.255.253

Non-authoritative answer:
Name:   b-web-1.site1.enterprise6.com
Address:   81.1.5.1
```

En el ejemplo anterior puedes ver que la dirección del DNS usado es del ISP, pero que devuelve el nombre que acabas de definir ya que está resolviendo recursivamente usando tu servidor.

Nota**: No uses el nombre dns.<tu dominio> como prueba ya que este nombre te puede llevar a engaño. Dicho nombre está definido en el DNS del operador como complemento a la entrada **NS

*que apunta a tu servidor externo. No se está obteniendo de tu servidor DNS externo sino del servidor DNS del ISP. Podría pasar que tu servidor no esté accesible y sin embargo creas que funciona cuando en realidad se está obteniendo de otro nivel de la jerarquía. Prueba mejor con otro nombre como **www**.*

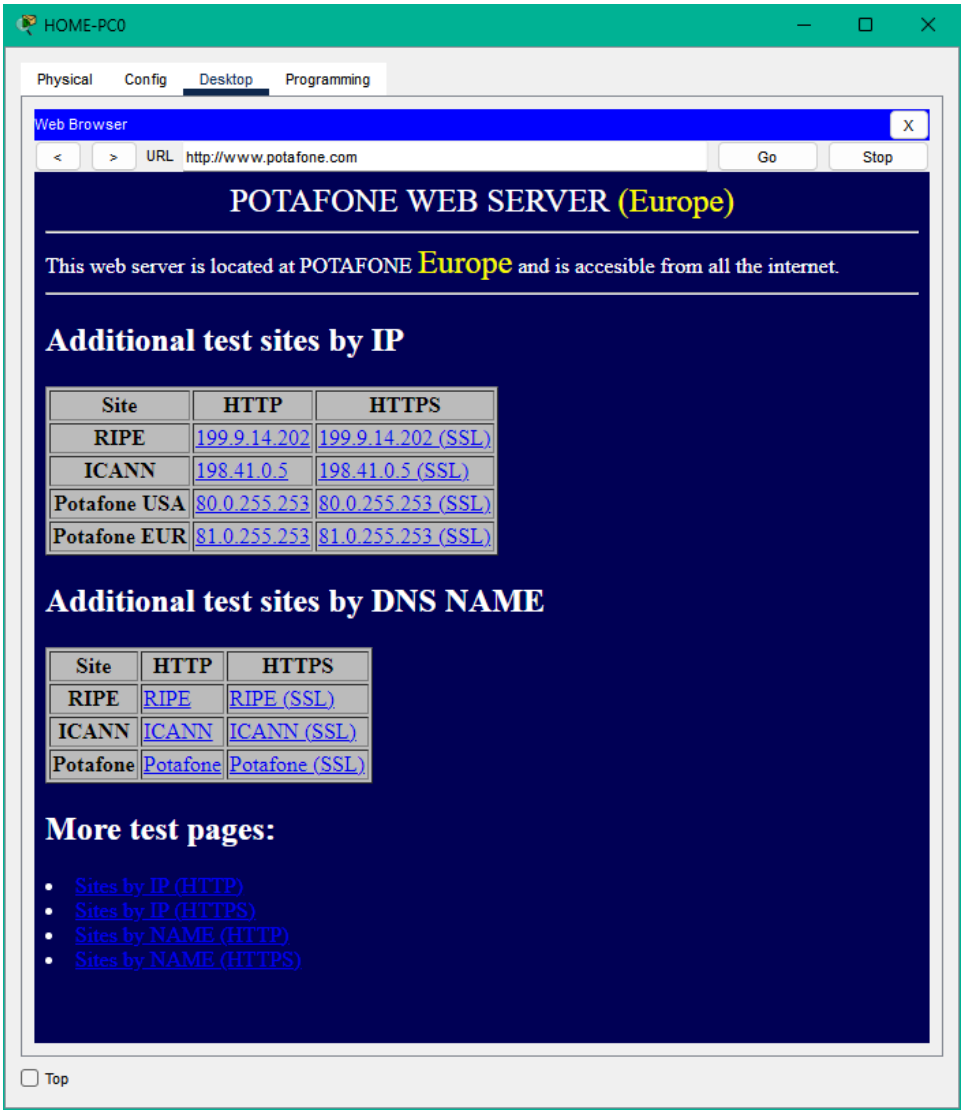
5. Pruebas sistemáticas de DNS interno y externo

Si has llegado hasta aquí, todo el trabajo de configuración (excepto la seguridad, está completo). Puedes hacer una prueba sistemática de toda tu configuración ayudado de las siguientes pruebas.

