Personal VPN Using UDP and TCP

A PROJECT REPORT

Submitted by

Vishal Haswani (19BCI0181)

Gokul Gopakumar (19BCI0187)

Atla Venkat Suhas (19BCI0205)

Krishna Yanmantram (19BCI0206)

Course Code: CSE 2008

Course Title: Network Security

Under the guidance of

Dr. S. Anto

Associate Professor, SCOPE,

VIT, Vellore.

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

JUNE, 2021

INDEX

ABSTRACT

A Virtual Private Network (VPN) creates an encrypted tunnel between you and an overseas server operated by a VPN service. All our internet traffic is routed through this tunnel, therefore making our data secure from prying eyes along the way. Because your traffic is exiting the VPN server, your true *IP address is hidden, masking your identity and location***.** VPNs also help in browsing websites that may be not available in our current location. VPN can be understood as a method, which by means of tunneling, encryption, authorization, access management, and many other technologies and services through the Internet to transfer data. TCP/UDP profile determines the *type and settings of the network protocol that a subscribing virtual service can use.* It sets a number of parameters, such as whether the virtual service is a TCP proxy versus a pass-through. Data packets sent over TCP/IP are *not private*, which means they can be seen or intercepted. For this reason, it is vital to *avoid using public Wi-Fi networks* for sending private data and to ensure information is encrypted.

MOTIVATION:

Normal VPNs are simply not safe enough, to maintain the hardware and expertise needed for large networks and secure users, VPN services have expensive bills to pay. As a VPN customer, you either pay for a premium VPN service with your Rupes or you pay for free services with your data. We want to build a VPN that is self-hosted and can be used without any limitations.

LITERATURE SURVEY:

The necessities of computerized correspondences for the majority of associations have become very Much during a years ago as a result of various reasons. For example, globalization of Economy expands the interest of media transmission among branch workplaces, and Between organizations arrangements force that asset are shared. Subsequently, diminishing the expense of media transmission frameworks is basic. Generally, organizations have utilized rented lines with that reason. The most Agent model is Frame Relay administration which depends on the transfer of data outlines between middle of the road exchanging workplaces. The assistance, that utilizations lasting virtual circuits (PVCs) through phone network switches, Presents a few downsides:

It becomes costly on the grounds that associations stay open for all time.
The engineering makes huge inactivity periods in view of the poor connectivity between moderate switches.

Full availability requires the augmentation of PVCs and, thus, of halfway Network switches; however, the expense of trying not to defeat issues in this manner is high.

The quantity of organizations that offer Frame-Relay administrations is little looked at to the quantity of Internet Service Providers (ISPs), so intensity is More restricted.

Then again, open organizations offer a more productive arrangement than rented Lines. Accordingly, for instance, Virtual Private Networks (VPNs) utilize generally low- Cost, broadly accessible admittance to public organizations to interface far off locales together securely. Organization structures characterized by VPNs are intrinsically more versatile and Adaptable than old style WANs, and they permit associations to add and eliminate Branch workplaces into their frameworks in a simple manner.

Notwithstanding, and as shown later, the investigation of the distinctive TCP/IP stack layers Uncovers that the various arrangements that empower building up a VPN basically Zero in on security viewpoints. Their primary points are to seclude an appropriated network from untouchables and to secure the protection and honesty of delicate data

Navigating the non-confided in open organizations, as the Internet. These methodologies fall flat t be finished. The principal disadvantage in imagining the security issue as the exceptional objective is that VPN clients experience the ill effects of limitations in getting to the Internet. That is,

- They can’t uninhibitedly utilize customary administrations, for example, electronic mail trade with non-VPN clients, and can’t uninhibitedly get to Web and FTP workers outside to the association

- . All things considered, inside a similar application, it is a troublesome assignment to empower Nonexclusive Internet admittance to VPN clients and, simultaneously, to give a solid and enough security mode

OVERVIEW OF THE WORK

OBJECTIVE OF THE PROJECT:

A VPN client uses special TCP/IP or UDP-based protocols, called tunneling protocols, to make a virtual call to a virtual port on a VPN server. In a typical VPN deployment, a client initiates a virtual point-to-point connection to a remote access server over the Internet. The remote access server answers the call, authenticates the caller, and transfers data between the VPN client and the organization's private network

This paper will present a method to create a user-friendly VPN network using C# using these special TCP and UDP based protocols. In this paper we present an alternative method, set at the TCP/IP transport layer that, whereas maintaining strong security measures, permits the open use of traditional network services running over the Transmission Control Protocol and User Datagram Protocol.

