sciendo

Providing the Ability of Working Remotely on Local Company Server via VPN

doi:10.2478/mape-2020-0017
Date of submission to the Editor: 03/2020
Date of acceptance by the Editor: 06/2020
MAPE 2020, volume 3, issue 1, pp. 195-205

Adam Muc
ORCID ID: 0000-0002-9495-087X

Tomasz Muchowski
ORCID ID: 0000-0002-0200-7041

Lech Murawski
ORCID ID: 0000-0003-0089-5492

Adam Szeleziński
ORCID ID: 0000-0003-2842-0683
Gdynia Maritime University, **Poland**

INTRODUCTION

Remote working is becoming increasingly popular in companies using new technologies. In most companies where employees use computers and use software that depends only on the resources of the computer or local server used, work can be done remotely. The solution to the problem of remote file access is the clouds, i.e. Google Drive, Microsoft OneDrive and Dropbox, while in the case of programming companies, repositories such as Github, Gitlab and Bitbucket using the GIT mechanism, which provides version control, can be used. Local servers that host websites, web applications or databases can be replaced by VPS (Virtual Private Server) (Blokdyk 2020) or cloud computing i.e. Amazon AWS or Microsoft Azure (Shaik and Vallarapu 2018).

There are cases where it is not possible or not cost-effective to transfer services to the cloud and VPS. The cost of cloud resources and the performance of the VPS server, depending on requirements, may exceed the cost-effectiveness threshold of offering remote working possibility to employees. Attention must be paid to both costs and security. VPS and cloud providers suffer from data leaks, and the network administrator does not always have full control over the data stored in the clouds (Jain and Mahajan 2017).

An alternative to using clouds and VPS can be port forwarding on the company's main router and providing local server services to the external network. This option, however, involves exposing the server to attacks from the external network. It is necessary to increase the effectiveness of security to an appropriate level so as not to expose the server to effective penetration. Not every enterprise has qualified network administrators and not every enterprise is able to spend enough money to hire them.

Contact address: a.szelezinski@wm.umg.edu.pl

A solution to the problem of the need to increase the level of security, as well as the need to incur financial costs associated with the purchase of cloud resources, may be VPN (Virtual Private Server). The goal of VPN is to create a secure, encrypted tunnel through which a device in a remote location can be connected to the local network and access services provided on the local network as if it were directly connected to it (Feilner 2006). Enabling a VPN service is not expensive, in most cases it is enough to equip an enterprise with an appropriate programmable router, but if the number of remote connections is large, a dedicated server may be required.

TEST BENCH

Test router

The authors used Mikrotik router with RouterOS 6.47 software flashed. The configured router (RB951G-2HnD model) has been reset to factory settings. The initial configuration has been done – IP address of LAN interface has been set, DHCP service has been configured and the required IP address pools have been created. WAN port of the router has been connected to the main router and NAT translation has been enabled. Table 1 contains information about the router's configuration, and Figure 1 shows the created IP address pools.

Table 1 Router's configuration

WAN interface IP address	192.168.100.247
LAN address	192.168.200.0/24
Router's LAN interface address	192.168.200.1
DHCP address pool	192.168.200.100-192.168.200.254
VPN address pool	192.168.200.10-192.168.200.99

Source: authors' own work

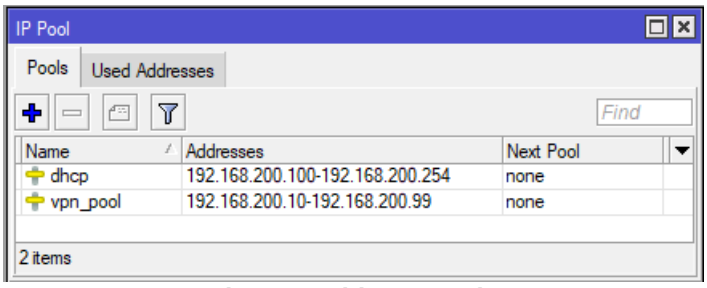

Fig. 1 IP address pools

Source: authors' own work

As it can be seen in the picture above, the local network has been divided into two pools. So the local and remote devices will be in the same subnet. On the main router there has been created static DNS mapping of bussiness.domain.com domain name to IP address 192.168.200.247.

All LAN interfaces are bridged and bridge's ARP configuration is set to *proxy-arp*. This is necessary for remote devices to be able to connect to local devices. Figure 2 shows the proces of changing bridge ARP to *proxy-arp*.

