Enhancing the security of Cloud Computing infrastructure and E-mail systems through the implementation of Mailcow and pfSense platforms

Ksenija Zrnić[0009-0009-8700-5536], Dragan Đokić[0009-0004-6887-1205] and Katarina Jovanović[0009-0007-7703-3854]

Faculty of Information Technology, Belgrade Metropolitan University, Belgrade 11000, Serbia
ksenija.zrnic@metropolitan.ac.rs
dragan.djokic@metropolitan.ac.rs
katarina.jovanovic.4556@metropolitan.ac.rs

Abstract. Cloud Computing is becoming a key technology in modern IT environments; however, many organizations continue to face numerous challenges in ensuring the security and reliability of their e-mail systems. The goal of this paper is to present the enhancement of cloud infrastructure through the implementation of Mailcow and pfSense platforms, emphasizing security, reliability, stability, and efficiency of e-mail systems. The implementation was carried out in a virtual environment using the VMware Type 2 hypervisor. The Docker Mailcow server was deployed on an Ubuntu 18.04 operating system, while the pfSense software was configured as a firewall solution for network traffic control and virtual infrastructure protection. All virtual machines within the virtual environment were connected to pfSense, enabling centralized management, filtering, and monitoring of network operations. The results demonstrated that the combination of Mailcow and pfSense platforms improved performance, stability, and security of the e-mail system, while simplifying user privilege and access management. This research may serve as a practical model for educational and business organizations aiming to enhance their e-mail and network systems through the use of open-source technologies in cloud environments.

Keywords: Cloud Computing, Mailcow, PfSense, e-mail server, security, virtualization.

1 Introduction

An increasing number of organizations, regardless of their size, rely on electronic mail (e-mail) as a primary channel of communication. Efficient and reliable e-mail communication is crucial for business processes and success in the modern world. However, managing e-mail can be challenging, especially in environments with many users, complex configurations, and high security requirements.

This is where the Docker Mailcow server [1] stands out — a powerful mail management tool providing a comprehensive solution for creating, configuring, and

Research Paper
DOI: https://doi.org/10.46793/BISEC25.147Z
Part of ISBN: 978-86-89755-40-4

maintaining electronic mail servers. Docker Mailcow enables organizations and individuals to manage their mail systems efficiently, ensuring secure and reliable communication. Additionally, this paper includes the implementation of the pfSense software [2], an open-source firewall solution that allows for network traffic control, filtering, and monitoring, thereby providing an additional level of infrastructure protection.

By combining these platforms, it is possible to build a reliable, stable, and secure e-mail system within a cloud environment, without relying on commercial services.

The goal of this paper is to present the process of implementing the Mailcow and pfSense platforms in a virtual VMware environment [3], and to analyze their efficiency in terms of security, stability, and performance. The testing results indicate that it is possible to build a scalable and secure system applicable to both educational and business environments.

The structure of the paper is organized as follows: Section two presents the theoretical background of cloud computing technology, including service types, deployment models, and key implementation challenges. Section three describes the pfSense firewall/router software, while Section four focuses on the Mailcow server. Section five provides a detailed explanation of the implementation of ESXi [4], Mailcow, and pfSense solutions in a VMware environment. Section six discusses the results and performance analysis of the implemented solution, followed by conclusions in Section seven.

2 Cloud Computing

2.1 Introduction of Cloud Computing

Cloud Computing represents a model for delivering computing services over the Internet, including infrastructure, databases, software, and data storage. Through this model, companies do not own their IT infrastructure or systems but instead access the necessary resources via the Internet [5].

Cloud computing offers agility, flexibility, and scalability. By adopting this model, companies reduce operational and maintenance costs while enabling faster and more flexible resource allocation and scaling compared to traditional infrastructure [5].

2.2 Types of Cloud Computing

As cloud computing continues to evolve, several deployment models have emerged. There are four primary types, each requiring its own specific security practices [5]:

1. Public Cloud: A model in which the network, storage, and computing infrastructure are owned by the cloud provider and services are delivered via the Internet. Users do not need to maintain their own data centers, which is particularly advantageous for startups and growing companies. Multiple users share the same infrastructure, while the provider ensures data security and isolation. Public clouds are the most cost-effective model due to large-scale resource sharing [5].

2. Private Cloud: Used exclusively by one organization and operated on dedicated servers and infrastructure. It can be hosted in a provider's data center or on the client's premises. Although more expensive and less flexible than public clouds, it offers the highest level of security and control — typically used by government institutions [5].
3. Hybrid Cloud: A combination of public and private clouds used by one organization. Sensitive applications are hosted on a private cloud for better control, while less critical workloads run on a public cloud. This model provides flexibility and enables gradual migration to the cloud [5].
4. Multicloud: The use of two or more cloud service providers. This approach allows organizations to leverage the strengths of each provider while reducing dependency on a single vendor. It often results from company mergers or diverse technology initiatives. With interoperability, applications can communicate between multiple clouds and on-premises data centers [5].

