Annales UMCS Informatica AI XIV, 3 (2014) 53–64
DOI: 10.2478/umcsinfo-2014-0014

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

# Performance Analysis of VPN Remote Access Tunnels

Paweł Dymora[1*], Mirosław Mazurek[1†], Tomasz Pilecki[1‡]

[1] *Rzeszów University of Technology, Faculty of Electrical and Computer Engineering, Department of Distributed Systems, ul. Wincentego Pola 2, 35-959 Rzeszów, Poland*

**Abstract** – The purpose of the study is to analyze the efficiency of communication with the server using the methods of secure remote access, as well as checking and comparing the quality of services provided by the server depending on the method of secure remote connection. The article focuses on VPN technology implemented in the latest Windows Server 2012 R2 operating system.

## 1 Introduction

Nowadays we can observecontinuous computer networks development. This creates need for developing a secure file transfer technology in those networks. Today the Internet handles not only simple data exchange like files, music or movies. The transferred information is very important for companies, even if classified, which is often essential for company further operation. Those data are highly exposed to dangers like data capture and content change. VPN technology is a response to those problems because of the use of bulit-in secure data protection mechanism, which is commonly used for transferring data through the public insecure networks, which are known as vulnerable to data intercept and read attacks [**1**].

## 2 Characteristics of the analyzed protocols

The article compares four most popular communication protocols which implement the VPN technology. The most commonly used is PPTP (Point-to-Point Tunneling

*pawel.dymora@prz.edu.pl
†mirekmaz@prz.edu.pl
‡123037@stud.prz.edu.pl

Protocol). PPTP is a new technology for creating Virtual Private Networks (VPN), developed jointly by Microsoft Corporation, U.S. Robotics, and several companies. PPTP is used to ensure that messages transmitted from one VPN node to another are secure. The second analyzed protocol is L2TP (Layer 2 Tunneling Protocol). It is a tunneling protocol used to support VPNs or as a part of the delivery of services by ISPs. It does not provide any encryption or confidentiality by itself. It relies on the encryption protocols that passe data within the tunnel to provide privacy. The third analyzed protocol is SSTP (Secure Socket Tunneling Protocol). SSTP provides a mechanism to transport PPTP or L2TP traffic through the SSL 3.0 channel with key-negotiation, encryption and traffic integrity checking. The last analyzed protocol is the Internet Key Exchange protocol (IKE or IKEv2). IKE is the protocol used to set up a security association (SA) in the IPsec protocol suite. IKE is built upon the Oakley protocol and ISAKMP. IKE uses X.509 certificates for authentication or DNS and a Diffie–Hellman key exchange distribution. A security policy for every peer which will connect must be manually maintained [**2, 3, 4, 5**].

## 3 The analyzed system model

In order to perform the study, two computers were used. One running Windows Server 2012 R2 operating system, and a client computer with the Windows 8.1 Professional operating system. Communication infrastructure was based on the TP Link TL-WDR4300 dual-band router. The topology depicted in Fig. 1 shows one computer which acts as an Active Directory controller and a remote access server.

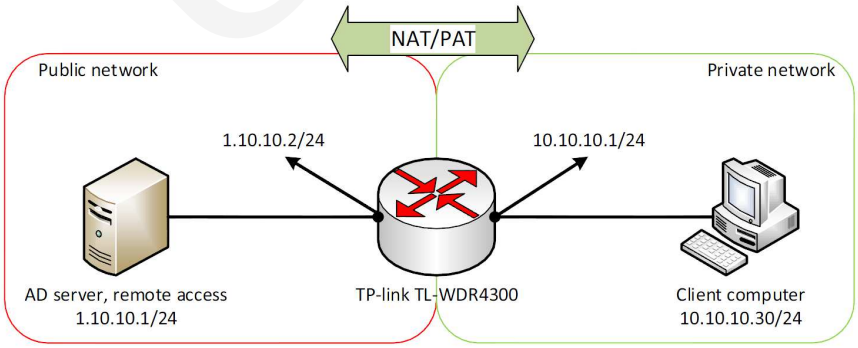


Fig. 1. The analyzed system model.

The Active Directory service is essential in this topology to maintain database, for the user's authorization. Remote access – so called Remote Access role is needed to listen to clients requests, who want to set up the VPN tunnel between their computer and server.