Since the implementation is located at the transport layer; thus, there is no need to modify any software previously installed, like FTP, Telnet, HTTP, electronic mail or other network applications.

SOFTWARE REQUIREMENTS:

C#: It is widely used for developing desktop applications, web applications and web services. It is used in creating applications of Microsoft at a large scale. This is a very easy to use PL and is fun to work with.

Putty: It is a free and open-source terminal emulator, serial console and network file transfer application. It supports several network protocols, including SCP, SSH, Telnet, rlogin, and raw socket connection. It has been used for connecting server to OpenVPN, connect to root of the server to change configurations, RSA key generation for SSL.

FileZilla: FileZilla is a powerful and free software for transferring files over the Internet. It has been used to config files of server with OpenVPN.

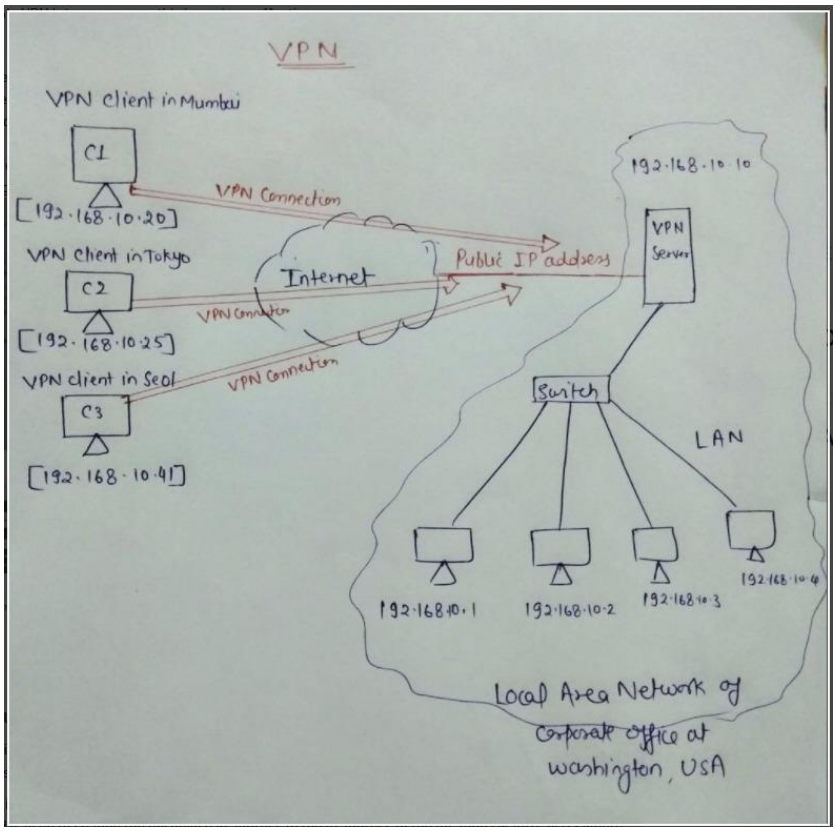
Digital Ocean: It is a cloud hosting provider that offers cloud computing services to business entities so that they can scale themselves by deploying Digital Ocean applications that run

parallel across multiple cloud servers without compromising on performance. It has been used to creating the droplet for VPN server.

OpenVPN: It is an open-source connection protocol used to facilitate a secure tunnel between two points in a network.

SYSTEM DESIGN

INTENDED USE OF THE VPN:



IMPLEMENTATION

PUTTY STEPS (to configure server to TCP and UDP):

```
login as: root
root@138.68.143.226's password:
Welcome to Ubuntu 21.04 (GNU/Linux 5.11.0-17-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri May 28 02:37:25 UTC 2021

  System load:  0.0               Users logged in:       0
  Usage of /:   6.2% of 24.06GB   IPv4 address for eth0: 138.68.143.226
  Memory usage: 19%               IPv4 address for eth0: 10.16.0.5
  Swap usage:   0%                IPv4 address for eth1: 10.106.0.2
  Processes:    94

12 updates can be applied immediately.
12 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@ubuntu-s-1vcpu-1gb-lon1-01:~# curl -O https://raw.githubusercontent.com/ang
ristan/openvpn-install/master/openvpn-install.sh
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 40671  100 40671    0     0   315k      0 --:--:-- --:--:-- --:--:--  315k
root@ubuntu-s-1vcpu-1gb-lon1-01:~# chmod +x openvpn-install.sh
root@ubuntu-s-1vcpu-1gb-lon1-01:~# ./openvpn-install.sh
Welcome to the OpenVPN installer!
The git repository is available at: https://github.com/angristan/openvpn-install

I need to ask you a few questions before starting the setup.
You can leave the default options and just press enter if you are ok with them.
```

I need to know the IPv4 address of the network interface you want OpenVPN listen
ing to.
Unless your server is behind NAT, it should be your public IPv4 address.
IP address: 138.68.143.226

Checking for IPv6 connectivity...

Your host does not appear to have IPv6 connectivity.

Do you want to enable IPv6 support (NAT)? [y/n]: n

What port do you want OpenVPN to listen to?
  1) Default: 1194
  2) Custom
  3) Random [49152-65535]
Port choice [1-3]: 1

What protocol do you want OpenVPN to use?
UDP is faster. Unless it is not available, you shouldn't use TCP.
  1) UDP
  2) TCP
Protocol [1-2]: 1

What DNS resolvers do you want to use with the VPN?
  1) Current system resolvers (from /etc/resolv.conf)
  2) Self-hosted DNS Resolver (Unbound)
  3) Cloudflare (Anycast: worldwide)
  4) Quad9 (Anycast: worldwide)
  5) Quad9 uncensored (Anycast: worldwide)
  6) FDN (France)
  7) DNS.WATCH (Germany)
  8) OpenDNS (Anycast: worldwide)
  9) Google (Anycast: worldwide)
  10) Yandex Basic (Russia)
  11) AdGuard DNS (Anycast: worldwide)
  12) NextDNS (Anycast: worldwide)
  13) Custom
DNS [1-12]: 3

Do you want to use compression? It is not recommended since the VORACLE attack m
ake use of it.
Enable compression? [y/n]: n

Do you want to customize encryption settings?
Unless you know what you're doing, you should stick with the default parameters
provided by the script.

Note that whatever you choose, all the choices presented in the script are safe.
(Unlike OpenVPN's defaults)
See https://github.com/angristan/openvpn-install#security-and-encryption to lear
n more.