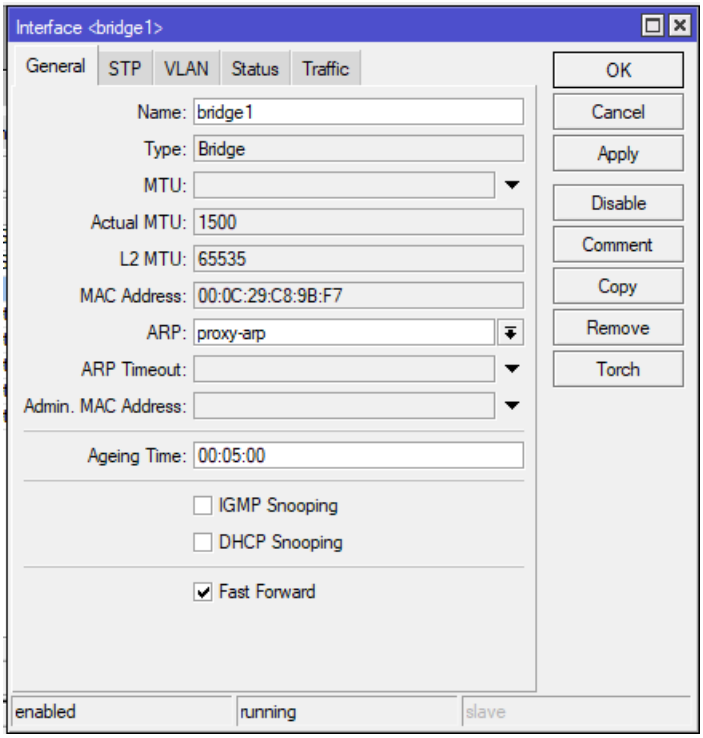

Fig. 2 Changing the bridge ARP to proxy-arp

Source: authors' own work

Test client devices

As client devices, the authors used two computers equipped with different operating systems. The first device has Windows 10 installed and the second one has Debian 10.04 installed, a popular Linux distribution. Both operating systems have been freshly installed for testing purposes and do not have any custom network interface configuration.

ENABLING VPN SERVICE

Service configuration

Among many VPN protocols, the authors chose OpenVPN because of high security of tunnel encryption as well as modernity. To create a proper client-server connection, certificates and a secret (credentials – login and password) are required. First, certificates were created and signed (System -> Certificates). CA certificate, server certificate and client certificates are needed (Feilner 2006). As part of the test configuration, only one client certificate and one secret for the network administrator were created. Creating more client certificates is similar to creating the Admin-Bussiness certificate, and creating more secrets is similar to creating the adminVPN secret. While creating certificates, the working domain name bussiness.domain.com is used, but it can be replaced by the IP address of WAN interface of the router being configured. The default RSA encryption method was not changed. Figure 3 shows the configuration details of the created certificates.

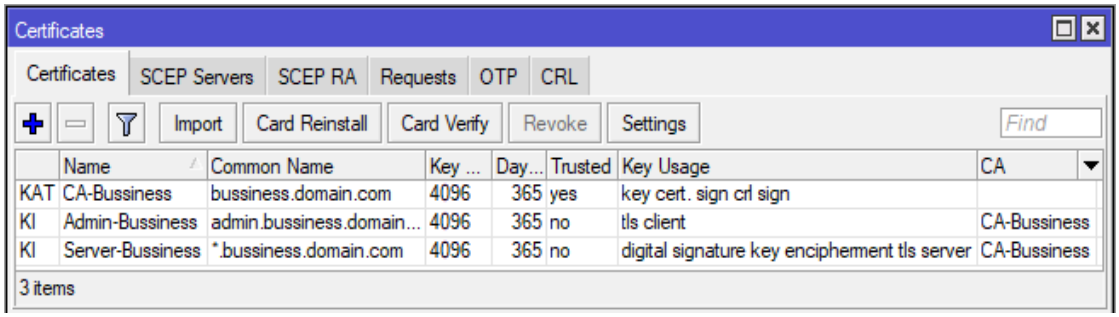

Fig. 3 View of created certificates

Source: authors' own work

Created certificates must be signed for them to be valid. Figure 4 shows the process of signing certificates. The order of signing is important.

