International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 10 Issue 6 June - 2026, Pages: 30-37

Implementing a Secure Data Transmission Channel Based on Mesh VPN Under Limited IPv4 Addressing

Yaroslav Bielyi[1], Svitlana Sotnik[2]

[1]Department of Computer-Integrated Technologies, Automation, Robotics and Safety Engineering,
Kharkiv National University of Radioelectronics, Kharkiv, Ukraine
e-mail: yaroslav.bielyi@nure.ua
[2]Department of Computer-Integrated Technologies, Automation, Robotics and Safety Engineering,
Kharkiv National University of Radioelectronics, Kharkiv, Ukraine
e-mail: svetlana.sotnik@nure.ua

Abstract: *The work explores the problem of organizing secure remote access to network nodes in conditions of IPv4 address shortage and widespread use of CGNAT technology. It examines modern approaches to building Mesh VPN networks using the WireGuard protocol and NAT Traversal mechanisms. Special attention is given to the principles for establishing a direct P2P connection between nodes via a coordination server and the UDP Hole Punching method, as well as backup relay mechanisms when a direct connection is not possible. As part of the study, an experimental environment was deployed using the Tailscale platform, which included nodes located in different network conditions (home NAT and mobile CGNAT). Testing was carried out on channel availability and latency, as well as checking the operation of an HTTP service through a secure tunnel. The results confirm that the mechanism for organizing secure inter-node communication works without using public IP addresses or manually setting up port forwarding. The proposed approach can be used for building remote monitoring systems, IoT solutions, and secure administration of distributed network infrastructures in situations with limited IPv4 addressing.*

Keywords—technologies; Mesh VPN; WireGuard; NAT Traversal, CGNAT, IoT, remote access.

1. INTRODUCTION

With the growing number of connected devices and the exhaustion of the IPv4 address space, internet providers are widely using Carrier-Grade Network Address Translation (CGNAT). This makes the traditional way of accessing a device from outside through «port forwarding» impossible, since the subscriber doesn't have a unique public IP address.

The problem is especially acute in mobile networks, where connecting to home servers or surveillance systems becomes impossible without using third-party cloud intermediaries, which lowers the level of privacy and security.

According to RIPE NCC, the last blocks of IPv4 addresses in the EMEA region ran out back in 2019 [1], and the transition to IPv6, despite being technically ready, is still slow because of providers' stagnant infrastructure and outdated user equipment. In this situation, CGNAT has turned from a temporary measure into a long-term reality for millions of subscribers around the world.

The consequences of this phenomenon go far beyond just being a technical hassle. The inability to establish a direct connection between nodes forces users and organizations to rely on centralized cloud intermediaries – VPN services, connection brokers, and remote access platforms. This creates several critical issues: dependence on the availability of a third-party service, routing traffic through infrastructure not controlled by the user, risks of confidential data leaks, and extra financial costs. For sensitive applications – industrial monitoring systems, video surveillance, medical devices, corporate infrastructure – such compromises are unacceptable.

Alongside addressing the problem, the number of autonomous devices that require a constant secure connection without human operator involvement is rapidly growing. The IoT concept assumes that network nodes exchange data on their own, send telemetry, and receive control commands in real time [2-3]. In a CGNAT environment, ensuring such autonomous interaction between devices becomes a separate engineering task, requiring not only solving the addressing issue but also automating the processes of establishing and maintaining secure connections.

An alternative to the centralized approach is Mesh VPN technology, which implements a decentralized model for building secure networks. Unlike traditional VPN solutions with a dedicated server hub, Mesh VPN aims to provide direct encrypted connections between network nodes, using NAT traversal mechanisms, and if a direct channel can’t be established, it relies on backup relay nodes.

Among modern implementations of this approach, Tailscale, Netbird, and ZeroTier stand out, each offering its own model for key management, authentication, and routing.

Despite the growing interest in Mesh VPN deployments, practical application under CGNAT conditions and experimental evaluation of secure connection establishment remain insufficiently explored in domestic literature. This highlights the relevance of this work and the need to develop well-founded recommendations for choosing and deploying

Mesh VPN to build secure data transmission channels in environments with limited IPv4 addressing.

Thus, the issue of organizing secure remote access to network nodes under limited IPv4 addressing and the use of CGNAT technology is extremely relevant.