2.3 Types of Cloud Computing services

Within the field of cloud computing, three distinct service models are recognized, and users may combine multiple services, either from a single provider or across several different providers, depending on their specific requirements [5]:

1. Software as a Service (SaaS): A model in which cloud providers develop, host, and maintain software applications that users access online without local installation. The provider manages technical aspects such as security, updates, and data storage. The pricing model typically follows a "pay-as-you-go" principle, reducing the need for license purchases. SaaS benefits include quick availability, automatic updates, and lower costs [5].
2. Platform as a Service (PaaS): A model where the provider offers a complete platform for developing, testing, and running applications, including necessary tools, databases, and infrastructure. Developers focus solely on building and managing applications, while the provider maintains servers, networks, and security. PaaS often integrates with AI, IoT, analytics, and databases, enabling advanced functionality and faster development [5].
3. Infrastructure as a Service (IaaS): A model where the provider supplies complete infrastructure — servers, storage, and networking — while users manage operating systems and applications. Billing is based on actual resource usage, such as CPU, RAM, and storage, optimizing cost efficiency. IaaS is ideal for organizations migrating existing data centers to the cloud due to its flexibility and scalability [5].

2.4 Advantages of Cloud Computing

In the modern era, due to rapid technological advancements and the exponential growth of data, traditional data centers and server rooms have become insufficient, lacking the agility and flexibility required to meet contemporary business needs. This is precisely where cloud computing proves its value, offering numerous advantages [5]. Advantages of cloud computing compared to traditional IT systems: [5]

1. Lower costs — pay per resource usage.
2. Speed — resources are provisioned within hours rather than months.
3. Global scalability — easy expansion through worldwide data centers.
4. Higher productivity — less maintenance time, more focus on development.
5. Better performance — improved efficiency and responsiveness.
6. Optimization — cloud applications outperform traditional software.
7. Reliability — fault-tolerant, scalable, and highly available systems.

2.5 Challenges in Cloud Computing

Despite its benefits, cloud computing presents challenges requiring careful management and planning.

The most significant is security and compliance, as data protection is a shared responsibility between the provider and the organization. Companies must control data access, prevent leaks, and comply with data protection regulations [5].

Another challenge is cost management — while cloud solutions lower capital expenses, poor resource management may lead to unexpectedly high costs. Continuous monitoring and optimization are essential [5].

Lack of expertise is also a growing concern, as rapid technology evolution demands highly skilled professionals capable of managing complex cloud architectures [5].

Multicloud data management adds complexity when organizations use multiple cloud platforms, making interoperability and data integration critical [5].

Finally, portability remains important — organizations should avoid vendor lock-in and embrace containerized, cloud-native technologies for easy migration between environments [5].

In addition to the previously mentioned challenges, there are further issues such as the migration of legacy systems, variations in service offerings across different cloud regions, and the need for a deeper understanding of each individual provider's services and models. All these factors require a strategic approach, continuous education, and the application of best practices in cloud infrastructure management [5].

3 PfSesne firewall/router software

PfSense is a free, open-source firewall and router software based on the FreeBSD operating system. It is known for its flexibility and provides advanced network traffic control through custom rules. Although pfSense is accessible due to its open-source model and is often used in small and medium-sized environments where cost-effectiveness is important, it is also widely adopted by large enterprises. Its reputation as a reliable open-source network security solution has earned strong trust among organizations worldwide.

A key feature of pfSense is scalability and compatibility across various hardware and cloud platforms. PfSense has an active community and forum support, while commercial support is available through Netgate and its partners, who provide extensive, continuously updated documentation [6].

The main features of pfSense include: Stateful Packet Inspection (SPI) firewall, Network Address Translation (NAT), VPN server (OpenVPN, IPsec), IDS/IPS support (Snort, Suricata), detailed traffic logging, and visual traffic reports.

In cloud environments, pfSense is often used as a gateway providing inbound and outbound traffic control, as well as network segmentation (e.g., LAN zone isolation).

4 Mailcow server

Mailcow is an open-source server solution with its own license and usage policies [7]. It represents a comprehensive system for e-mail implementation and management, including modern security mechanisms such as spam protection, antivirus filtering, calendar and contact synchronization, and shared resources [8].

Its intuitive interface and active user community contribute to ease of use, configuration of advanced functions, setup of security features, and faster issue resolution [9]. Mailcow is scalable and suitable for complex environments with many users, and regular updates by the development team further enhance its stability and security [8].

Mailcow provides a complete e-mail infrastructure with advanced security mechanisms: SPF, DKIM, and DMARC authentication, TLS/SSL encryption, spam and malware filtering, and two-factor authentication for user accounts.

Mailcow represents a modern alternative to commercial e-mail systems, offering complete data control within a self-hosted cloud environment.