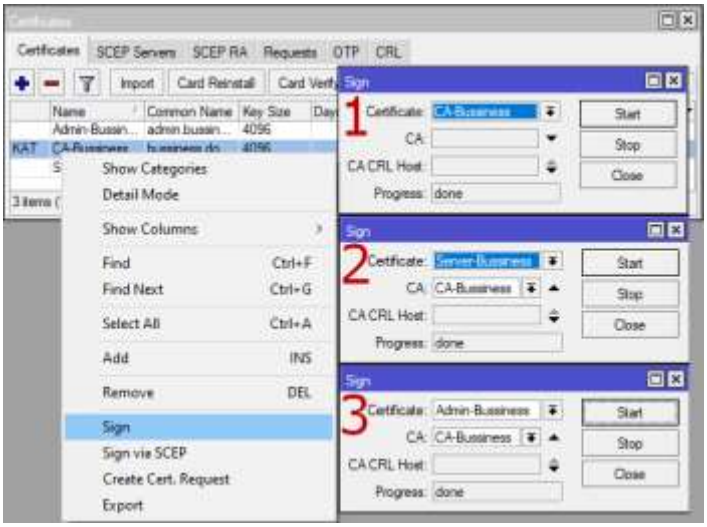

Fig. 4 The process of signing certificates

Source: authors' own work

After signing the certificates, the CA certificate and the client certificate must be exported (Fig. 5).

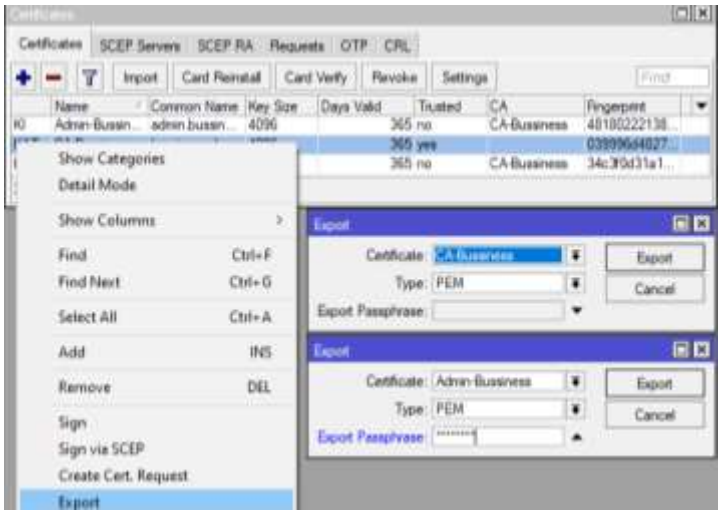

Fig. 5 Process of exporting certificates

Source: authors' own work

When exporting, it is worth providing a password for the client certificate, but it is optional. After exporting, the certificates and the key will be stored in the memory of the device from where they can be downloaded.
Creating a Point-to-Point Protocol (PPP -> Profiles) profile is the next step. Specifying the name of the profile, the local address that the router is to use as a service address, and the pool of addresses from which IP addresses will be assigned to clients is required. It is necessary to also specify the DNS server address (this may be the local default gateway address) and force the use of encryption. Figure 6 shows the process of creating a PPP profile.

