

# NetInfo

Software Documentation

Author: matiwa

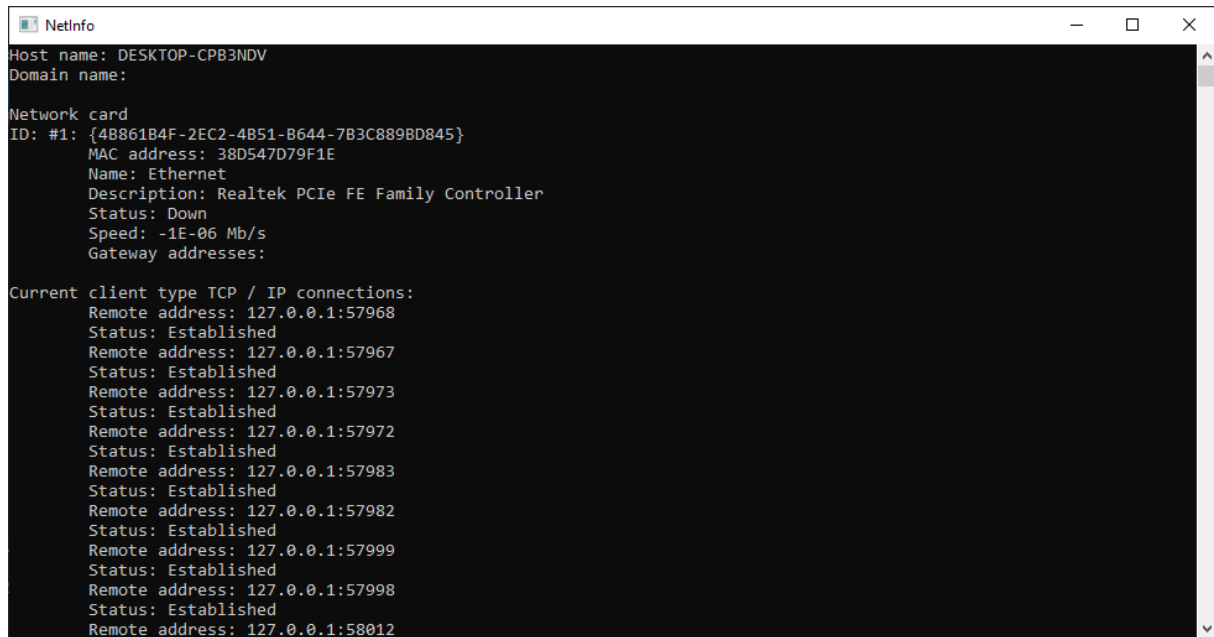
## Table of contents

Table of contents.....	2
Introduction.....	3
Describing of the application's operation.....	3
What is needed for use?.....	3
Algorithm used.....	3
Interface description.....	4
Source code description.....	4
List of drawings.....	6
List of listings.....	6

## Introduction

This application is used to display a complete set of information on all network connections on your computer, information on network interfaces, and information on current TCP and UDP network connections.

## Describing of the application's operation

A screenshot of a Windows application window titled "NetInfo". The window has a black background with white text. It displays the following information: Host name: DESKTOP-CPB3NDV, Domain name: (empty), Network card ID: #1: {48861B4F-2EC2-4B51-B644-7B3C889BD845}, MAC address: 38D547D79F1E, Name: Ethernet, Description: Realtek PCIe FE Family Controller, Status: Down, Speed: -1E-06 Mb/s, Gateway addresses: (empty). Below this, it lists "Current client type TCP / IP connections:" followed by a list of 10 entries, each showing a Remote address and Status: Established. The remote addresses are 127.0.0.1:57968, 127.0.0.1:57967, 127.0.0.1:57973, 127.0.0.1:57972, 127.0.0.1:57983, 127.0.0.1:57982, 127.0.0.1:57999, 127.0.0.1:57998, and 127.0.0.1:58012. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
NetInfo
Host name: DESKTOP-CPB3NDV
Domain name:

Network card
ID: #1: {48861B4F-2EC2-4B51-B644-7B3C889BD845}
MAC address: 38D547D79F1E
Name: Ethernet
Description: Realtek PCIe FE Family Controller
Status: Down
Speed: -1E-06 Mb/s
Gateway addresses:

