

Producto 3. Automatización básica de la administración de un sistema operativo

Rubén Vicente Gilabert

Índice

1. Objetivo del producto.....	2
2. Descripción de funcionalidades implementadas.....	2
2.1 Preguntar inicialmente el nombre y la ruta del archivo dnsips.txt y mostrar el contenido de este archivo por pantalla... 2	
2.2 Preguntar al usuario el adaptador de red que quiere modificar y mostrar los servidores DNS configurados en este.... 2	
2.3 Leer el archivo proporcionado inicialmente con las IP de los servidores DNS, y chequear si esas IPs son accesibles, y en caso de que lo sean es necesario escribirlas en un archivo temporal (esto se considerará un requisito).....3	
2.4 Abrir el archivo temporal creado en el punto anterior y chequear de entre los dns configurados en el adaptador determinado en el punto 7 y de los escritos en este archivo, cuál de estos es más rápido..... 3	
2.5 Cambiar en caso de no tener configurado el DNS más rápido, el DNS del adaptador indicado en el punto 6 por el más veloz.....4	
2.6 En caso de que dos servidores DNS empaten en velocidad, se reconfigurará el DNS por el que esté a un menor número de saltos, puesto que se considera que la cercanía es un punto a tener en cuenta..... 4	
3. Consideraciones técnicas.....	5
4. Resultados obtenidos.....	5
6. Archivos generados.....	6
7. Recursos utilizados.....	7

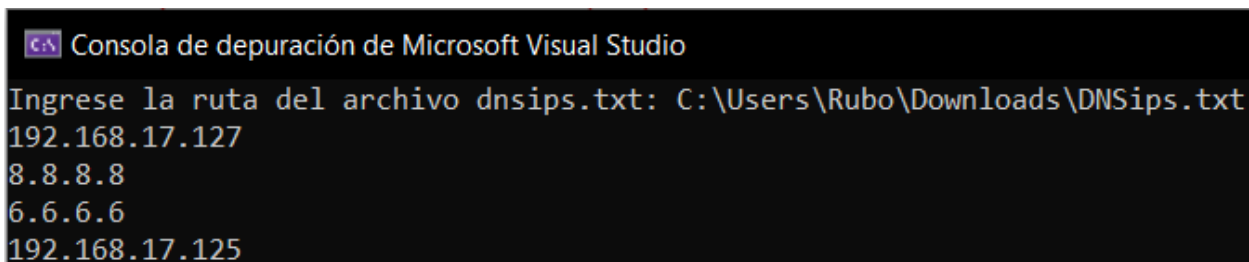
1. Objetivo del producto

- Desarrollar una pequeña aplicación escrita en C.
- Desarrollar una pequeña aplicación que sea capaz modificar la configuración de red de la máquina local cambiando el servidor DNS de esta por el más rápido entre los testeados.

2. Descripción de funcionalidades implementadas

2.1 Preguntar inicialmente el nombre y la ruta del archivo dnsips.txt y mostrar el contenido de este archivo por pantalla.

La aplicación solicita la ruta del archivo DNSips.txt

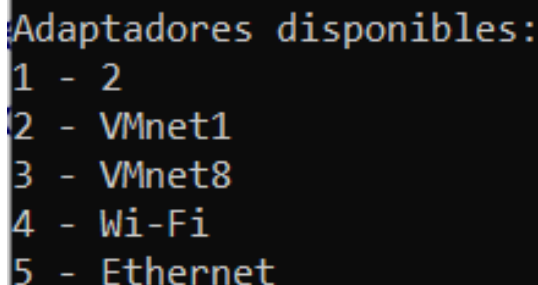


```

Consola de depuración de Microsoft Visual Studio
Ingrese la ruta del archivo dnsips.txt: C:\Users\Rubo\Downloads\DNSips.txt
192.168.17.127
8.8.8.8
6.6.6.6
192.168.17.125
  
```

2.2 Preguntar al usuario el adaptador de red que quiere modificar y mostrar los servidores DNS configurados en este.

- El programa muestra los adaptadores disponibles para que el usuario escoja el que desea configurar.



```

Adaptadores disponibles:
1 - 2
2 - VMnet1
3 - VMnet8
4 - Wi-Fi
5 - Ethernet
  
```

- El programa solicita que se seleccione un adaptador y guarda el nombre para después editar la configuración.