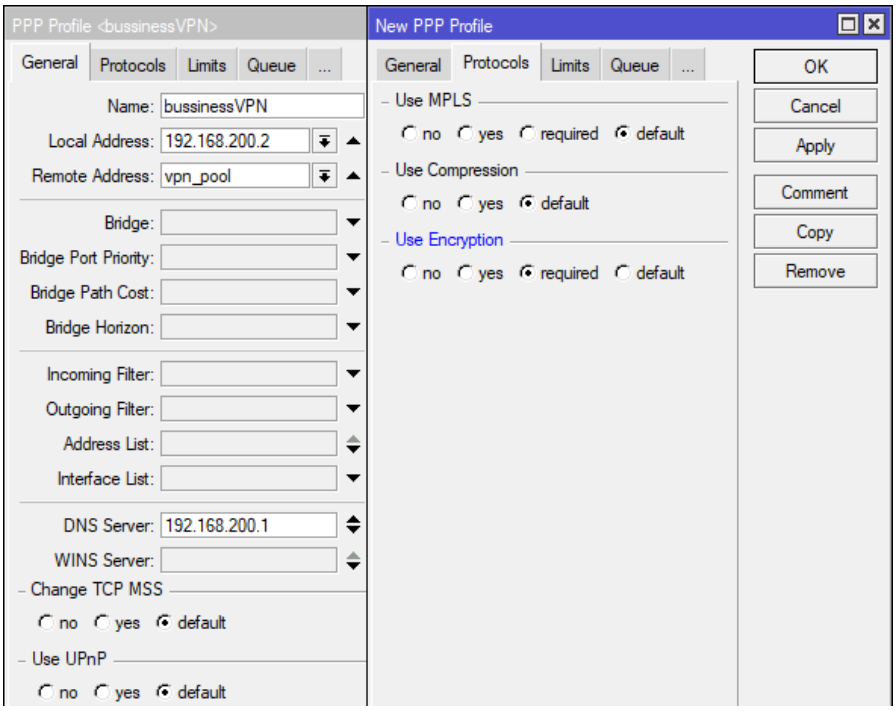

Fig. 6 Creating PPP profile

Source: authors' own work

Once the profile is created, it is needed to create a user secret (PPP -> Secrets). It is necessary to specify the login, password, service (in this case ovpn) and profile (reference to the profile created in the previous step). For the test configuration, only the secret for the network administrator has been created (Fig. 7) for testing purposes, but it is necessary to create unique secrets for each VPN service user.
The next step is to create an OpenVPN interface (PPP -> Interface). It is necessary to specify the port (default 1194), select the profile (reference to the previously created profile) and select the server certificate (reference to the previously created certificate). It is also necessary to specify the encryption algorithm used for authentication (in this case SHA1) and tunneling (in this case AES256).

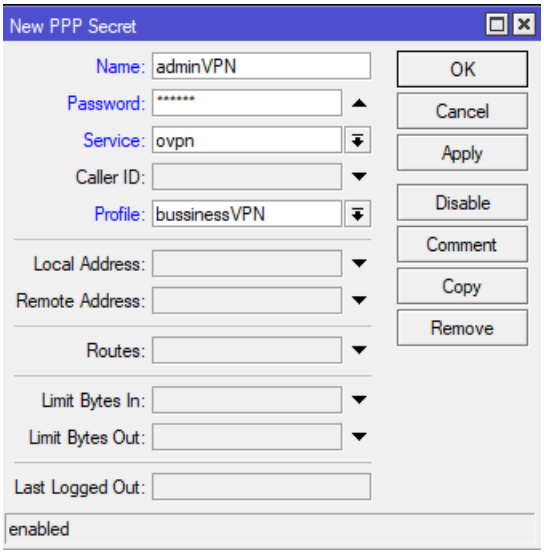

Fig. 7 Creating a secret

Source: authors' own work

The next step is to create an OpenVPN interface (PPP -> Interface). It is necessary to specify the port (default 1194), select the profile (reference to the previously created profile) and select the server certificate (reference to the previously created certificate). It is also necessary to specify the encryption algorithm used for authentication (in this case SHA1) and tunneling (in this case AES256). It is worth to use strong encryption algorithms to ensure greater security of tunneling (Crist and Keijser 2015). Figure 8 shows the process of creating OpenVPN interface.

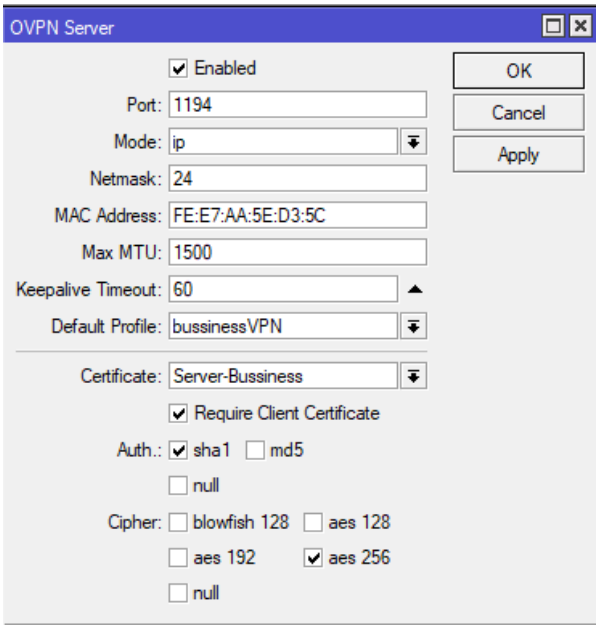

Fig. 8 Creating the OpenVPN interface

Source: authors' own work

After creating and configuring the OpenVPN service, configure the firewall (IP -> Firewall -> Filter Rules) is required (Crist 2017). It is necessary to create a

filter rule for incoming traffic to make the router able to listen on the TCP port used by VPN service (in this case 1194). Figure 9 shows the process of creating a filtering rule.