The purpose of this work is to explore the possibilities of Mesh VPN technology for setting up a secure data transmission channel between nodes that are behind network address translation mechanisms (CGNAT).

So, to achieve this goal, the following tasks are planned:

- Analyze the current issues of organizing remote access amid the shortage of IPv4 addresses and the use of CGNAT technology.
- Explore the operating principles of Mesh VPN technology and the mechanisms for bypassing network address translation (NAT Traversal).
- Set up a test environment based on the Tailscale platform to connect nodes located in different networks.
- Conduct an experimental check of node availability and the possibility of data transfer through a secure tunnel.
- Evaluate the effectiveness of the proposed solution based on accessibility, latency, and practical suitability for remote monitoring systems and the Internet of Things (IoT).

2. Related Work

Analysis of current scientific publications and practical developments shows a rapid increase in interest in the problem of ensuring reliable remote access under the conditions of a limited IPv4 address space. The issue of public IP address exhaustion and the specifics of how carrier-grade address translation (CGNAT) technologies work are being studied in detail both by international networking institutions and by individual scientific communities engaged in designing distributed information systems.

Foreign and domestic researchers have made significant contributions to studying the vulnerabilities of classic remote access architectures and analyzing network latency. In particular, works related to building modern Internet of Things (IoT) ecosystems consistently emphasize the critical importance of automating the connection processes between autonomous devices without human operator involvement [4, 5]. The specifics of processing telemetry data and ensuring stable communication in industrial applications and information-measuring systems are highlighted in studies [6, 7], which stress the need to create fault-tolerant and secure information transmission channels.

The development of virtual private network protocols has gone from heavy, centralized solutions (like IPsec and OpenVPN) to lightweight next-generation protocols. A turning point in this field was the creation of the WireGuard protocol, which is known for fast cryptographic operations at the OS kernel level, use of modern encryption (ChaCha20, Poly1305), and minimal overhead traffic – these features are documented in the core works of the protocol developers and confirmed by numerous independent benchmarks. However, the original WireGuard architecture by default assumes a static configuration like 'client-server' or 'point-to-point' (P2P), which requires at least one node to have a fixed public IP address. This significantly limits its direct use in mobile networks or when using CGNAT.

In modern scientific literature, solving the problem of establishing direct connections between nodes located behind different types of NAT (NAT Traversal) [14, 15] is most often based on the STUN (RFC 8489), TURN (RFC 8656) standards, and the ICE (RFC 8445) methodology [16, 17, 18]. Researchers evaluate the effectiveness of the UDP Hole Punching method, showing that the success of tunnel punching directly depends on the symmetry and port configuration of routers.

Recent work in network engineering focuses on moving from traditional hub-and-spoke topologies to full mesh topologies [19, 20]. Platforms like Tailscale, Netbird, and ZeroTier represent a new class of software-defined networks (SDN) [21], which automate the exchange of cryptographic keys and the coordination of endpoint parameters through cloud or isolated control servers (Control Plane), while data transfer (Data Plane) happens in a decentralized way [22].

Despite the availability of commercial reviews, there's a lack of experimental research in domestic scientific publications that evaluates delay parameters (RTT) and the overall stability of direct P2P Mesh VPN connections directly in local 4G/LTE networks, where CGNAT is implemented in the strictest ways (symmetric NAT with dynamic port changes). This creates the need for practical testing and forming recommendations for implementing Mesh VPN for remote monitoring, which is what this work covers.

Despite a significant number of studies in the field of network technologies, most of them focus on the theoretical aspects of NAT Traversal, cryptographic protection, or individual remote access protocols. At the same time, there's not enough attention paid to a comprehensive practical analysis of Mesh VPN solutions in real network conditions, especially in mobile operator environments, where CGNAT with strict address translation policies is widely used. Experimental studies that evaluate the actual latency, stability, and quality of P2P connections in such conditions are also limited.

So, analyzing the scientific sources shows the need for further experimental studies to evaluate the effectiveness of Mesh VPN solutions, particularly based on Tailscale, Netbird, and ZeroTier, in real-world usage scenarios. It's worth paying special attention to practically testing the stability of P2P connections, the impact of different NAT types on the success of UDP Hole Punching, and assessing the performance of backup relay mechanisms when a direct connection isn't

possible. This highlights the relevance of the experimental study conducted in this work.