This scenario shows a typically used topology for this kind of connections. Usually is a client a computer located in private network connected to the Internet. This computer needs a mechanism which changes its private address to the public, one because private addresses are not processed in the Internet. This action requires a device which can swap the private addness to the public one (this is executed by NAT translation and PAT translation in the case when there is a need to use only one external IP address). The remaining element of this topology is the TP Link TL-WDR4300 router, which is capable of performing NAT and PAT translations. It supports also data transfer over VPN technology like L2TP, IPsec and PPTP. It has 6 ports, where 5 ports are used for the internal network and 1 for the external network [**6, 7, 8**].

## 4 Data link throughput tests

The bandwidth tests have been performed by using the Jperf programme. In those tests the JAVA GUI-based version of Jperf application has been used. Jperf is an open source program mainly used for computer networks tuning. It has been written to check if this program can work with various types of networks. It can be launched in Windows, UNIX and even in Mac OS operating systems. That forces to review and reconfigure its parameters according to the used type of operating system. In the analyzed case it was essential to use 64 kilobit TCP Window size and the same size of Buffer Length which are the typical values used in the Windows operating system. Jperf calculates the bandwidth based on the size of transmitted data. The single measurement cycle is based on two end hosts where one of them is the server and the other acts as the client machine. The server listens for incoming connections and the client sends data. It has to be decided which port will be used for data measurement. The client must use the same port number as the server. In every single case, single measurement lasts 5 minutes (300 s), where every single measurement value is taken every second. Five series of measurements were performed for each type of VPN tunnel and one additional to check whether the link operates correctly. The whole measurement was conducted for both Ethernet and FastEthernet links.

As it was shown in Fig. 2, the result bandwidth value is above 90 Mbit/s, but it is not equal to 100 Mbit/s. This result is considered to be correct. In reality the average link bandwidth was 94.12 Mbit/s. That shows the situation when link operates normally. The similar test was performed for Ethernet link, where average bandwidth was equal to 9.49 Mbit/s. The VPN measurements were performed for a theoretical bandwidth of 100 Mbit/s as shown in Table 1. The average values of all measurements were calculated on the basis of 300 values taken in all tests.

The results of every single measurement of VPN tunnel were surprisingly high. The results have been expected to have lower values than the theoretical physical link bandwidth. The fact that the VPN results were better than the expected ones was not a matter of chance. As painted out earlier the Jperf application works by measuring amounts of transferred data. VPN tunnels support data compression which is the main

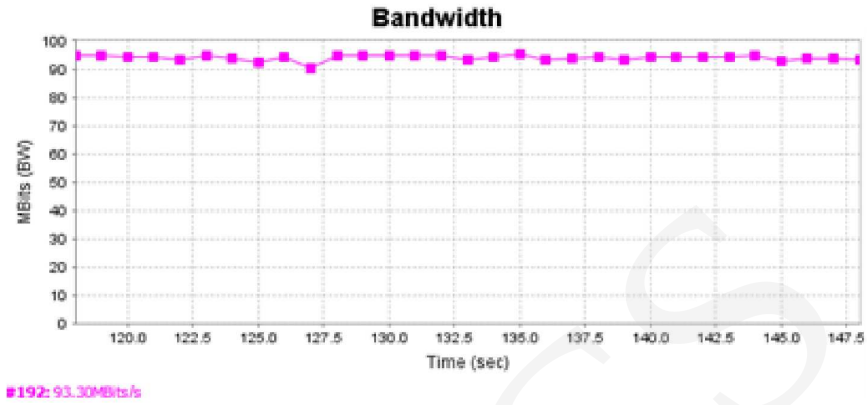


Fig. 2. 100 Mbit/s link test measurements.

Table 1. The statement of theoretical 100 Mbit/s bandwidth.

| Average throughput of VPN tunnels – 100 Mbit/s link | | | | |
|---|---|---|---|---|
| **Test number** | **PPTP (Mbit/s)** | **L2TP/IPsec (Mbit/s)** | **SSTP (Mbit/s)** | **IKEv2 (Mbit/s)** |
| 1 | 119.16 | 91.98 | 111.75 | 88.48 |
| 2 | 121.75 | 91.88 | 110.70 | 88.73 |
| 3 | 119.68 | 87.04 | 133.02 | 88.95 |
| 4 | 120.93 | 90.40 | 111.33 | 86.70 |
| 5 | 121.61 | 89.12 | 111.30 | 89.44 |
| **Average throughput of physical link: 94.12 Mbit/s** | | | | |

cause of higher final measurement value. If one single datagram is compressed, it will have a lower size. The langer quantity of compressed datagrams makes the end result much better than expected, which further gives better final results. It can be assumed that the end results are wrong because of strange value greater than 100 Mbit/s. To prove that this theory is correct, additional measurements were performed.

