Proyecto

Redes de Computadoras

Mini-Cliente de Consultas de Servidores DNS

**Autores**

Agra Federico | 94186

Loza Carlos | 94399

# Índice

|  |  |
| --- | --- |
| Descripción del Proyecto | 3 |
| Implementación de la Solución | 3 |
| Códigos de Retorno | 6 |
| Compilación del Programa | 6 |
| Ejecución del Programa | 7 |

# Descripción del Proyecto

Empleando el lenguaje de programación C, se implementó un programa para el sistema operativo Linux, que se utilizará en consola, el cual realiza determinadas consultas a servidores DNS. Tiene como propósito servir de herramienta de consulta y exploración de la jerarquía DNS.

**Implementación de la Solución**

El programa implementado, denominado *dnsquery.c*, es utilizaado mediante el Shell en una terminal.

El programa esta dividido en dos módulos con una funcionalidad definida:

dnsquery.c

Este es el módulo principal del programa. Se encarga de recibir los argumentos pasados por consola, verificar la validez de los mismos y luego enviarlos al módulo de consultas según corresponda.

Este módulo consta de los siguientes métodos:

* void main(int argc, char\* argv[])

Es el método principal del módulo. Recibe como parámetro los argumentos ingresados por consola y la cantidad de los mismos. Invoca al método encargado de procesar los parámetros. Luego, comienza a delegar las consultas pertinentes al módulo de consultas.

* int desglosar(char\* consul, char\*\* salida)

Recibe como parámetro una cadena de caracteres y un arreglo de strings. Este último es completado con el desglose de la consulta ingresada, la cual es dividida en partes para obtener las consultas parciales (Iterativa). Luego, retorna el tamaño del arreglo completo.

* void obtenerHost(char\* host\_argumento)

Obtiene la dirección del host a consultar y setea la variable correspondiente.

* void get\_dns\_default()

Obtiene del archivo resolv.conf de Linux el servidor DNS utilizado como defecto.

* void ayuda(int completa)

Muesta un mensaje de ayuda por consola según el modo ingresado se muestra una ayuda completa del modo de uso del programa o una ayuda parcial según los parámetros ingresados por consola.

consultar.c

Este módulo se encarga de definir las estructuras necesarias para realizar las consultas requeridas por el usuario. Esta tarea la lleva a cabo mediante la utilización de sockets, tanto para enviar solicitudes DNS como para recibir respuestas de los servidores correspondientes. Estas respuestas son filtradas según su tipo:

* A: Consulta de resolución de nombre de dominio.
* NS: Consulta sobre los servidores que contienen el dominio especificado.
* MX: Consulta sobre el servidor a cargo del correo electrónico.
* LOC: Consulta sobre la información de geo-localización del dominio.

También son filtradas respuestas de tipo CNAME y AAAA pero solo se informa que se recibió una respuesta de estos tipos en particular, pero ninguna información sobre ellos. Para los tipos especificados se muestra la siguiente información:

* Número de IPv4 del dominio (A).
* Servidores contenedores del dominio (NS).
* Nombre de dominio de los servidores a cargo de servicio de correo electrónico y su número de preferencia (MX).
* Datos referidos a la geo-localización de un dominio, tales como tamaño, precisión horizontal y vertical, latitud y longitud (mostrada en grados, minutos, segundos y cuadrante), así como también la altura (LOC).

Este módulo contiene los siguientes métodos:

* char\* consultar(hots,dns,puerto,tipo,recursion,modoPrint)

Este método recibe los datos necesarios para realizar las consultas requeridas. Crea el socket para realizar el envió y la recepción de mensajes DNS. Si se recibe una respuesta procede a leer todas los mensajes recibidos, y muestra por consola la información correspondiente a cada respuesta y su tipo. Luego, retorna la dirección del servidor recibido como respuesta (puede ser una IPv4 (preferentemente) o un nombre de dominio).

* unsigned int getLatLonAlt(unsigned int n\_lla[])

Convierte un cuarteto de enteros (4 números Hexa en representación decimal) ingresado por parámetro a su representación decimal completa.

* int convertirAMetros(int n\_hex[])

Recibe una dupla de enteros y retorna los metros según la siguiente fórmula:

( x \* 10y ) / 100;se divide para obtener en metros

Donde x es la base e y la potencia de 10 a la cual se multiplica la base.

* void pasarAHexa(int entero, int n\_hex[])

Recibe un entero de dos dígitos el cual es pasado a hexadecimal y guarda una cifra en cada celda del arreglo recibido como parámetro.

* void formatoDNS(unsigned char\* dns,unsigned char\* host)

Recibe una dirección del estilo nombre de dominio (ejemplo cs.uns.edu.ar) y la convierte al formatoDNS (ejemplo 3www2cs3uns3edu2ar0).

* u\_char\* leerHost(unsigned char\* \_lec, char \_buff[],int\*\_cont)

Lee los datos recibidos en las respuestas y los convierte en nombres de Host utilizables.

consultar.h

Se refiere a la interface del módulo secundario consultar.c, el cual es utilizado por el módulo principal del programa dnsquery.c.

Códigos de Retorno

Los códigos de salida (retorno del programa) se clasifican de la siguiente manera:

|  |  |  |
| --- | --- | --- |
| Constante | Valor | Descripción |
| EXIT\_SUCCES | 0 | Finalización exitosa. |
| ERR\_PARAM | 1 | Error de parámetros. |
| ERR\_ENV | 2 | Error al enviar un mensaje (Socket). |
| ERR\_REC | 3 | Error al recibir un mensaje (Socket). |
| ERR\_SOCK\_CREAT | 4 | Error al crear el socket. |

Compilación del Programa

Para compilar el programa se debe utilizar la siguiente linea de comando:

$ gcc consultar.c consultar.h dnsquery.c -o dnsquery -lm

Se debe agregar el parametro -lm ya que se incluye la librería Math.h, la cual debe ser enlazada mediante este parámetro.

Ejecución del Programa

Para su invocación se debe ejecutar un comando del estilo:

$ dnsquery <consulta> [@<servidor>[:<puerto>]] [-a | -mx | -loc] [-r | -t] [-h]

Donde:

* <consulta>: Es la consulta (dirección IP o nombre de Host) que se desea resolver.
* <@servidor>: Servidor DNS con el que se resolverá la consulta. Si no se determina se usa el servidor DNS por defecto.
* <puerto>: Puerto DNS con el cual el servidor que resolverá la consulta esta ligado. Si no se determina se utiliza el puerto DNS estándar (53). Debe estar definido el servidor al cual se le realizará la consulta.
* <tipo>
  + -a: Consulta de resolución de nombre.
  + -mx: Servidor a cargo de la recepción de correo electrónico para el dominio indicado en la consulta.
  + -loc: Información relativa a la localización geográfica del dominio indicado en la consulta.
* <modo>
  + -r: Consulta Recursiva.
  + -t: Consulta Iterativa.
* -h: Muestra una ayuda para su correcta invocación.