3. Modern Problems of Organizing Remote Access in Conditions of IPv4 Address Shortage and Using CGNAT Technology

The rapid increase in the number of devices connected to the Internet has led to the depletion of the available IPv4 address space. As a result, many providers use CGNAT technology, which allows a single public IP address to be shared among a large number of users.

Using CGNAT allows for more efficient use of the limited IPv4 address resource, but it creates a number of restrictions when setting up remote access to devices. In traditional networks, access to internal services can be arranged by configuring port forwarding on a router. With CGNAT, this approach becomes impossible because the user doesn't have direct control over the public IP address used to access the Internet.

This problem is especially noticeable in mobile networks, where CGNAT technology is used almost everywhere. As a result, users can't directly connect to home servers, surveillance systems, IoT devices, or other network services.

To provide remote access in such conditions, people often use centralized cloud services or traditional VPN solutions. However, classic VPNs like OpenVPN or WireGuard in a 'client-server' setup also need a server with a public IP address. So, the problem isn't completely solved – it's usually worked around by using a separate node with direct internet access.

Given the spread of IoT devices, telemetry systems, and remote administration, a current task is finding solutions that can ensure secure data exchange between nodes when using CGNAT. One such approach is using Mesh VPN technology, which allows you to set up secure connections between nodes without needing a public IP address on each of them.

4. Principles of Mesh VPN Technology Operation and NAT Traversal Mechanisms

Mesh VPN technology is one of the modern approaches to building secure virtual networks, where each node can directly interact with other network participants. Unlike the traditional client-server VPN architecture, where all traffic goes through a central gateway, Mesh VPN uses a decentralized interaction model that ensures efficient data routing and reduces dependence on individual network nodes. A general topology of a Mesh VPN network is shown in Fig. 1.

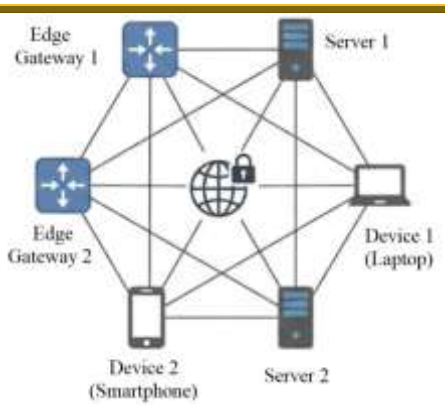

Fig. 1. *Generalized topology of a Mesh VPN network*

One of the main problems when setting up direct connections between nodes is the use of network address translation (NAT) mechanisms. Since devices behind NAT or CGNAT usually don't have a publicly accessible IP address from the Internet, establishing a direct connection between them requires using special methods to bypass these limitations.

To solve this problem, modern Mesh VPN solutions use NAT Traversal technologies. At the initial stage, nodes exchange service information through a coordination server and receive the data needed to establish a connection. In the implementation considered here, the coordination server combines signaling and STUN functionalities, providing both the exchange of service information between nodes and the determination of their public endpoint parameters. The coordination server handles control plane functions, while data transfer between nodes occurs within the data plane.

One of the most common ways to get around NAT is UDP Hole Punching, which allows nodes to initiate a direct connection even when address translation is in place, as shown in detail in Fig. 2. The effectiveness of this method directly depends on the type of NAT the node is behind – the compatibility of UDP Hole Punching with different NAT types is discussed in detail in Table 1.

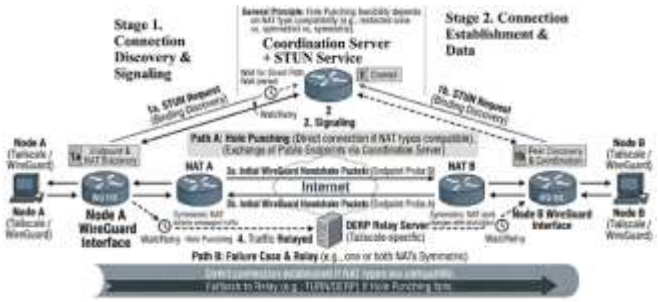

Fig. 2. *The process of establishing a direct, secure connection between Mesh VPN nodes using the UDP Hole Punching method*

Fig. 2 shows the two-step process for establishing a direct secure connection between Mesh VPN nodes in NAT conditions.