Current client type TCP / IP connections:
Remote address: 127.0.0.1:57968
Status: Established
Remote address: 127.0.0.1:57967
Status: Established
Remote address: 127.0.0.1:57973
Status: Established
Remote address: 127.0.0.1:57972
Status: Established
Remote address: 127.0.0.1:57983
Status: Established
Remote address: 127.0.0.1:57982
Status: Established
Remote address: 127.0.0.1:57999
Status: Established
Remote address: 127.0.0.1:57998
Status: Established
Remote address: 127.0.0.1:58012
Status: Established
```

Drawing 1: The beginning of the application's operation [own study]

After starting the program, the user does nothing. The effects of the program's work can be seen immediately. The application displays everything about each of the user's computer network connections, network interfaces as well as current TCP and UDP network connections. There are certainly some bugs that the developer didn't discover while working on the app.

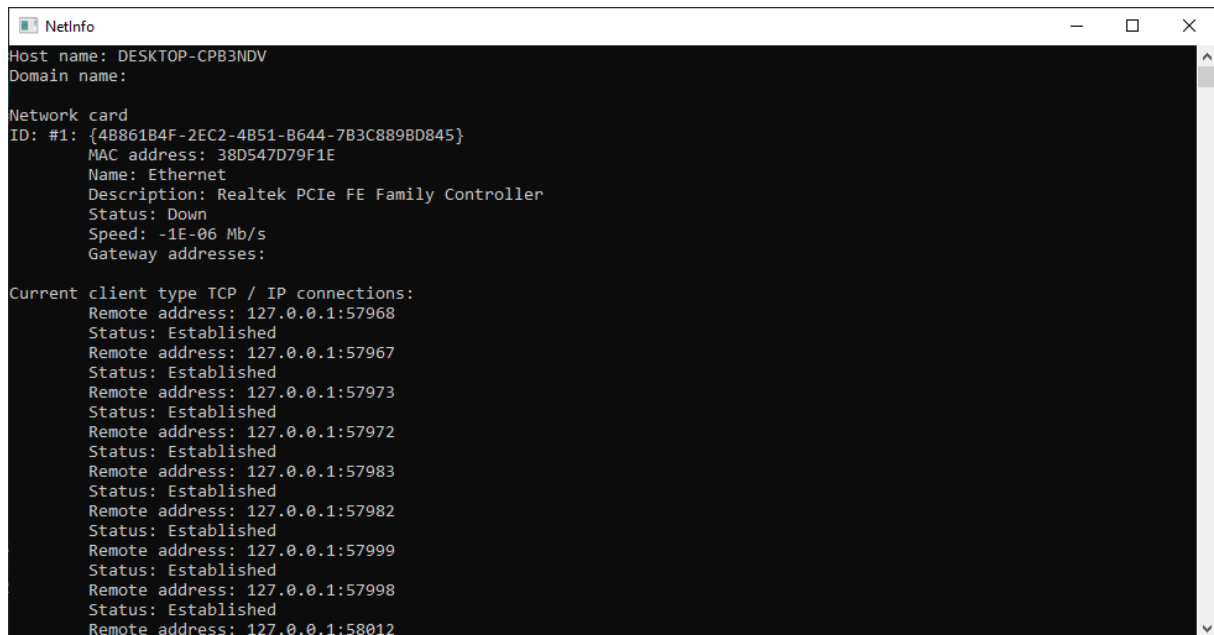
## What is needed for use?

The application does not require installation. It only needs the Windows operating system.

## Algorithm used

The basic form of the algorithm can be deduced from the previous section. It only needs the Windows operating system. In summary, the application displays a complete set of information on all network connections on your computer, information on network interfaces, and information on current TCP and UDP network connections.

## Interface description

A screenshot of a Windows console application window titled "NetInfo". The window has a black background with white text. It displays the following information: Host name: DESKTOP-CPB3NDV, Domain name: (empty). Under "Network card", it shows ID: #1: {48861B4F-2EC2-4B51-B644-7B3C889BD845}, MAC address: 38D547D79F1E, Name: Ethernet, Description: Realtek PCIe FE Family Controller, Status: Down, Speed: -1E-06 Mb/s, and Gateway addresses: (empty). Under "Current client type TCP / IP connections:", it lists ten established connections with remote addresses ranging from 127.0.0.1:57968 to 127.0.0.1:58012.