Customize encryption settings? [y/n]: n

Okay, that was all I needed. We are ready to setup your OpenVPN server now.
You will be able to generate a client at the end of the installation.
Press any key to continue...
Hit:1 http://mirrors.digitalocean.com/ubuntu hirsute InRelease
Get:2 http://mirrors.digitalocean.com/ubuntu hirsute-updates InRelease [109 kB]
Get:3 http://security.ubuntu.com/ubuntu hirsute-security InRelease [101 kB]
Hit:4 http://mirrors.digitalocean.com/ubuntu hirsute-backports InRelease
Get:5 http://mirrors.digitalocean.com/ubuntu hirsute-updates/main amd64 Packages
[170 kB]
Get:6 http://mirrors.digitalocean.com/ubuntu hirsute-updates/main amd64 c-n-f Me
tadata [3516 B]
Get:7 http://mirrors.digitalocean.com/ubuntu hirsute-updates/universe amd64 Pack
ages [215 kB]
Get:8 http://mirrors.digitalocean.com/ubuntu hirsute-updates/universe amd64 c-n-
f Metadata [4432 B]
Fetched 601 kB in 1s (1061 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20210119build1).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.20-1ubuntu3).
gnupg set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 12 not upgraded.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20210119build1).
curl is already the newest version (7.74.0-1ubuntu2).
curl set to manually installed.
iptables is already the newest version (1.8.7-1ubuntu2).
iptables set to manually installed.
openssl is already the newest version (1.1.1j-1ubuntu3).
openssl set to manually installed.
wget is already the newest version (1.21-1ubuntu3).
wget set to manually installed.
The following additional packages will be installed:
  libpkcs11-helper1
Suggested packages:

```
  resolvconf openvpn-systemd-resolved easy-rsa
The following NEW packages will be installed:
  libpkcs11-helper1 openvpn
0 upgraded, 2 newly installed, 0 to remove and 12 not upgraded.
Need to get 623 kB of archives.
After this operation, 1795 kB of additional disk space will be used.
Get:1 http://mirrors.digitalocean.com/ubuntu hirsute/main amd64 libpkcs11-helper
1 amd64 1.27-1 [44.4 kB]
Get:2 http://mirrors.digitalocean.com/ubuntu hirsute-updates/main amd64 openvpn
amd64 2.5.1-1ubuntu1.1 [579 kB]
Fetched 623 kB in 0s (2584 kB/s)
Preconfiguring packages ...
Selecting previously unselected package libpkcs11-helper1:amd64.
(Reading database ... 66333 files and directories currently installed.)
Preparing to unpack .../libpkcs11-helper1_1.27-1_amd64.deb ...
Unpacking libpkcs11-helper1:amd64 (1.27-1) ...
Selecting previously unselected package openvpn.
Preparing to unpack .../openvpn_2.5.1-1ubuntu1.1_amd64.deb ...
Unpacking openvpn (2.5.1-1ubuntu1.1) ...
Setting up libpkcs11-helper1:amd64 (1.27-1) ...
Setting up openvpn (2.5.1-1ubuntu1.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn.service → /l
ib/systemd/system/openvpn.service.
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for libc-bin (2.33-0ubuntu5) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.
--2021-05-28 02:40:23--  https://github.com/OpenVPN/easy-rsa/releases/download/v
3.0.7/EasyRSA-3.0.7.tgz
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-releases.githubusercontent.com/4519663/0fa24e00-72ba-11
ea-9afe-6e5829eec4a4?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAIWNJY                AX4CSVEH53A%2F20210528%2Fus-
east-1%2Fs3%2Faws4_request&X-Amz-Date=20210528T02402
4Z&X-Amz-Expires=300&X-Amz-
Signature=eddbab6d514452053770f3c8feb36b353452a534d63
fe80bc4f8ca3166651302&X-Amz-
```

SignedHeaders=host&actor_id=0&key_id=0&repo_id=45196
63&response-content-disposition=attachment%3B%20filename%3DEasyRSA-3.0.7.tgz&res
ponse-content-type=application%2Foctet-stream [following]
--2021-05-28 02:40:24-- https://github-releases.githubusercontent.com/4519663/0
fa24e00-72ba-11ea-9afe-6e5829eec4a4?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-
Amz-Crede
ntial=AKIAIWNJYAX4CSVEH53A%2F20210528%2Fus-east-
1%2Fs3%2Faws4_request&X-Amz-Date
=20210528T024024Z&X-Amz-Expires=300&X-Amz-
Signature=eddbab6d514452053770f3c8feb3
6b353452a534d63fe80bc4f8ca3166651302&X-Amz-
SignedHeaders=host&actor_id=0&key_id=
0&repo_id=4519663&response-content-
disposition=attachment%3B%20filename%3DEasyRS A-
3.0.7.tgz&response-content-type=application%2Foctet-stream
Resolving github-releases.githubusercontent.com (github-releases.githubuserconte
nt.com)... 185.199.108.154, 185.199.110.154, 185.199.109.154, ...
Connecting to github-releases.githubusercontent.com (github-releases.githubuserc
ontent.com)|185.199.108.154|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 48215 (47K) [application/octet-stream]
Saving to: ‘/root/easy-rsa.tgz’

/root/easy-rsa.tgz 100%[==================>] 47.08K --.-KB/s in 0.001s

2021-05-28 02:40:24 (79.8 MB/s) - ‘/root/easy-rsa.tgz’ saved [48215/48215]

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars

init-pki complete; you may now create a CA or requests.
Your newly created PKI dir is: /etc/openvpn/easy-rsa/pki

Using SSL: openssl OpenSSL 1.1.1j 16 Feb 2021
read EC key
writing EC key

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars
Using SSL: openssl OpenSSL 1.1.1j 16 Feb 2021
Generating an EC private key
writing new private key to '/etc/openvpn/easy-rsa/pki/easy-rsa-14194.6xKnhY/tmp.
i98EAu'

Using configuration from /etc/openvpn/easy-rsa/pki/easy-rsa-14194.6xKnhY/tmp.oTr
roX

Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName :ASN.1 12:'server_V1gZlfvMVIodwsb4'
Certificate is to be certified until Aug 31 02:40:24 2023 GMT (825 days)

Write out database with 1 new entries
Data Base Updated

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars
Using SSL: openssl OpenSSL 1.1.1j 16 Feb 2021
Using configuration from /etc/openvpn/easy-rsa/pki/easy-rsa-14271.54nLix/tmp.dWC
HIX

An updated CRL has been created.
CRL file: /etc/openvpn/easy-rsa/pki/crl.pem

2021-05-28 02:40:24 WARNING: Using --genkey --secret filename is DEPRECATED. Us
e --genkey secret filename instead.
* Applying /etc/sysctl.d/10-console-messages.conf ...
kernel.printk = 4 4 1 7
* Applying /etc/sysctl.d/10-ipv6-privacy.conf ...
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
* Applying /etc/sysctl.d/10-kernel-hardening.conf ...
kernel.kptr_restrict = 1
* Applying /etc/sysctl.d/10-magic-sysrq.conf ...
kernel.sysrq = 176
* Applying /etc/sysctl.d/10-network-security.conf ...
net.ipv4.conf.default.rp_filter = 2
net.ipv4.conf.all.rp_filter = 2
* Applying /etc/sysctl.d/10-ptrace.conf ...
kernel.yama.ptrace_scope = 1
* Applying /etc/sysctl.d/10-zeropage.conf ...
vm.mmap_min_addr = 65536
* Applying /usr/lib/sysctl.d/50-default.conf ...
net.ipv4.conf.default.promote_secondaries = 1
sysctl: setting key "net.ipv4.conf.all.promote_secondaries": Invalid argument
net.ipv4.ping_group_range = 0 2147483647
net.core.default_qdisc = fq_codel
fs.protected_regular = 1
fs.protected_fifos = 1
* Applying /usr/lib/sysctl.d/50-pid-max.conf ...
kernel.pid_max = 4194304
* Applying /etc/sysctl.d/99-cloudimg-ipv6.conf ...

```
net.ipv6.conf.all.use_tempaddr = 0
net.ipv6.conf.default.use_tempaddr = 0
* Applying /etc/sysctl.d/99-openvpn.conf ...
net.ipv4.ip_forward = 1
* Applying /etc/sysctl.d/99-sysctl.conf ...
* Applying /usr/lib/sysctl.d/protect-links.conf ...
fs.protected_fifos = 1
fs.protected_hardlinks = 1
fs.protected_regular = 2
fs.protected_symlinks = 1
* Applying /etc/sysctl.conf ...
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn@server.servi
ce → /etc/systemd/system/openvpn@.service.
Created symlink /etc/systemd/system/multi-user.target.wants/iptables-openvpn.ser
vice → /etc/systemd/system/iptables-openvpn.service.

Tell me a name for the client.
The name must consist of alphanumeric character. It may also include an undersco
re or a dash.
Client name: vitprj

Do you want to protect the configuration file with a password?
(e.g. encrypt the private key with a password)
   1) Add a passwordless client
   2) Use a password for the client
Select an option [1-2]: 1

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars
Using SSL: openssl OpenSSL 1.1.1j  16 Feb 2021
Generating an EC private key
writing new private key to '/etc/openvpn/easy-rsa/pki/easy-rsa-14551.ndTPbW/tmp.
6W6dkp'
-----
Using configuration from /etc/openvpn/easy-rsa/pki/easy-rsa-14551.ndTPbW/tmp.ugF
Abs
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'vitprj'
Certificate is to be certified until Aug 31 02:42:25 2023 GMT (825 days)

Write out database with 1 new entries
Data Base Updated

Client vitprj added.

The configuration file has been written to /root/vitprj.ovpn.
```