In the first step (Connection Discovery & Signaling), each of the nodes (Node A and Node B) sends a request to determine its public endpoint (STUN functionality of the coordination server) and receives a response with information about its public endpoint – the external IP address and port corresponding to its location behind the NAT. After that, the coordination server performs signaling, exchanging information about the nodes' public endpoint parameters.

In the second stage (Connection Establishment & Data), the nodes simultaneously send each other the initial WireGuard handshake packets through their respective NAT devices (NAT A and NAT B). This process, known as UDP Hole Punching, forces each NAT to open a temporary entry in its translation table, allowing packets from the opposite side to pass through. After successful mutual NAT traversal, an encrypted P2P tunnel is established between the WireGuard interfaces of the nodes, through which user traffic is then transmitted directly between the end devices without involving any intermediate servers.

If the NAT Traversal procedure is successful, data is transmitted directly between the end devices through an encrypted tunnel. In this mode, the coordination server doesn’t take part in the user traffic transfer, which helps reduce delays and improve network performance. However, if a direct channel can’t be established due to network configuration or NAT restrictions, traffic can go through intermediate relay nodes while still keeping end-to-end encryption.

One implementation of the Mesh VPN concept is the Tailscale platform, which is built on the WireGuard protocol. Tailscale uses a coordination server to authenticate nodes and exchange service information, after which it tries to establish a direct connection between network participants. This approach ensures secure data exchange between devices behind NAT or CGNAT, without the need to set up your own VPN server with a public IP address.

To make it possible to establish connections between nodes behind network address translation (NAT) mechanisms, the Tailscale platform implements the NAT Traversal mechanism described above using the WireGuard protocol. After determining the public endpoint parameters and performing a UDP Hole Punching procedure, the system tries to create a direct encrypted P2P channel between the nodes (Path A). If the attempt fails, the system keeps retrying to establish a direct connection for a set period and then switches to a fallback data transmission mode (Path B). If a direct connection isn't possible due to specific network configurations, the fallback DERP mechanism (Designated Encrypted Relay for Packets) is used. This is a TURN-compatible relay that ensures traffic transit while keeping end-to-end encryption. In this way, Tailscale reliably establishes secure connections between nodes behind NAT and CGNAT.

The effectiveness of using the UDP Hole Punching method directly depends on the type of NAT the node is behind. A comparative overview of the main NAT types and their compatibility with this method is shown in Table 1.

Table 1: Comparative characteristics of NAT types and compatibility with the UDP Hole Punching method

#	***Tun NAT***	***Translation characteristics***	***UDP Hole Punching***	***Note***
1	Full Cone NAT	Any external host can send packets to the open external port after the NAT mapping is created	It works	Most favorable case
2	Restricted Cone NAT	Allows packets only from IP addresses that the node has previously contacted	It works	Needs prior exchange
3	Port Restricted Cone NAT	Filtering by IP and port number	It works	May require more precise coordination
4	Symmetric NAT	Different external ports are created for different recipients	Usually doesn't work	Usually requires a relay (DERP/TURN)

Note. In real networks, NAT behavior depends on the specific implementation of the router or provider's equipment. So, the possibility of successfully using UDP Hole Punching is determined not only by the type of NAT but also by its configuration features.

5. Setting up a Test Environment Based on the Tailscale Platform to Connect Nodes Located in Different Networks

For this task, the tool Tailscale was chosen, which is based on the WireGuard protocol. Unlike traditional VPNs that use a Client-Server architecture and require a central gateway

with a public IP address, Tailscale implements a network with a mesh topology. The main advantage is the use of NAT Traversal.

The coordination server is used for authenticating nodes, exchanging service information, and public endpoint parameters, after which traffic between nodes is transmitted directly through an encrypted tunnel.

To test the effectiveness of the technology, a test setup was deployed consisting of two nodes located in different networks.

Node 1 (Server): A personal computer running Linux OS. Connection: home provider (NAT). Internal address in the overlay network: 100.99.50.36.

Node 2 (Client): Xiaomi Redmi 4A smartphone (Android). Connection: 4G/LTE mobile network using CGNAT. Internal address: 100.121.144.77.