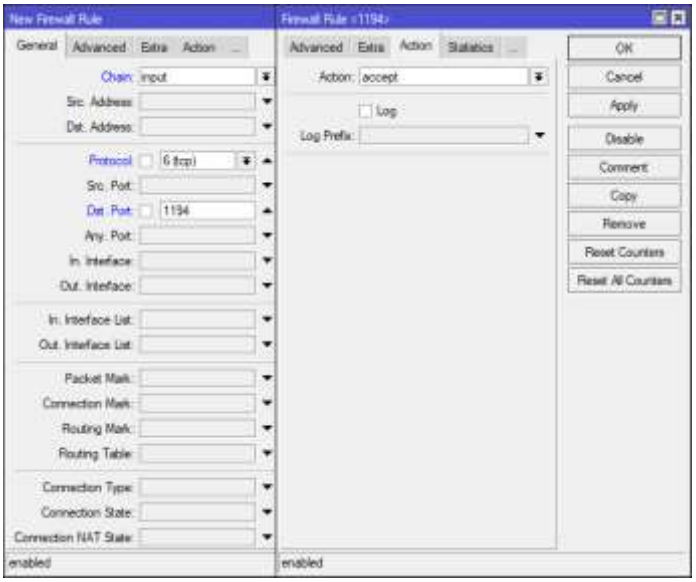

Fig. 9 Creating a firewall filtering rule

Source: authors' own work

After the configuration is completed, the created certificates must be downloaded from the device memory (Files -> Download). The files can be stored in any location on the computer. Figure 10 shows the process of downloading certificates and the key from the device memory.

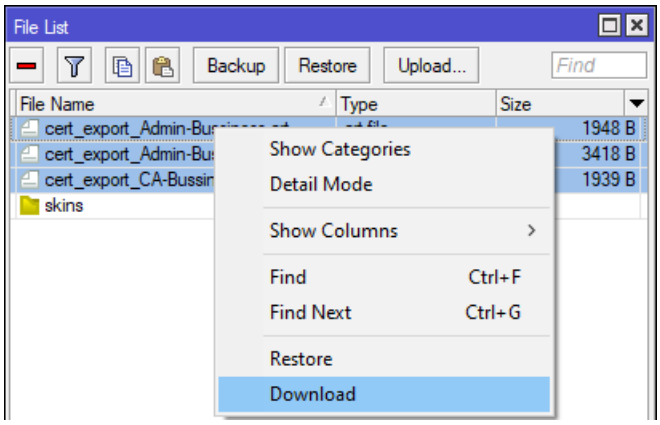

Fig. 10 Downloading the certificates and the key

Source: authors' own work

In order for the client device to connect to the created OpenVPN server, it is necessary to create a configuration file describing the server configuration. This file should have the *.ovpn extension and contain the address, protocol and connection port to the server, mapping of the certificate and key file names (the file names can be changed to simpler and more universal, i.e. ca.crt, client.crt and passwd.key so that creating separate configuration files for other clients is

not necessary) to their respective fields. It is also important to specify encryption algorithms. Figure 11 shows the contents of the file created for the test configuration.

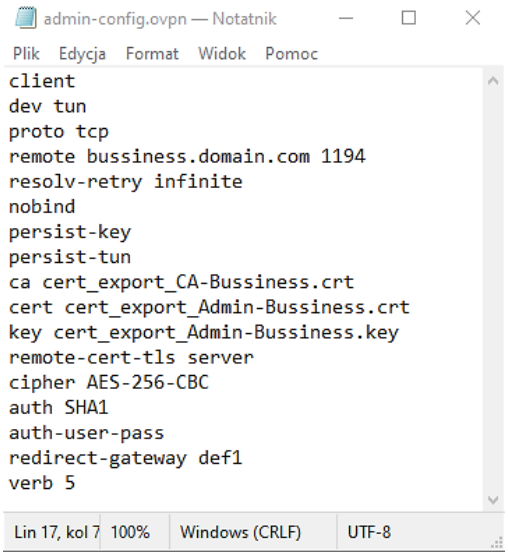

Fig. 11 Creation an OpenVPN configuration file

Source: authors' own work

After creating the configuration file, it should be transferred along with the certificate files and the key to the client device.