Fig. 3 shows the measurement of link bandwidth without a VPN tunnel in comparison to the bandwidth reported by an operating system. We can admit that this measurement using Jperf is correct. Please have a look at the CPU usage value which is equal to 23%. The same figure shows that the presence of compression of data was confirmed at the time of this measurement.

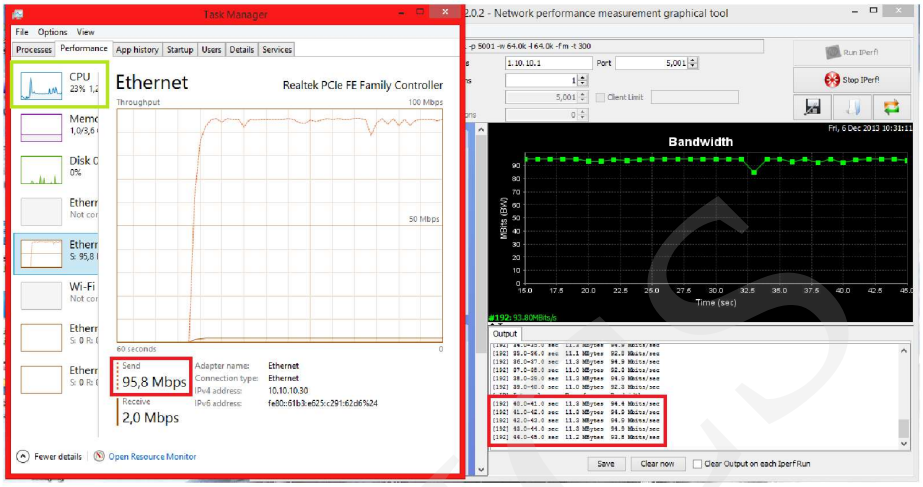


Fig. 3. 100Mbit/s link measurement without VPN tunnel.

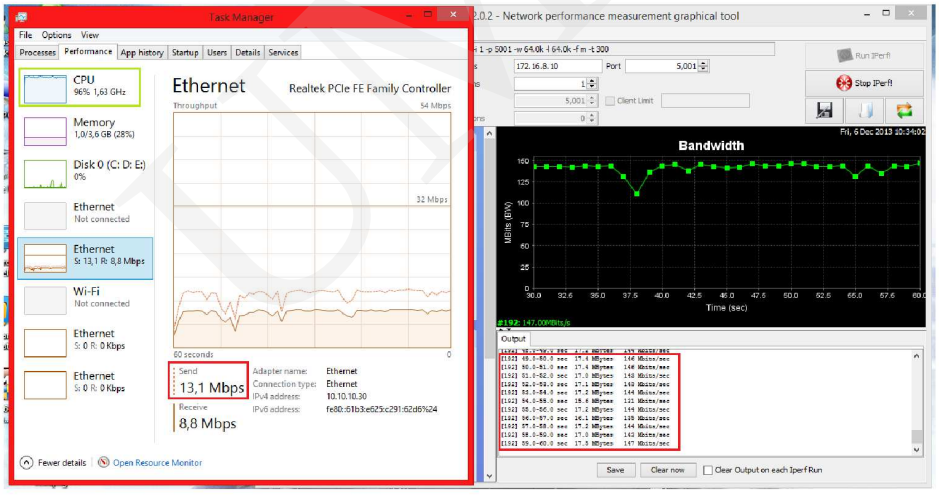


Fig. 4. 100Mbit/s link measurement with VPN tunnel.

Let us have a look at the reported values of bandwidth presented in Fig. 4. There are almost 147 Mbit/s in Jper. These values can not appear during standard measurements of FastEthernet network – it is not physically possible to obtain these values. Let us compare them with the reported bandwidth of the operating system which is equal to 13.1 Mbit/s. This value is very low but it is correct in relation to this type of link. It is essential to look also at the CPU usage value marked with a green frame – it is 90%. This shows that data compression needs additional CPU cycles to compress transmitted data. In addition, let us compare the network traffic flow graphs with

each other – these graphs are identical. The above confirms the theory that data compression is present in the VPN tunnel. Based on that we can conclude that Jperf has not measured the physical link bandwidth but the exact VPN tunnel bandwidth. That proves that the measurements results are correct.