Both devices have the Tailscale client installed and are authorized on the same network.

Fig. 3 and Fig. 4 show the connection status of both devices to the network.

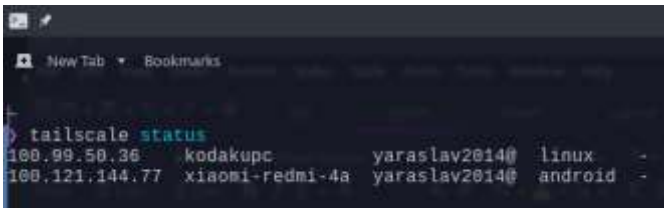

Fig. 3.*Connection status of Node 1 (Linux, home NAT)*

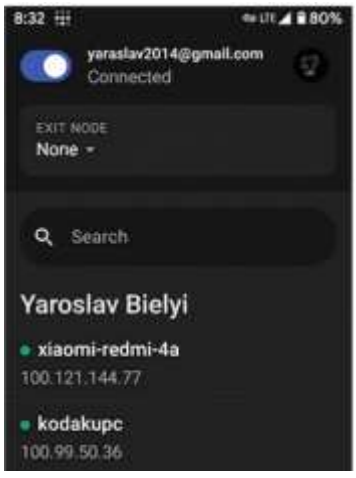

Fig. 4.*Connection status of Node 2 (Android, CGNAT)*

Node accessibility was tested using the ping utility. The measured delay (ping) results between nodes through the Tailscale tunnel are shown in Fig. 5.

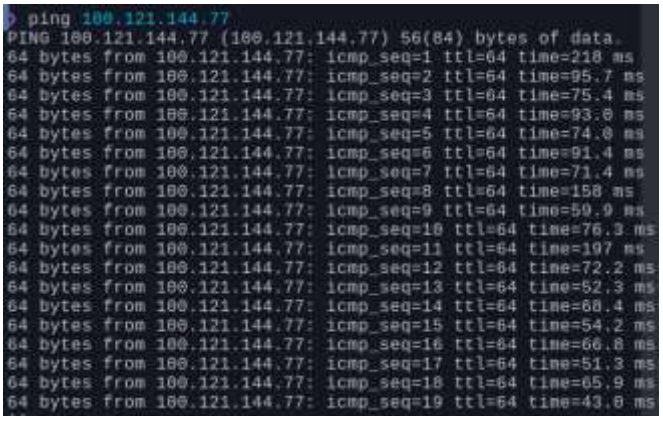

Fig. 5.*Ping measurement results between nodes through the Tailscale tunnel*

The analysis of the results shows behavior typical for a NAT Traversal mechanism: the first packet experiences higher latency due to the process of establishing a P2P tunnel – performing the UDP Hole Punching procedure and exchanging WireGuard cryptographic parameters. Once this process is complete, the latency stabilizes, confirming the establishment of a direct connection between the nodes, bypassing intermediate relays. The values obtained are acceptable for a 4G/LTE mobile network, considering the presence of CGNAT.

To simulate the operation of a web service or IoT control panel, an HTTP server was set up on Node 1 using Python (python3-m http.server 8000), with the log of its activity shown in Fig. 6.

From Node 2, a request was made through a web browser to http://100.99.50.36:8000.

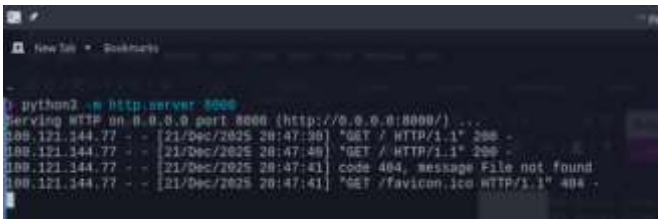

Fig. 6.*HTTP server log on Node 1: successful reception of requests from Node 2 (100.121.144.77) through the Tailscale tunnel*

As seen in Fig. 6, the server successfully accepted a connection from Node 2 with the address 100.121.144.77 — which corresponds to the internal address of the Tailscale overlay network. The main GET / HTTP/1.1 request received a 200 response code, confirming successful data transfer

through the secure tunnel. The 404 responses are related to the missing favicon.ico file and do not affect the test results.