Download the .ovpn file and import it in your OpenVPN client.
root@ubuntu-s-1vcpu-1gb-lon1-01:~# lsof -i:1194
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
openvpn 14367 nobody 7u IPv4 280665 0t0 UDP *:openvpn
root@ubuntu-s-1vcpu-1gb-lon1-01:~# iptables -A input -i eth0 -m state --state NEW -p tcp --dport 1194 -j ACCEPT
iptables: No chain/target/match by that name.
root@ubuntu-s-1vcpu-1gb-lon1-01:~# iptables -A input -i eth0 -m state --state NEW -p tcp --dport 1194 -j ACCEPT
iptables: No chain/target/match by that name.
root@ubuntu-s-1vcpu-1gb-lon1-01:~# iptables -A INPUT -i eth0 -m state --state NEW -p tcp --dport 1194 -j ACCEPT
root@ubuntu-s-1vcpu-1gb-lon1-01:~# iptables -t nat -A POSTROUTING -s 10.9.0.0/24 -o eth0 -j MASQUERADE
root@ubuntu-s-1vcpu-1gb-lon1-01:~# iptables -A OUTPUT -o tun+ -j ACCEPT
root@ubuntu-s-1vcpu-1gb-lon1-01:~# service openvpn@server restart
root@ubuntu-s-1vcpu-1gb-lon1-01:~# service openvpn@server2 restart
root@ubuntu-s-1vcpu-1gb-lon1-01:~# service openvpn@server restart
root@ubuntu-s-1vcpu-1gb-lon1-01:~# service openvpn@server2 restart
root@ubuntu-s-1vcpu-1gb-lon1-01:~# lsof -i:1194
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
openvpn 18619 nobody 7u IPv4 340852 0t0 UDP *:openvpn
openvpn 18633 nobody 7u IPv4 340929 0t0 TCP *:openvpn (LISTEN)
root@ubuntu-s-1vcpu-1gb-lon1-01:~#

SOURCE CODE FOR APPLICATION:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Diagnostics;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Net;
using System.Net.NetworkInformation;

namespace VPNMAIN
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void connectstatus_CheckedChanged(object sender, EventArgs e)
```

```
        {
            if (connectstatus.Checked)
            {
                if (USchk.Checked || Italychk.Checked || Cambodiachk.Checked)
                {
                    Process process = new Process();
                    ProcessStartInfo startInfo = new ProcessStartInfo();

                    startInfo.FileName = @"C:\Program Files\OpenVPN\bin\openvpn.exe";
                    startInfo.Arguments = "--config vitprjtcp.ovpn";//configuration
file
                    startInfo.Verb = "runas"; //Run as Administrator
                    process.StartInfo = startInfo;
                    process.Start();

                    MessageBox.Show("Connected TCP ");
                }
                else if(UKchk.Checked || Germanychk.Checked)
                {
                    Process process = new Process();
                    ProcessStartInfo startInfo = new ProcessStartInfo();

                    startInfo.FileName = @"C:\Program Files\OpenVPN\bin\openvpn.exe";
//file location of openvpn
                    startInfo.Arguments = "--config vitprj.ovpn";//configuration file
                    startInfo.Verb = "runas"; //Run as Administrator
                    process.StartInfo = startInfo;
                    process.Start();

                    MessageBox.Show("Connected UDP ");
                }
                else
                {
                    MessageBox.Show("Please select one of the Countries Listed ");
                    connectstatus.Checked = false;
                }

            }
            else
            {
                if ((USchk.Checked || Italychk.Checked || Cambodiachk.Checked ||
UKchk.Checked || Germanychk.Checked))
                {
                    disconnection();
                    MessageBox.Show("DISCONNECTED ");
                }
            }
        }

        private void disconnection()
        {
            Process.Start(new ProcessStartInfo
            {
                FileName = "taskkill",
                Arguments = $"/f /IM openvpn.exe",
                CreateNoWindow = true,
                UseShellExecute = false
            }).WaitForExit();
```

```
        }

        public static double Speed(string url)
        {
            WebClient wc = new WebClient();
            DateTime dt = DateTime.Now;
            byte[] data = wc.DownloadData("https://www.google.com/");
            DateTime dt2 = DateTime.Now;

            return (data.Length*8) / (dt2-dt).TotalSeconds;
        }

        private void tmr_Tick(object sender, EventArgs e)
        {
            float show = (float) Speed("https://www.google.com/") / 1048576;

            speedshow.Text = show +" MB";
        }

        private void Logout_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

LOGING IN AS ROOT AFTER ADDING SSH KEY:

ssh-rsa

AAAAB3NzaC1yc2EAAAADAQABAAABAQCl8rHbHa3R1nrzQSTBIBlXKZIGdTzj5o4

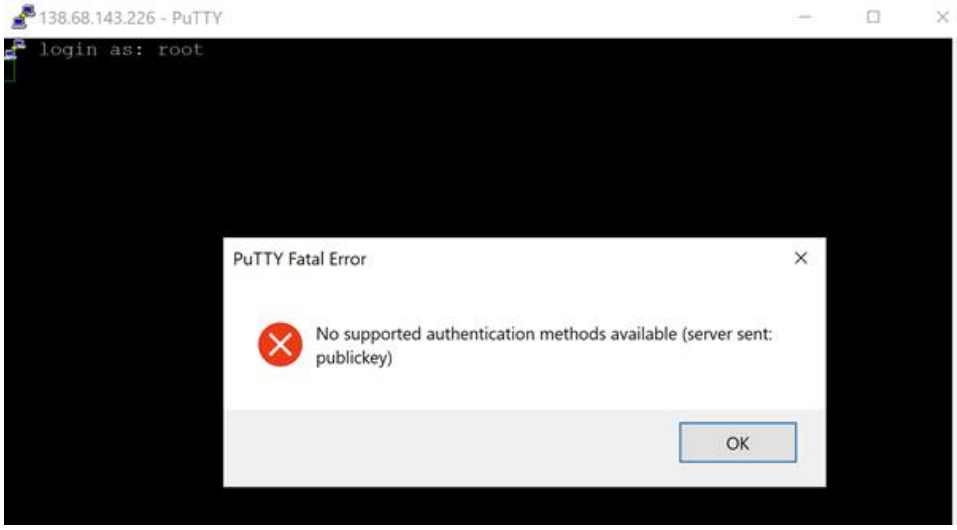
NIETb29wCCiKcb9szzMktSrDH28Zw/talc9Rwg3BaBc4lQj/n+uKLK0Kb6ydDD+iMkY/rsz

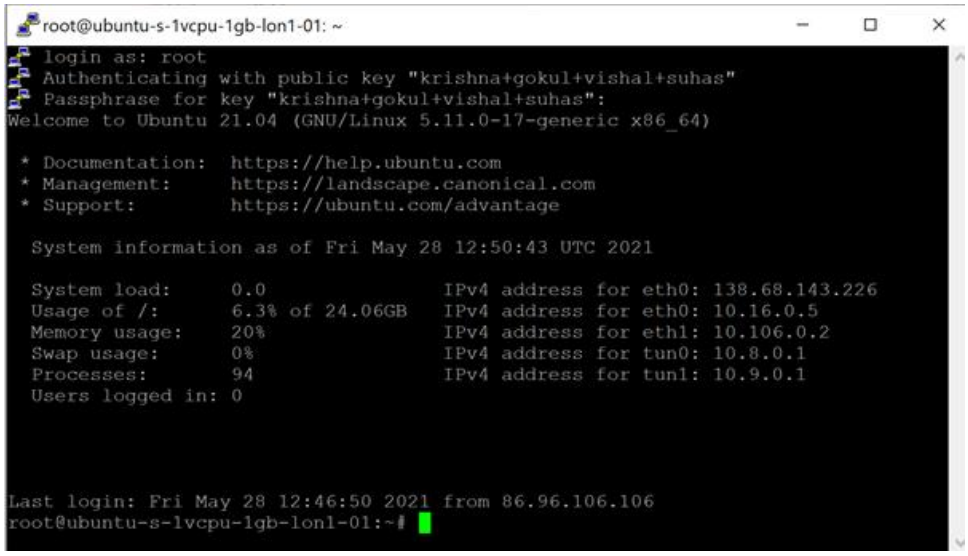
71E90hfrhNJIFjdq5EwPZG1CxxpsfieoEtSYrUSLsvGV5TYss44FrtLR7l23VkbmlNanr3N5