Client device configuration

Depending on the operating system installed on the client device, other actions are required. For most Linux distributions it is enough to import a configuration file *.ovpn to the *Network Manager* (wiki.archlinux.org 2020) and the tunnel will be ready to use. However, for Windows 10, this operation is more complicated because this system does not support OpenVPN tunneling by default. It is necessary to install OpenVPN software, which can be downloaded from the manufacturer's website (openvpn.net 2020). After the installation is completed, a folder containing the program's configuration (C:\Program Files\OpenVpn\config\) should be created in the server configuration folder (in this case *bussiness*), where the created configuration file, certificate files and key file should be placed. Figure 12 shows the contents of the folder.

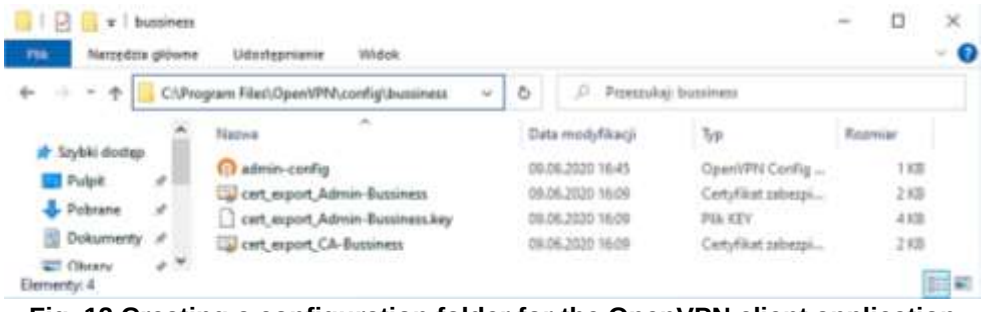

Fig. 12 Creating a configuration folder for the OpenVPN client application

Source: authors' own work

Running the application is the next step. It will be necessary to provide the login and password of the secret and the password of the certificate (if were created). The authorization process is presented in Figures 13 and 14.