A similar analysis was made for VPN tunnels created in Ethernet networks. In that case the measurements were conducted in the same way as the previous series and the test results – even lower, were satisfactory as shown in Table 2.

Table 2. The statement of theoretical 10 Mbit/s bandwidth.

| Average throughput of VPN tunnels – 10Mbit/s link | | | | |
|---|---|---|---|---|
| **Test number** | **PPTP (Mbit/s)** | **L2TP/IPsec (Mbit/s)** | **SSTP (Mbit/s)** | **IKEv2 (Mbit/s)** |
| 1 | 84.22 | 60.56 | 66.24 | 8.99 |
| 2 | 82.53 | 61.07 | 66.79 | 9.00 |
| 3 | 83.17 | 61.79 | 64.08 | 8.99 |
| 4 | 86.55 | 61.79 | 72.08 | 8.99 |
| 5 | 83.33 | 61.46 | 80.53 | 9.00 |
| **Average throughput of physical link: 9.49Mbit/s** | | | | |

In that case the results are eight times as high as the expected ones. That is also the cause of compression mechanisms inside VPN tunnels. Comparing all these results it can be assumed that the worst results were obtained using the IKEv2 VPN tunnel. This mechanism provides the worst type of packet compression which can be confirmed by the CPU usage graph analysis.

To confirm that network interface still works properly (with the maximum available bandwidth – 10 Mbit/s), the check has been made by using "Task Manager" tool built-in Windows 8.1. Fig. 5 shows the real bandwidth reported by client host which was sending data needed for further calculations.

As it can be seen the result presented in Fig. 5 is marked with a red frame, represents the value of traffic which was sent, is correct according to the standard of the Ethernet link bandwidth. During that measurement the maximal value equal to 10 Mbit/s of sent traffic appeared in very short time periods – even shorter than 1 second. This prevented it from being registered by Task Manager. It is essential to look at the value registered by Jperf – 9.44 Mbit/s. This shows that this rate of bandwidth is correct for that type of link. The CPU usage is also pretty low – 12% along with all operating system processes working in the background.

Similarly, Fig. 6 shows the results obtained by measuring 100 Mbit/s link. In that case it was much harder to observe the compression phenomenon because of frequent

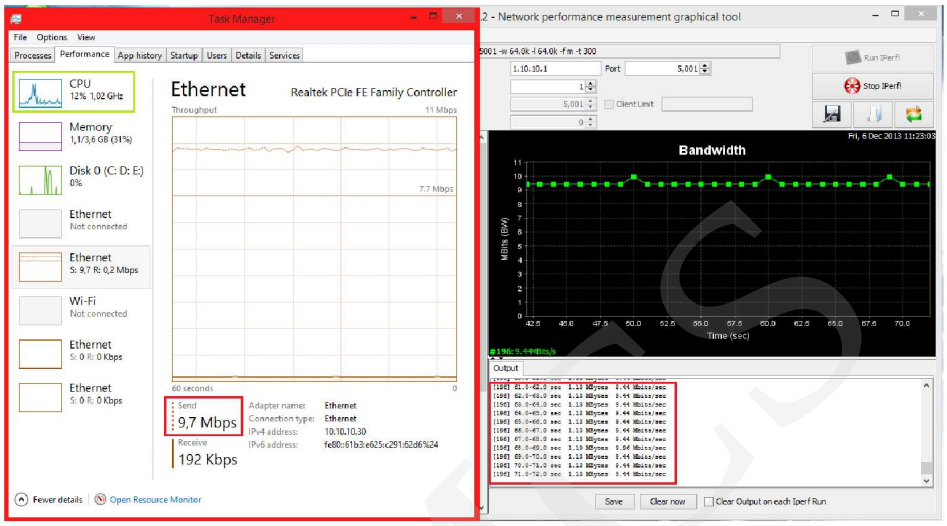


Fig. 5. 10Mbit/s link measurement without VPN tunnel.