MVE/8I+HN5Jf5aV4kcYUpeRrXMGrXH9TlSY3a4X6z2SwnHpJVaVGaQqGYfm293a7oZ

Mb+0B4E0YsUKsnJqedLZUaMqPSDs+ejQygNTbZjztecBs4WkicX9ayRK1b7k/sh7R7pzW

XuNPRWn3Nztr8jUeJTJ+ZxRV+Xuknat krishna+gokul+vishal+suhas

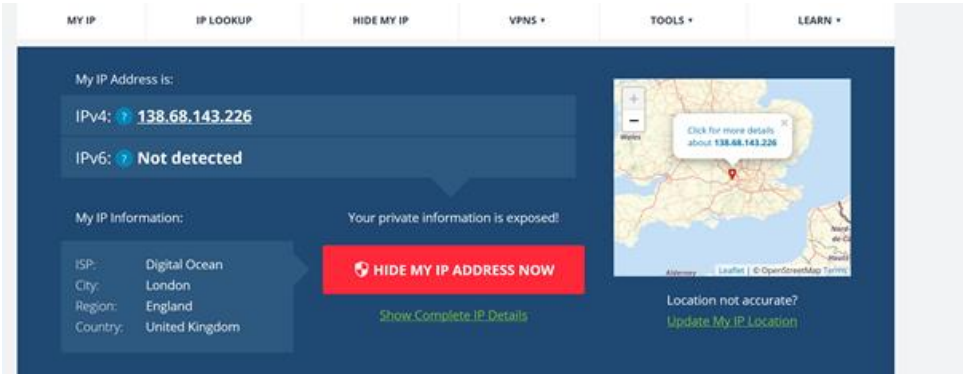

138.68.143.226 - PuTTY
login as: root
PuTTY Fatal Error
No supported authentication methods available (server sent: publickey)
OK


root@ubuntu-s-1vcpu-1gb-lon1-01: ~
login as: root
Authenticating with public key "krishna+gokul+vishal+suhas"
Passphrase for key "krishna+gokul+vishal+suhas":
Welcome to Ubuntu 21.04 (GNU/Linux 5.11.0-17-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information as of Fri May 28 12:50:43 UTC 2021
System load: 0.0
Usage of /: 6.3% of 24.06GB
Memory usage: 20%
Swap usage: 0%
Processes: 94
Users logged in: 0
IPv4 address for eth0: 138.68.143.226
IPv4 address for eth0: 10.16.0.5
IPv4 address for eth1: 10.106.0.2
IPv4 address for tun0: 10.8.0.1
IPv4 address for tun1: 10.9.0.1
Last login: Fri May 28 12:46:50 2021 from 86.96.106.106
root@ubuntu-s-1vcpu-1gb-lon1-01:~#

OUTPUT AND PERFORMACE ANALYSIS

EXECUTION SCREENSHOTS:

This is the screenshot of another app showing our IP address after the VPN is active:



CONCLUSION

In this paper we have presented a new solution for the implementation of VPNs at the transport layer that, while maintaining strong security features, allows the open use of traditional Internet services that run over TCP and UDP layers.

Moreover, it does not require the modification of the software of traditionally insecure applications such as FTP, HTTP, Telnet, electronic mail, etc. We have also successfully secured the VPS Server using SSH keys to make it protected from attackers.

As a result, the C# VPN project is a simple and cheap solution for those organizations that want to install a VPN.

REFERENCES

[1] A. Skendzic and B. Kovacic, “Open source system OpenVPN in a function of virtual private network,” In IOP Conference Series: Materials Science and Engineering, vol. 200, no. 1, p. 12065, 2017

[2] K. Karuna Jyothi, Dr. B. Indira Reddy, "Study on Virtual Private Network (VPN), VPN’s Protocols And Security", (http://ijsrcseit.com/paper/CSEIT1835225.pdf), 2018

[3] Muhammad Iqbal, Imam Riadi, "Analysis of Security Virtual Private Network (VPN) Using OpenVPN", ResearchGate, International Journal of Cyber-Security and Digital Forensics 8(1):58-65, May, 2019

[4] Nico Surantha, Rino, “Secure Portable Virtual Private Network with Rabbit Stream Cipher Algorithm”, Procedia Computer Science, Volume 135, 2018, Pages 259-266, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2018.08.173. (https://www.sciencedirect.com/science/article/pii/S1877050918314625)

[5] Zhang S., Li A., Zhu H., Sun Q., Wang M., Zhang Y. (2018) Research on the Protocols of VPN. In: Xhafa F., Patnaik S., Zomaya A. (eds) Advances in Intelligent Systems and Interactive Applications. IISA 2017. Advances in Intelligent Systems and Computing, vol 686. Springer, Cham. https://doi.org/10.1007/978-3-319-69096-4_77

[6] Lopez, Javier & Montenegro, José & Roman, Rodrigo & Dávila, Jorge. (2002). Design of a VPN software solution integrating TCP and UDP services. 2437. 325-338. 10.1007/3-540-45831-X_23.