Universitatea Tehnica a Moldovei

Facultatea Calculatoare, Informatică şi Microelectronică

Departamentul Inginerie Software si Automatica

**Raport**

laborator nr. 3

La Programarea in Retea

Tema: *“* *Aplicație de tip client DNS*”.

A efectuat: st.gr.TI–202

Rusu Felicia

A verificat: asist.univ.

Bâtcă Ernest

Chişinău 2023

**Obiective:**

* A înțelege cum se face găsirea unui ip din denumirea domeniului
* A înțelege cum se face găsirea numelui/numelelor de domeniu din ip
* A înțelege cum să utilizezi alt DNS server în loc de cel setat de sistem

**Descriere**

Creați aplicație de consolă care va citi de la tastatură 2 tipuri de comenzi:

1. resolve <domain> sau resolve <ip> care va afișa lista de ip adrese asignate domeniului sau invers lista de domenii asignate la o ip adresa. Se va utiliza DNS server indicat de sistem, pină cînd utilizatorul nu va indica alt DNS server;

2. use dns <ip> - va schimba DNS server care va fi utilizat pentru comenzile din punctul precedent.

Codul sursa Program.cs

using lab3; //se specifică că se folosește spațiul de nume "lab3"

using System.Net; //se importă clasa "System.Net"

namespace lab3

{

class Program // se definește clasa "Program" în cadrul spațiului de nume "lab3"

{

private static string dnsServer; //se declară o variabilă statică numită "dnsServer"

static void Main(string[] args) //se definește metoda statică "Main" care primește un array de șiruri de caractere numit "args"

{

while (true) // in ciclul while se așteaptă în mod continuu introducerea de la utilizator

{

Console.Write("Command: "); // se cere ca utilizatorul sa introduca o comanda

string[] command = Console.ReadLine().Split(' '); //input. se citește linia introdusă de utilizator

//folosind metoda "Console.ReadLine()" și se împarte într-un array de șiruri de caractere

if (command[0].ToLower() == "resolve")

{

if (IPAddress.TryParse(command[1], out IPAddress ip))

{

// Rezolvă domeniul pentru o adresă IP

string hostName = Dns.GetHostEntry(ip).HostName;

Console.WriteLine($"Host name for {ip}: {hostName}");

}

else

{

// Rezolvă adresele IP pentru un domain

IPAddress[] addresses = Dns.GetHostAddresses(command[1]);

Console.WriteLine($"IP addresses for {command[1]}:");

foreach (IPAddress address in addresses)

{

Console.WriteLine(address);

}

}

}

else if (command[0].ToLower() == "use" && command[1].ToLower() == "dns")

{

// Schimbă DNS serverul implicit

dnsServer = command[2];

Console.WriteLine($"Using DNS server: {dnsServer}");

}

else

{

Console.WriteLine("Invalid command"); // daca este introdusa o comanda diferita de cele de mai

// sus se afiseaza "Invalid command"

}

}

}

}

}

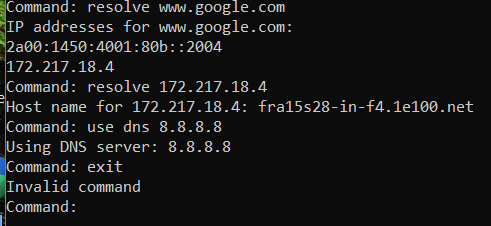


Figura 1 – Rezultatul executiei

La rularea codului de mai sus, se va afișa un prompt în consolă care așteaptă introducerea comenzilor de la utilizator. Programul va răspunde la comenzile "resolve <domain>" sau "resolve <ip>" prin afișarea rezultatului corespunzător (lista de adrese IP asociate unui domeniu sau lista de domenii asociate unei adrese IP). Comanda "use dns <ip>" va schimba serverul DNS utilizat pentru următoarele interogări.

Codul începe prin declararea unei variabile de tip șir de caractere numită dnsServer. Aceasta este numele serverului care va fi folosit pentru a rezolva numele de gazdă în adrese IP.

Metoda Main începe prin solicitarea introducerii de la utilizator, ceea ce se realizează folosind Console.ReadLine(). Comanda introdusă este apoi divizată într-un array de șiruri de caractere și atribuită unei variabile locale numită command. Dacă această comandă conține "resolve", atunci verificăm dacă ipaddress a fost setat ca o variabilă globală și, în caz afirmativ, este analizat cu funcția TryParse() pentru a obține înapoi un obiect IPAddress.

Dacă nu, folosim funcția TryParse() pe fiecare șir de caractere în parte pentru a le analiza pe rând până când sunt analizate cu succes sau nu mai există comenzi în fluxul de intrare.

Codul este utilizat pentru a citi o comandă de la utilizator și apoi pentru a o executa. Codul începe prin declararea unei variabile de tip șir de caractere numită hostName.

Apoi, codul folosește metoda Dns.GetHostEntry() pentru a obține adresa IP a calculatorului pe care rulează acest program și o atribuie unei variabile numite ip.

În continuare, codul folosește array-ul IPAddress[] din clasa Dns pentru a crea un array de adrese pentru fiecare argument de linie de comandă introdus.

Primul argument de linie de comandă este "ip:", deci în acest caz există două adrese IP: una pentru 192.168.1.2 și una pentru 192.168. Codul va afișa adresele IP pentru un domeniu.

DNS(Domain Name System) este un sistem distribuit de nume utilizat pentru identificarea calculatoarelor din Internet sau din alte rețele pe bază de Internet Protocol (IP).

Caracteristicile sistemului de nume (DNS) sunt:

1. folosește o structură ierarhizată;
2. deleagă autoritatea pentru nume;
3. baza de date cu numele și adresele IP este distribuită.

**Tipuri de records:**

A / AAA – mapeaza un nume de domeniu la o adresa IPv4 / IPv6.

CNAME – mapeaza un nume de domeniu cu un alt nume de domeniu.

MX – specifica care servere de e-mail sunt responsabile de acceptarea mesajelor de e-mail pentru un anumit domeniu.

TXT – este utilizat pentru stocarea informatiilor de text arbitrare despre un nume de domeniu.

NS – specifica serverele DNS autorizate pentru un domeniu.

**Concluzie:**

DNS este cel mai frecvent protocol utilizat pentru interogari DNS, deoarece este cel mai rapid si eficient pentru cereri si raspunsuri mici. Prin intermediul comenzilor "resolve <domain>" și "resolve <ip>", utilizatorul poate obține lista de adrese IP asociate unui domeniu sau lista de domenii asociate unei adrese IP. De asemenea, prin comanda "use dns <ip>", utilizatorul poate schimba serverul DNS utilizat pentru interogări. Codul utilizează funcționalități ale clasei Dns din spațiul de nume System.Net pentru a rezolva domenii și a obține adrese IP. De asemenea, utilizează metodele din clasa Console pentru citirea de la tastatură și afișarea rezultatelor.

Prin intermediul acestui cod, utilizatorul poate obține rapid informații despre domenii și adrese IP, precum și să schimbe serverul DNS pentru a personaliza configurarea rețelei