Seleccione un adaptador (1 - 5): 4

Nombre del adaptador seleccionado guardado: Wi-Fi

- Se muestra la configuración del adaptador seleccionado.

Configuración actual del adaptador (Wi-Fi):

```
Configuración para la interfaz "Wi-Fi"
DHCP habilitado: Si
Dirección IP: 192.168.0.182
Prefijo de subred: 192.168.0.0/24 (máscara 255.255.255.0)
Puerta de enlace predeterminada: 192.168.0.1
Métrica de puerta de enlace: 0
Métrica de interfaz: 40
Servidores DNS configurados estáticamente: 8.8.4.4
Registrar con el sufijo: Solo el principal
Servidores WINS configurados a través de DHCP: ninguno
```

2.3 Leer el archivo proporcionado inicialmente con las IP de los servidores DNS, y chequear si esas IPs son accesibles, y en caso de que lo sean es necesario escribirlas en un archivo temporal (esto se considerará un requisito).

- La aplicación crea el archivo dns_temp.txt con las IPs accesibles.



dns_temp: Bloc de notas

Archivo Edición Formato Ver Ayuda

8.8.8.8

2.4 Abrir el archivo temporal creado en el punto anterior y chequear de entre los dns configurados en el adaptador determinado en el punto 7 y de los escritos en este archivo, cuál de estos es más rápido.

Archivo: util.c

- int medirLatencia(const char* ip); → mide ping
- int contarSaltos(const char* ip); → cuenta saltos vía tracer

```
int medirLatencia(const char* ip) {
    char comando[128];
    snprintf(comando, sizeof(comando), "ping -n 1 -w 100 %s > temp_ping.txt", ip);
    system(comando); // Ejecuta el comando y guarda la salida
}
```

```
int contarSaltos(const char* ip) {
    char comando[128];
    snprintf(comando, sizeof(comando), "tracert -h 30 -d %s > temp_tracert.txt", ip);
    system(comando); // Ejecutar tracer
```

2.5 Cambiar en caso de no tener configurado el DNS más rápido, el DNS del adaptador indicado en el punto 6 por el más veloz.

- El programa muestra por pantalla la nueva configuración DNS del adaptador seleccionado.

```
DNS cambiado a: 8.8.8.8
Configuracin del adaptador despus del cambio (Wi-Fi):
Configuración para la interfaz "Wi-Fi"
DHCP habilitado: Si
Dirección IP: 192.168.0.182
Prefijo de subred: 192.168.0.0/24 (máscara 255.255.255.0)
Puerta de enlace predeterminada: 192.168.0.1
Métrica de puerta de enlace: 0
Métrica de interfaz: 40
Servidores DNS configurados estáticamente: 8.8.8.8
Registrar con el sufijo: Solo el principal
Servidores WINS configurados a través de DHCP: ninguno
```

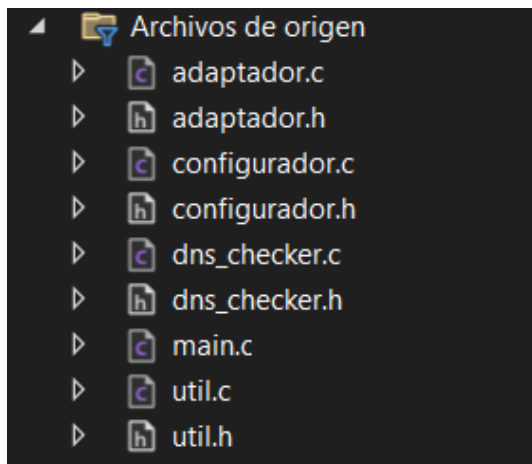
2.6 En caso de que dos servidores DNS empaten en velocidad, se reconfigurará el DNS por el que esté a un menor número de saltos, puesto que se considera que la cercanía es un punto a tener en cuenta.

El programa compara así:

- Primero: elige el DNS con menor latencia.
- Si hay empate: elige el DNS con menor número de saltos.

```
// Elegir la mejor IP:
// 1. Si es más rápida
// 2. Si empatan en latencia pero tiene menos saltos
if (latencia < mejorTiempo || (latencia == mejorTiempo && saltos < mejorSaltos)) {
    mejorTiempo = latencia;
    mejorSaltos = saltos;
    strcpy(mejor, linea); // Guardar la mejor IP hasta ahora
}
```

3. Consideraciones técnicas.



- El código ha sido modularizado en múltiples archivos .c y .h para facilitar su mantenimiento y legibilidad.