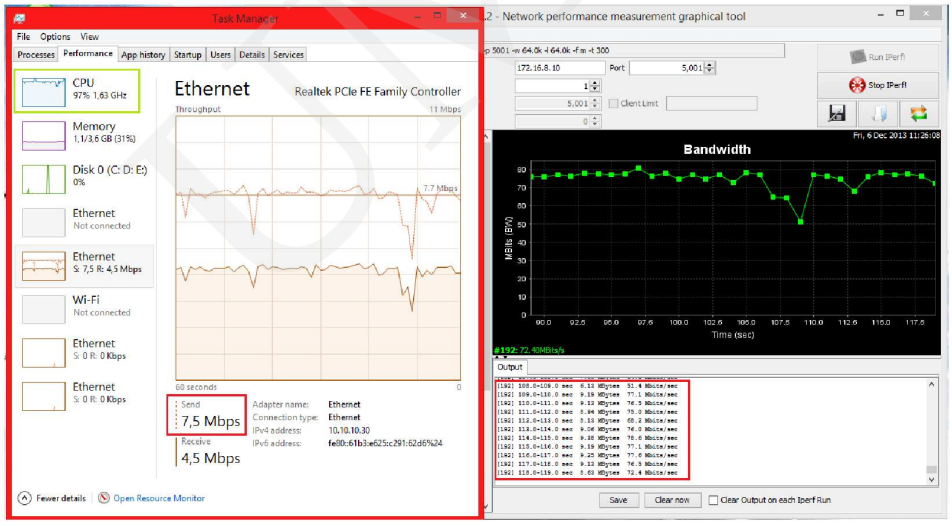


Fig. 6. 10Mbit/s link measurement with VPN tunnel.

fluctuations of reading the physical link bandwidth. However, it can be concluded by looking at these two situations that VPN also compresses data here. The results from Jperf are 7 or 8 times as high as the physical bandwidth of the network adapter. These results are wrong from the physical network interface point of view. However, Jperf does not measure physical network bandwidth here, but the logical VPN tunnel. It can be concluded from the similarity of graphs that those results are wrong but

the whole network operates normally. Note that the high CPU load is higher than when exchanging data without established VPN tunnel. All these observed results also provide information about the presence of compression when communicating through a VPN tunnel.

The most accurate information was presented by a SpeedTest application. This program can eliminate all the delays which can occur writing and reading data from the hard disk. This makes the VPN tunnel and network bandwidth measurement much more accurate. SpeedTest does not report the results during the test performance. These results can be obtained from successful measurement. Fig. 7 shows the obtained results for a network without VPN tunnel. All those results are lower from those seen before because the measurements are the average transfer results for each measured VPN tunnel [**9**].

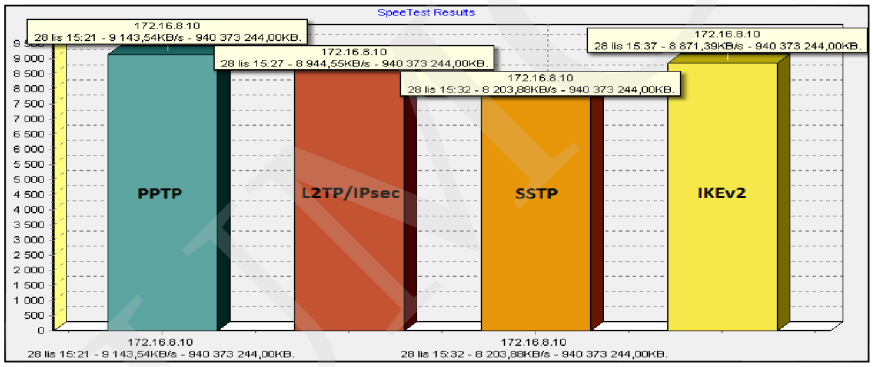


Fig. 7. FastEthernet measurement results with the VPN tunnel (SpeedTest).

When combined PPTP registered transfer ranged from 10 to 10.3 MB/s, the 10.3 MB/s transfer was the maximum possible for this connection. The value of 10 MB/s is equal to about 80 Mb/s. For L2TP protocol the average transfer rate corresponds to the maximum speed of 9.7 MB/s whose value is about 77.6 Mbit/s. For the file transfer in the SSTP tunnel the values ranged from 5.6 MB/s to 5.7 MB/s, which gives a value of around 44.8 and 45.6 Mbit/s. This is the worst result obtained for this type of tests. In the case of IKEv2 tunnel the transfer was 9.4 MB/s, which temporarily changed to 9.3 MB/s. This value is about 75.2 Mb/s and 74.4 Mb/s. Similarly, the same measurements were conducted for all VPN types of tunnels for 10 Mbit/s Ethernet network (Fig. 8).