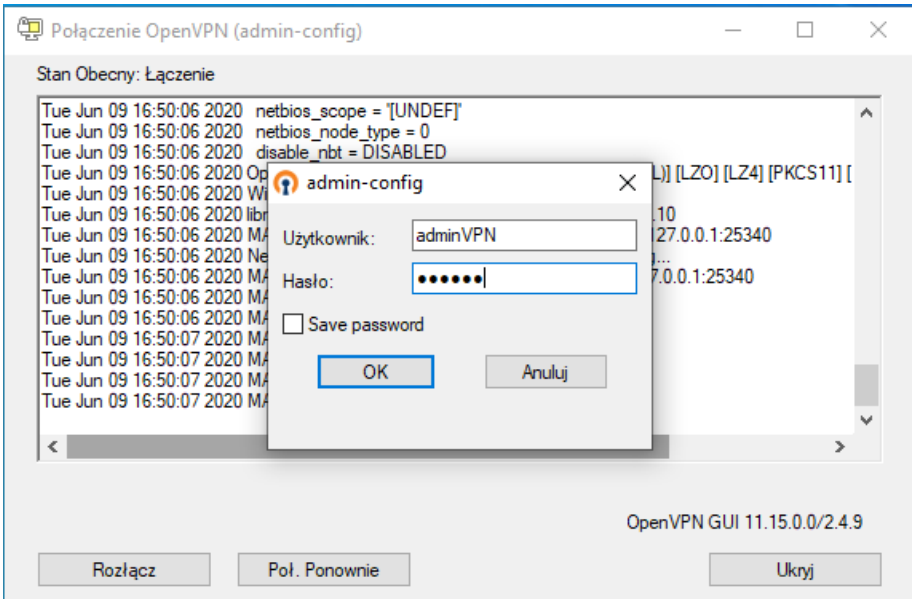

Fig. 13 Providing user credentials

Source: authors' own work

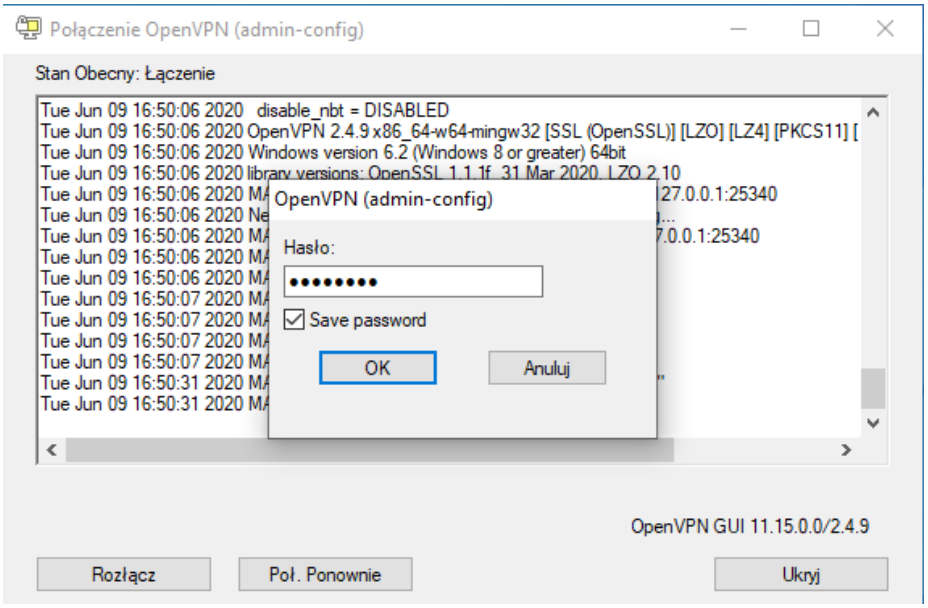

Fig. 14 Providing the certificate password

Source: authors' own work

User authentication will be completed and the tunnel creation process will start. Depending on the bandwidth speed, this process may take several minutes. After completion, the program logs will display the information about the remote IP address assigned to device (Fig. 15).

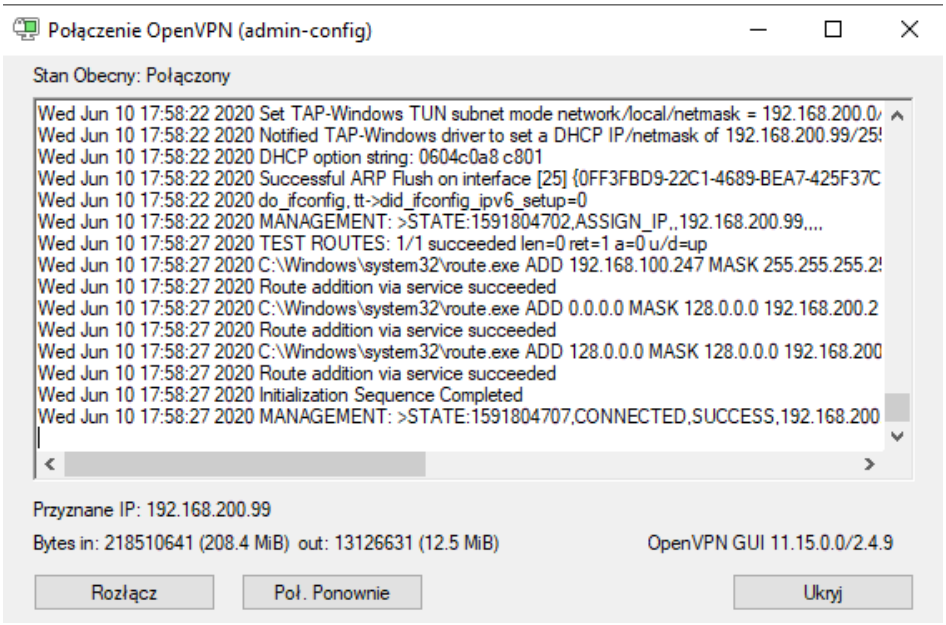

Fig. 15 View of logs

Source: authors' own work

After creating the tunnel, it is worth to check the configuration of the device's network interfaces. The virtual tunneling interface should receive an address from the VPN address pool specified in the router configuration. Information about network interfaces of the client device is shown in Figure 16.