Desde cualquier PC con acceso a Internet (en Packet Tracer) puedes acceder a la página web del ISP tanto en Europa como en USA:

- 80.0.255.253 (www.potafone.com)
- 81.0.255.253 (www.potafone.com)

En la parte de debajo de la página aparecen diversos tests que puedes hacer:



Pruebas de acceso desde el Interior de la empresa a Internet

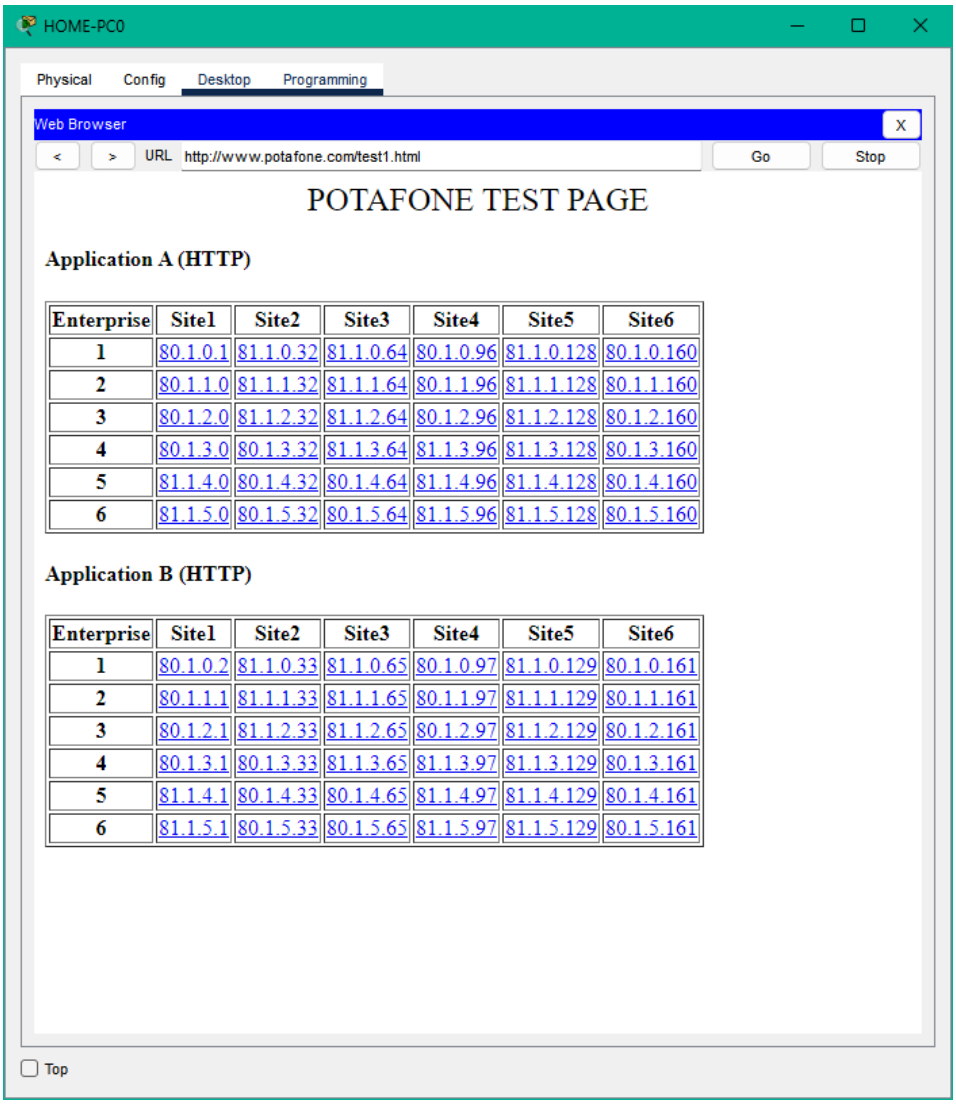
La primera tabla sirve para probar el acceso desde los PCs de los departamentos de tu empresa a varios sites fuera de la misma mediante IP (sin usar DNS). Si todos los tests funcionan, tu conexión al ISP y el NAT de salida funcionan correctamente.