Thus, the experiment confirmed the possibility of setting up a full network connection between a node in a home network with NAT and a mobile device behind CGNAT without using a public IP address or configuring port forwarding.

The result of the request from Node 2 is shown in Fig. 7.

The 200 response code confirms successful data transfer through the secure tunnel.

Fig. 7 confirms the successful data transfer through the Tailscale secure tunnel: the contents of Node 1's file system are fully displayed in Node 2's browser, showing that the HTTP connection between nodes in different networks with different types of NAT is working correctly.

The results show that the Tailscale-based Mesh VPN technology provides a functional, secure data transmission channel between nodes located behind NAT and CGNAT, without needing to use a public IP address, set up port forwarding, or involve third-party cloud intermediaries. The established connection has acceptable latency for a 4G/LTE mobile network and ensures end-to-end traffic encryption using the WireGuard protocol.

Fig. 7. *Displaying the contents of the HTTP server directory of Node 1 in the browser of Node 2 through the Tailscale tunnel*

The practical usefulness of the proposed approach is confirmed by successfully simulating a scenario of remote access to a web service or an IoT device control panel: the HTTP server on Node 1 correctly handled requests from Node 2, which connected through a mobile network with strict CGNAT. This shows that Mesh VPN technology can be effectively used to build secure channels in remote monitoring systems, video surveillance, and the Internet of Things, where IPv4 addressing is limited.

6. CONCLUSION

The research confirmed the feasibility of using Mesh VPN technology to set up secure remote access to nodes in diverse network environments. The goal of the study was achieved, and the experiment demonstrated the possibility of organizing a secure data transfer channel between nodes located behind network address translation mechanisms, without needing a

public IP address for each node and without complex network infrastructure setup.

In the course of the study, the current problems of organizing remote access in conditions of IPv4 address shortages and widespread use of CGNAT technology were analyzed. It was found that traditional approaches – port forwarding and classic client-server VPN solutions – do not fully solve the problem under CGNAT conditions, since they require a node with a public IP address. Mesh VPN technology, implemented on the Tailscale platform and using the WireGuard protocol, allows for secure connections between nodes even when CGNAT is in place. It combines the STUN protocol for determining public endpoint parameters, the UDP Hole Punching method for establishing direct P2P connections, and a backup DERP relay mechanism in cases where a direct connection is impossible due to network configuration specifics.

The expanded test environment consisted of two nodes in isolated networks: a personal computer running Linux connected through a home NAT, and a mobile device connected via a 4G/LTE network with strict CGNAT. Latency measurement results confirmed the successful establishment of a direct encrypted tunnel between the nodes, and HTTP connection testing showed proper data transmission through the secure channel. The latency values obtained do not interfere with the operation of typical remote monitoring services, web access, or IoT applications.

An important feature of Mesh VPN technology is the ability to build a network with elements of a Full Mesh topology, which allows for direct secure connections between nodes without the need to use a central gateway for user traffic. The combination of modern cryptographic mechanisms of the WireGuard protocol and NAT Traversal tools makes this solution practically suitable for building Internet of Things systems, remote monitoring, and administration of network devices when using NAT and CGNAT.

At the same time, the study has certain limitations: the experiment was conducted on a testbed with two nodes using only the Tailscale platform, and channel bandwidth measurements were not performed. Future research directions include a comparative analysis of alternative Mesh VPN solutions – specifically Netbird and ZeroTier, performance testing with an increased number of nodes, as well as assessing the applicability of the technology in industrial monitoring and critical infrastructure management systems.

The practical value of the study is that the proposed approach can be directly applied to solving real engineering problems without high financial costs and without the need for deep knowledge in network administration.