In the PPTP tunnel observed data transfer rate ranged from 1.1 - 1.7 MB/s. These results provide a transfer of about 8 - 13.6 Mbit/s. It was observed that the compression caused a temporary increase in the transfer. In the L2TP tunnel it was difficult to read the average value of transfer. The results ranged from 1.1 to 1.6 MB/s which gives 8.8 - 12.8 Mbit/s. For the SSTP tunnel we have not recorded any good results. There were values from 1 to 1.7 MB/s, which is 8 - 13.6 Mbit/s. IKEv2 offered a transfer of

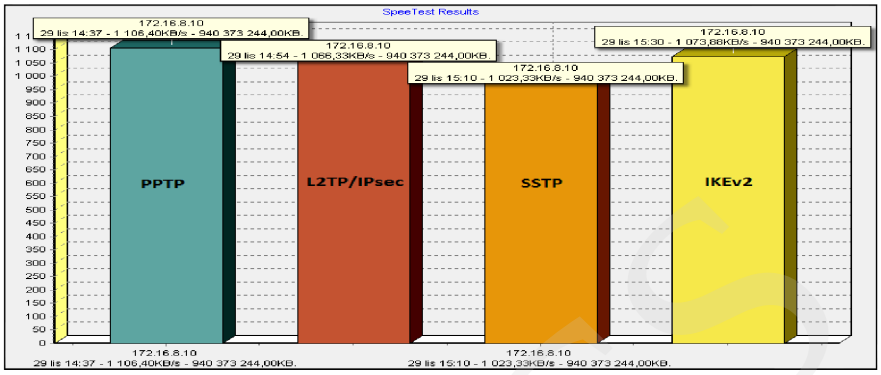


Fig. 8. Ethernet measurement results with the VPN tunnel (SpeedTest).

1.1 MB/s (8.8 Mbit/s) - it was the maximum speed for the tunnel network based on the Ethernet standard.

## 5 The compression influence on the system load

The processes associated with VPN technology, such as creating connection, packets encryption and compression require many computer computations. This involves an increase of the cpu usage coefficient. In the professional network infrastructures, this fact is taken into account [**10, 11**]. Devices that implement the VPN tunnel must have a very high computing power. For the system parameters measurements a free program called System Explorer was used. This program allows to preview the list of running processes, system event log , as well as statistics of the network interfaces use, or CPU, or RAM [**12**]. The study used a server with an Intel Core 2 Duo T5750 2.0 GHz processor, and the client machine with an AMD E -450 1.65 GHz processor (both processors are 2 core) . Figures 9-12 show the CPU load on the client and server computers registered in the described four protocols variants for Fast Ethernet VPN tunnels. The test includes generating traffic with the Jperf program during 5 minutes in the direction from the client to the server.

When comparing the results obtained in the System Explorer, we can see that during the data transfer through the VPN tunnel endpoints, both CPUs are much loaded. This is particularly true in the case of client computer. Most aggravating for the client computer was to transfer data via PPTP and SSTP protocols. Going back to the bandwidth measurement, and comparing it with each other graphs it can be stated that with the increased consumption of the CPU processing power it was possible to obtain VPN tunnel bandwidth with a higher throughput. This, in turn, results in compression level of information sent through the VPN tunnel. Comparing the load charts of the client and the server machines, specificity of the processor load-balancing of different CPU implementations can be noticed. The server CPU (Intel) relatively


Fig. 9. The CPU usage for the VPN tunnel with PPTP/IPsec in FastEthernet: a) server; b) client.


Fig. 10. The CPU usage for the VPN tunnel with L2TP/IPsec in FastEthernet: a) server; b) client.

evenly distributes the load on its cores while the graphs showing the load level of cores in the client CPU (AMD) present significant imbalance.