```
NetInfo
Host name: DESKTOP-CPB3NDV
Domain name:

Network card
ID: #1: {48861B4F-2EC2-4B51-B644-7B3C889BD845}
MAC address: 38D547D79F1E
Name: Ethernet
Description: Realtek PCIe FE Family Controller
Status: Down
Speed: -1E-06 Mb/s
Gateway addresses:

Current client type TCP / IP connections:
Remote address: 127.0.0.1:57968
Status: Established
Remote address: 127.0.0.1:57967
Status: Established
Remote address: 127.0.0.1:57973
Status: Established
Remote address: 127.0.0.1:57972
Status: Established
Remote address: 127.0.0.1:57983
Status: Established
Remote address: 127.0.0.1:57982
Status: Established
Remote address: 127.0.0.1:57999
Status: Established
Remote address: 127.0.0.1:57998
Status: Established
Remote address: 127.0.0.1:58012
```

Drawing 2: Graphical interface [own study]

The interface is typical for a Console Application.

## Source code description

The project was made in the C# programming language, in the Visual Studio Community 2017 programming environment. All work was done on the Windows 10 operating system. The application's source code looks like this.

```
using System;
using System.Net;
using System.Net.NetworkInformation;

namespace NetInfo
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Title = "NetInfo";
            IPGlobalProperties wlasnosciIP =
IPGlobalProperties.GetIPGlobalProperties();
            Console.WriteLine("Host name: {0}", wlasnosciIP.HostName);
            Console.WriteLine("Domain name: {0}\r\n", wlasnosciIP.DomainName);
            int licznik = 0;
            foreach (NetworkInterface kartySieciowe in
NetworkInterface.GetAllNetworkInterfaces())
            {
                Console.WriteLine("Network card");
                Console.WriteLine("ID: #{0}: {1}", ++licznik, kartySieciowe.Id);
```

```

        Console.WriteLine("\tMAC address: {0}",
kartySieciowe.GetPhysicalAddress());
        Console.WriteLine("\tName: {0}", kartySieciowe.Name);
        Console.WriteLine("\tDescription: {0}", kartySieciowe.Description);
        Console.WriteLine("\tStatus: {0}", kartySieciowe.OperationalStatus);
        Console.WriteLine("\tSpeed: {0} Mb/s", (kartySieciowe.Speed) /
(double)1000000);
        Console.WriteLine("\tGateway addresses:");
        foreach (GatewayIPAddressInformation adresBramy in
kartySieciowe.GetIPProperties().GatewayAddresses)
        {
            Console.WriteLine("\t\t{0}",
            Console.WriteLine("\t\tDNS servers:");
            foreach (IPAddress adresip in
kartySieciowe.GetIPProperties().DnsAddresses)
                Console.WriteLine("\t\t\t{0}", adresip);
            Console.WriteLine("\t\tDHCP servers:");
            foreach (IPAddress adresip in
kartySieciowe.GetIPProperties().DhcpServerAddresses)
                Console.WriteLine("\t\t\t{0}", adresip);
            Console.WriteLine("\t\tWINS servers:");
            foreach (IPAddress adresip in
kartySieciowe.GetIPProperties().WinsServersAddresses)
                Console.WriteLine("\t\t\t{0}", adresip);
            Console.WriteLine();
        }
        Console.WriteLine("\r\nCurrent client type TCP / IP connections:");
        foreach (TcpConnectionInformation polaczenieTCP in
IPGlobalProperties.GetIPGlobalProperties().
        GetActiveTcpConnections())
        {
            Console.WriteLine("\tRemote address: {0}:{1}",
polaczenieTCP.RemoteEndPoint.Address,
            polaczenieTCP.RemoteEndPoint.Port);
            Console.WriteLine("\tStatus: {0}", polaczenieTCP.State);
        }
        Console.WriteLine("\r\nCurrent TCP / IP server connections:");
        foreach (IPEndPoint polaczenieTCP in
IPGlobalProperties.GetIPGlobalProperties().
        GetActiveTcpListeners())
            Console.WriteLine("\tRemote address: {0}", polaczenieTCP.Port);
        Console.WriteLine("\r\nCurrent UDP connections:");
        foreach (IPEndPoint polaczenieUDP in
IPGlobalProperties.GetIPGlobalProperties().
        GetActiveUdpListeners())
            Console.WriteLine("\tRemote address: {0}", polaczenieUDP.Port);
        Console.ReadKey();
    }
}
}
}

```

Listing 1: Source code [own study]

## **List of drawings**

Drawing 1: The beginning of the application's operation [own study].....	3
Drawing 2: Graphical interface [own study].....	4

## **List of listings**

Listing 1: Source code [own study].....	4
---	---