7. REFERENCES

[1] All European IPv4 addresses, including spares, are used up. (2019, November 26). GIGAZINE. Retrieved from https://gigazine.net/gnews/20191126-europe-out-of-ipv4-addresses/

[2] Nestorenko, D. O., et al. (2026). Analysis of modern IoT solutions for automation of urban parking spaces. Modern science: trends, challenges, solutions. Proceedings of the 8th International scientific and practical conference. Cognum Publishing House. Liverpool, United Kingdom, 94-103

[3] Taran, A., et al. (2026). Impact of 5G/6G Networks on the Development of IOT, Robotics, and Autonomous Systems. Low Latest and Mass Connection of Devices. All-Ukrainian Conference "Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations" (ICSTRO-2026) February 12-13, 114-118

[4] Marunich, R. V., et al. (2025). Features of IoT application in the security sector. «Computer-integrated technologies, automation and robotics» CITAR-2025, 80-84

[5] Khalimonov, Y., et al. (2024). Approaches to ensuring proper working conditions using sensor technologies IoT. International Conference «DIGITAL INNOVATION & SUSTAINABLE DEVELOPMENT 2024», 24-25

[6] Mathews, M. J., Vosloo J. C., Prinsloo J. (2020). The value of a telemetry monitoring system in sustaining the operational performance of industrial information systems. South African Journal of Industrial Engineering. – Т. 31. – №. 2, 129-142

[7] Prinsloo, J., et al. (2019). Development of a software-based monitoring and information system for industrial telemetry applications. South African Journal of Industrial Engineering. – Т. 30. – №. 1, 54-68

[8] Rizaldi, I. D., Suhartana M., Hassan R. (2025). Performance and Security Assessment of IPSec VPN Implementations Using Various Network Devices. 2025 International Conference on Information Management and Technology (ICIMTech), 420-424

[9] Fitri, R. N., et al. (2025). OSPF-Driven Assessment of SSTP, PPTP, and L2TP/IPSec VPN Protocols for FTP-SFTP Services. 2025 IEEE 15th Symposium on Computer Applications & Industrial Electronics (ISCAIE), C. 97-101.

[10] Demirdelen, T., Kırmızı, S. (2025). Performance Evaluation of WireGuard and IPSec Protocols in Various Network Configurations. Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi. – Т. 8. – №. 3, 1353-1369

[11] Hussein, S. G., Rehman, S. M. F. U. (2025). Protocol Efficiency and Resource Utilization in VPN Technologies: A Comparative Analysis of OpenVPN and WireGuard. Journal of Techniques. – Т. 7. – №. 4, 37-42

[12] WireGuard. (2026). *Performance*. Retrieved June 18, 2026, from https://www.wireguard.com/performance/?source=post_page-----8a270f5d97e9

[13] Purba, C. M., Ibnugraha, P. D. (2025). Securing Data Transmission in IoT Devices Using Wireguard VPN. 2025 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS), 1-5

[14] Olson, K., Wampler, J., Keller, E. (2023). Doomed to repeat with IPv6? Characterization of NAT-centric security in SOHO routers. ACM Computing Surveys. – T. 55. – №. 14s, 1-37

[15] Seemann, M., Inden, M., Vyzovitis, D. (2022). Decentralized hole punching. 2022 IEEE 42nd International Conference on Distributed Computing Systems Workshops (ICDCSW), 96-98

[16] Petit-Huguenin, M. et al., (2020). Session traversal utilities for NAT (STUN). – №. rfc8489. https://www.rfc-editor.org/rfc/rfc8489.html

[17] Reddy, T., et al. (2020). Traversal using relays around NAT (TURN): Relay extensions to session traversal utilities for NAT (STUN). Proposed Standard of the Internet Engineering Task Force (IETF). https://datatracker.ietf.org/doc/html/rfc8656

[18] Keranen, A., Holmberg, C., Rosenberg, J. (2018). Interactive connectivity establishment (ICE): A protocol for network address translator (NAT) traversal. – №. rfc8445. https://datatracker.ietf.org/doc/html/rfc8445

[19] Klophaus, R., Merkert, R., Lordan, O. (2021). Mesh network as a competitive advantage for European LCCs: An alternative topology to hub-and-spoke for selling online connections. Transport Policy. – T. 106, 196-204

[20] Roig, P. J., et al. (2023). Arithmetic Study about Efficiency in Network Topologies for Data Centers. Network. – T. 3. – №. 3, 298-325

[21] Kjorveziroski, V., et al. (2024). Full-mesh VPN performance evaluation for a secure edge-cloud continuum //Software: Practice and Experience. – T. 54. – №. 8, 1543-1564

[22] Das, D., et al. (2023). Blockchain enabled sdn framework for security management in 5g applications. Proceedings of the 24th International Conference on Distributed Computing and Networking, 414-419