La segunda tabla sirve para verificar el DNS interno y la resolución jerárquica en Internet. Si todos los tests funcionan, tu DNS interno funciona correctamente.

Pruebas de acceso desde el PC doméstico a los servicios de tu empresa

La siguiente prueba requiere acceder a la página anterior desde el PC doméstico. En la parte de debajo de la página hay enlaces a 4 páginas con tests de verificación que puedes usar para comprobar si te funciona el NAT de entrada y el DNS de entrada.

Las dos primeras presentan enlaces web para acceder a los servicios de cada site y empresa sin usar DNS. Para ejecutar el test, prueba a pinchar **solamente** en el enlace que corresponde con tu site y empresa. Prueba la aplicación A y luego la B, suponiendo que has asociado el puerto 80 de ambas a la primera y segunda direcciones IP públicas de las que has recibido.



Si el test te funciona para ambos servicios, ambos servicios HTTP son accesibles desde Internet.

El siguiente enlace da una tabla idéntica, pero usa **https** en las URLs. Si este test funciona para ambas aplicaciones, ambos servicios HTTPS son accesibles desde Internet.

El tercer enlace vuelve a dar una tabla idéntica, pero en este caso las URLs usan el nombre DNS de tu site para ambos servicios (www y b-web-1). Si este test funciona, tu DNS externo está resolviendo los nombres de tus servicios correctamente.

El cuarto enlace usa los nombres con **https** en las URLs. Obviamente este test no aporta nada a los tres anteriores ya que la resolución DNS del *hostname* es idéntica para http y https.

Evidentemente, si pruebas otros sites que no sean el tuyo, el resultado no depende de ti sino de si el otro site está bien configurado y además está conectado en este momento.

Pruebas de resolución DNS entre sites

Para comprobar la resolución DNS a través de los enlaces WAN con los demás sites de tu empresa puedes usar las páginas 3 y 4 de los test anteriores desde un PC de un departamento de tu site.

Sin embargo, dicha prueba puede dar resultados inesperados. Nuestro PC usa el DNS interno. Cuando el PC trate de resolver un *hostname* de otro site pueden pasar dos cosas:

- Que decida usar la entrada NS que apunta al DNS interno del otro site, lo que devolverá una dirección local. El acceso irá a través de la red WAN.
- Que decida usar la entrada NS que apunta al root domain (“.”) que apunta al DNS del ISP, lo que devolverá una dirección pública. El acceso irá a través de Internet.

Efectivamente. El resultado depende del orden en que nuestro servidor DNS procese las entradas y abre la puerta a situaciones muy complicadas de resolver. Tener un horizonte dividido es también una fuente de problemas adicionales.

Lo mismo ocurre si en lugar de un horizonte dividido tenemos un proxy web definido en los navegadores. Los nombres van a ser resueltos por el proxy y no por el navegador, por lo que el proxy puede acabar llevándonos a la máquina *correcta* desde su punto de vista, pero *incorrecta* desde el punto de vista del cliente.