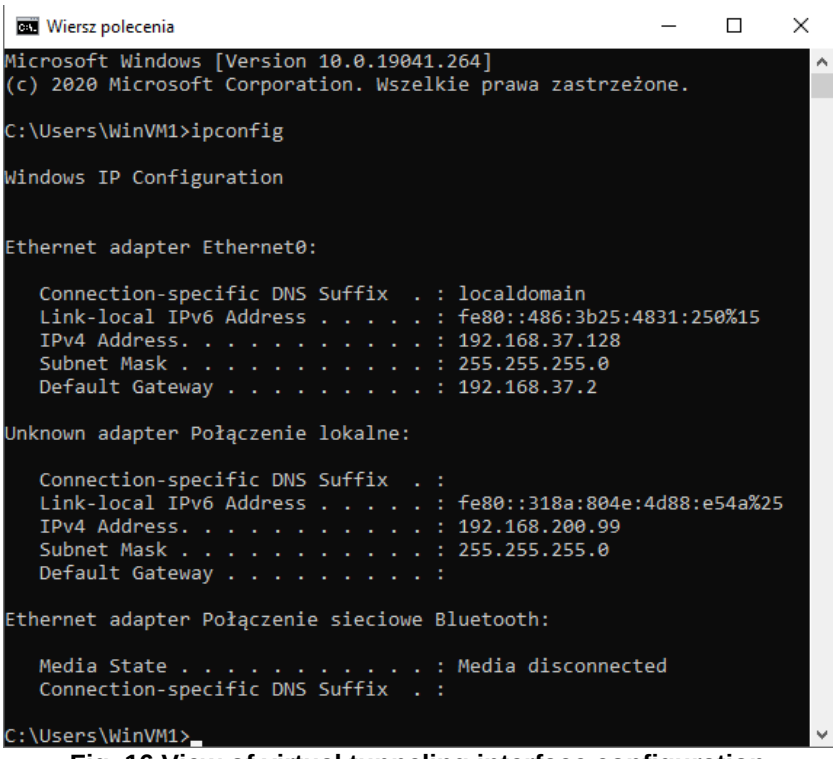

Fig. 16 View of virtual tunneling interface configuration

Source: authors' own work

As it can be seen in the picture above, the virtual network interface received the address 192.168.200.99 from the VPN address pool of the router. This means

that network traffic will be tunneled to the router's local network and then through its gateway. The created tunnel allows to connect to devices in the enterprise's local network as if the device was physically connected to the enterprise's local network.

CONCLUSION

VPN tunneling consists in redirecting the device's network traffic to the VPN server's local network. A device using a VPN tunnel has full access to resources and services provided by other devices on the remote local network. This means that the device can access the services provided by the enterprise's local server as if it were connected to the local network directly via a physical transmission medium. This technology allows employees to work remotely without generating high costs like clouds and VPS servers do. As the network communication in the tunnel is encrypted, using a properly configured server does not adversely affect the security of corporate data and that is not something that can be said about clouds and VPS. VPN also does not require placing company data outside the local server, thus not exposing it to theft by the server service provider. VPN technology has practical application in every enterprise where it is necessary to have access to the local server from a remote location.

REFERENCES

Blokdyk G. (2020) Virtual Private Server A Complete Guide – 2020 Edition. 5STARCooks, pp. 30-45.

Crist E.F. and Keijser J.J. (2015) Mastering OpenVPN. Packt Publishing, pp. 16-23.

Crist E.F. (2017) Troubleshooting OpenVPN. Packt Publishing, pp. 94-123.

Feilner M. and Graf N. (2009) Beginning OpenVPN 2.0.9. Packt Publishing, pp. 10-21

Feilner M. (2006) OpenVPN: Building and Integrating Virtual Private Networks: Learn how to build secure VPNs using this powerful Open Source application. Packt Publishing, pp. 20-25.

Jain A. and Mahajan N. (2017) The Cloud DBA-Oracle: Managing Oracle Database in the Cloud. Apress, pp. 25-28.

Shaik B. and Vallarapu A. (2018) Beginning PostgreSQL on the Cloud: Simplifying Database as a Service on Cloud Platforms. Apress, pp. 5-14.

https://openvpn.net/community-downloads/ [June 1, 2020]

https://wiki.archlinux.org/index.php/Networkmanager-openvpn [June 1, 2020]

Abstract: Increasingly popular remote work requires the use of modern network technologies to provide employees in a remote location with access to the company's IT resources. The answer to the needs of remote access to files and server services can be the use of clouds and VPS. However, this involves high costs and requires entrusting the enterprise's data to the providers of these services. Both for reasons of data security and too high costs, enterprises sometimes cannot use these technologies. The solution to the problem may be the use of encrypted VPN tunneling, which allows the device to be connected at a remote location to the company's local network and use its resources as if it was connected to the local network with physical transmission medium.

Keywords: remote work, server services, VPN, OpenVPN, server costs