## 6 Conclusions

Various test results were obtained in many different network environments. All those surprisingly large value results were obtained due to the MPPC compression algorithm. It provides 8:1 level of datagram compression. The high level of CPU utilization on both client and server machines were observed along with this phenomenon. The CPU load fluctuated at 85-96% of all available computer processing power. Generally,

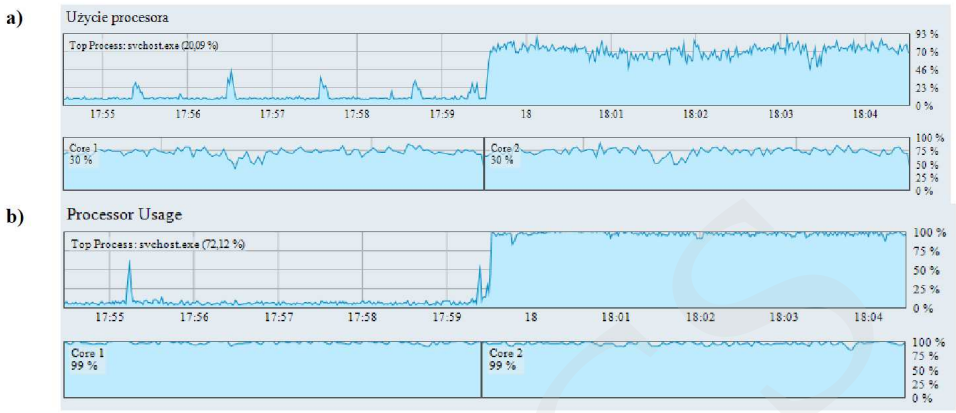


Fig. 11. The CPU usage for the VPN tunnel with SSTP/IPsec in FastEthernet: a) server; b) client.


Fig. 12. The CPU usage for the VPN tunnel with IKEv2 in FastEthernet: a) server; b) client.

it is very hard to determine which type of VPN would be the best choice for the use in professional network environments. The VPN tunnel types, which have the best available bandwidth rate, are known for being much more vulnerable to network attacks because of having outdated types of data protection mechanisms. In one type of VPN implementation the data protection mechanisms are its great advantage but different type of VPN tunnel has better data transfer rate, which is also essential. If we had to choose the best solution based on the security level of transmitted data, the IKEv2 VPN tunnel would be the best although even not all devices support it. In a situation when an administrator would have a task to configure secure remote access connection to a company's server, the good choice would be the type of protocol commonly supported by most available devices in network architecture. In another

case it would be better to choose L2TP/IPsec. This type of VPN tunnel is a kind of compromise between good security provided by SSTP and quite good data transfer rate.

## References

[1] Snader J. C., VPNs Illustrated: Tunnels, VPNs, and IPSec, publisher Addison Wesley Professional (2005).
[2] Lewis M., Comparing, Designing, and Deploying VPNs, publisher Cisco Press (2006).
[3] Kivinen T., Swander B., Huttunen A., Volpe V., Negotiation of NAT-Traversal in the IKE, RFC 3947 (2005).
[4] Kaufman C., Hoffman P., Nir Y., Eronen P., Internet Key Exchange Protocol Version 2 (IKEv2), RFC 5996 (2010).
[5] http://www.windowsecurity.com/articles-tutorials/firewalls_and_VPN/Secure-Socket-Tunneling-Protocol.html
[6] Mazurek M., Dymora P., Network anomaly detection based on the statistical self-similarity factor for HTTP protocol, Przegląd elektrotechniczny, ISSN 0033-2097, R. 90 NR 1/2014 (2014): 127.
[7] Stanek W. R., Windows Server 2012 Inside Out, publisher Microsoft Press (2013).
[8] Ben-Ari E., Natarajan B., Windows Server 2012 Unified Remote Access Planning and Deployment, publisher Packet Publishing (2012).
[9] http://www.raccoonworks.com/index.php?option=com_content&view=article&id=50&Itemid=58 – SpeedTest
[10] Dymora P., Mazurek M., Delay analysis in wireless sensor network protocols, PAK 2013 nr 10 (2013): 1054.
[11] Dymora P., Mazurek M., Gawron R., Cluster computing performance in the context of nonextensive statistics, JIMER International Journal of Modern Engineering Research (IJMER), 3(6) (2013): 3872; ISSN: 2249-6645.
[12] Strzałka B., Mazurek M., Strzałka D., Queue Performance in Presence of Long-Range Dependencies – an Empirical Study, International Journal of Information Science 2(4) (2012): 47.