4. Resultados obtenidos

Tras ejecutar la aplicación, se observaron los siguientes pasos y resultados a través de la consola:

1. Carga y visualización del archivo dnsips.txt:

Se ingresó correctamente la ruta al archivo de IPs DNS. El contenido fue mostrado por pantalla:

```
Ingrese la ruta del archivo dnsips.txt: C:\Users\Rubo\Downloads\DNSips.txt
192.168.17.127
8.8.8.8
6.6.6.6
192.168.17.125
```

2. Selección de adaptador de red:

Se mostraron los adaptadores disponibles, y el usuario seleccionó el adaptador "Wi-Fi":

```
Adaptadores disponibles:
1 - 2
2 - VMnet1
3 - VMnet8
4 - Wi-Fi
5 - Ethernet

Seleccione un adaptador (1 - 5): 4

Nombre del adaptador seleccionado guardado: Wi-Fi
```

3. Lectura y muestra de configuración actual del adaptador:

*Antes del cambio, el adaptador tenía configurado el DNS 8.8.4.4.

```
Configuracin actual del adaptador (Wi-Fi):

Configuración para la interfaz "Wi-Fi"
DHCP habilitado:                Sí
Dirección IP:                   192.168.0.182
Prefijo de subred:              192.168.0.0/24 (máscara 255.255.255.0)
Puerta de enlace predeterminada: 192.168.0.1
Métrica de puerta de enlace:    0
Métrica de interfaz:            40
Servidores DNS configurados estáticamente: 8.8.4.4
Registrar con el sufijo:        Solo el principal
Servidores WINS configurados a través de DHCP: ninguno
```

4. Comparación de velocidad y cercanía entre servidores DNS:

El programa identificó que 8.8.8.8 era más rápido y/o cercano que 8.8.4.4.

5. Cambio de servidor DNS:

La aplicación modificó la configuración del adaptador, reemplazando 8.8.4.4 por 8.8.8.8.

```
DNS cambiado a: 8.8.8.8

Configuracin del adaptador despues del cambio (Wi-Fi):

Configuración para la interfaz "Wi-Fi"
DHCP habilitado:                Sí
Dirección IP:                   192.168.0.182
Prefijo de subred:              192.168.0.0/24 (máscara 255.255.255.0)
Puerta de enlace predeterminada: 192.168.0.1
Métrica de puerta de enlace:    0
Métrica de interfaz:            40
Servidores DNS configurados estáticamente: 8.8.8.8
Registrar con el sufijo:        Solo el principal
Servidores WINS configurados a través de DHCP: ninguno
```

6. Ejecución final:

El programa terminó su ejecución sin errores.

6. Archivos generados

Durante la ejecución, el programa genera los siguientes archivos:

- adaptador.txt: Guarda el nombre del adaptador de red seleccionado por el usuario.
- dns_temp.txt: Guarda únicamente las direcciones DNS que respondieron correctamente al ping.
- temp_ping.txt: Archivo temporal usado para medir la latencia (borrado automáticamente).
- temp_tracert.txt: Archivo temporal usado para contar los saltos (tracert) (borrado automáticamente).

7. Recursos utilizados

- Microsoft. (2023). *Netsh Command Syntax, Contexts, and Formatting*. Microsoft Learn. Recuperado de <https://learn.microsoft.com/en-us/windows-server/networking/technologies/netsh/netsh-contexts>
- Microsoft. (2023). *tracert*. Microsoft Learn. Recuperado de <https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/tracert>
- Stack Overflow. (2013). *How to ping using C sockets - Windows*. Recuperado de <https://stackoverflow.com/questions/14307203/how-to-ping-using-c-sockets>
- GeeksforGeeks. (2023). *Ping in C*. Recuperado de <https://www.geeksforgeeks.org/ping-in-c/>
- Stack Overflow. (2011). *Need a way in C++ to set the DNS servers on all interfaces*. Recuperado de <https://stackoverflow.com/questions/3955919/need-a-way-in-c-to-set-the-dns-servers-on-all-interfa>
[ces](https://stackoverflow.com/questions/3955919/need-a-way-in-c-to-set-the-dns-servers-on-all-interfa)