5 Implementation of selected technologies

Due to the limited availability of hardware resources on the physical computer, the implementation of the entire system was carried out using nested virtualization. On the physical machine running the Windows 11 operating system [10], a type two hypervisor — VMware Workstation 17.0 [3] — was installed, enabling the creation and management of multiple virtual machines within a single physical system. This approach provided a layered and functional environment for demonstrating and testing network and e-mail services, as well as security mechanisms, without the need for additional physical resources.

Within the VMware Workstation environment, a virtual structure consisting of three main virtual machines was created, encompassing several layers. On one of these virtual machines, the VMware ESXi 8.0.1 [4] type one hypervisor was installed. For demonstration purposes, two additional virtual machines were deployed on ESXi — one running Windows 10 [11] and the other running Linux Ubuntu 18.04 [12] — between which communication was established through the Mailcow server [1].

In addition, a separate virtual machine running the Ubuntu 18.04 operating system was created, on which the Docker Mailcow server was implemented. This server acts as the central e-mail server, enabling message exchange between users and serving as a platform for testing various security mechanisms. The Mailcow server uses Docker container-based virtualization, providing numerous advantages for the organization. The e-mail server is implemented as a set of Docker containers, allowing for better utilization of hardware resources and increased system scalability. Docker containers are lightweight, quickly instantiated, and easily scalable based on system requirements, providing flexibility in adding or removing containers without interrupting system operation.

The Docker Mailcow server includes a wide range of advanced functionalities. It also incorporates built-in mechanisms for filtering and blocking spam messages, which reduces system load. Security is further enhanced through integrated SSL/TLS encryption support.

The Docker Mailcow server also contributes to improving overall system security. Docker containers provide isolation between containers and the host system, preventing potential attacks and damage to the system. Containers also simplify the process of managing and maintaining security updates, as they can be individually upgraded and tested.

The third virtual machine within the VMware environment was designated for the installation of pfSense 2.7.2 [2] firewall/router software, which serves as the central security component of the system — functioning as the primary firewall. All other machines in the network are connected to this virtual machine, through which Internet access is controlled (i.e., all outbound network traffic to the Internet passes through pfSense). Such an architecture enables an additional layer of security through centralized access control and allows for more efficient monitoring and filtering of network threats.

In the schematic representation shown in Figure 1, the rectangles illustrate different layers and components of the virtual infrastructure. The largest rectangle represents the physical machine, which serves as the base layer of the infrastructure. Inside it is the frame labeled as the type two hypervisor — VMware 17.0 — encompassing all virtual machines used in this study. Within this framework, the left section depicts the ESXi hypervisor and its two internal virtual machines — Windows 10 and Ubuntu 18.04. The central section represents the Mail Server running Ubuntu 18.04 with the Docker Mailcow system installed, while the right section illustrates the pfSense virtual machine functioning as a protective and primary network gateway.

The implementation of the Docker Mailcow server and pfSense firewall combination provides the organization with a modern and efficient solution for e-mail communication and network security. This architecture reduces operational costs, increases flexibility and scalability, and enhances the overall security of network communications. Through centralized network access management and advanced traffic filtering options, the organization achieves a stable and secure IT environment. This optimization allows resources to be focused on core business activities.

Figure 1 illustrates the architecture of the described solution, depicting the nested structure of virtual machines, their interconnections, and the role distribution within the VMware environment.

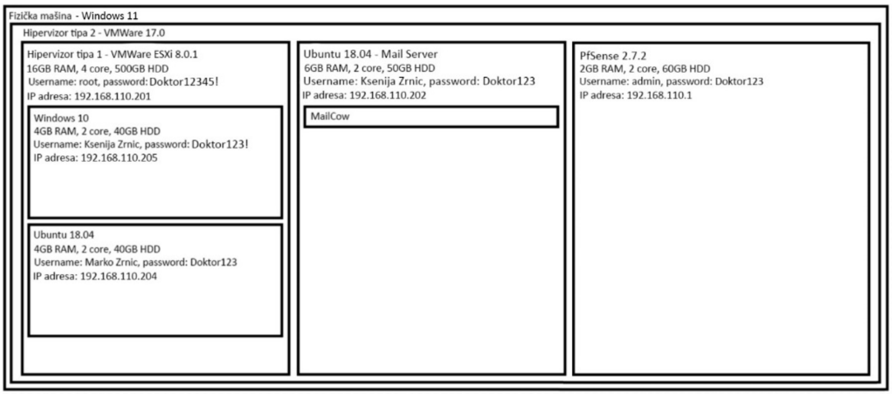

Fig. 1. System architecture diagram

5.1 Implementation of the pfSense software

Within the proposed system, pfSense 2.7.2 [2] was installed and configured in the VMware [3] virtual environment, serving as the central security layer responsible for network traffic control.

PfSense was installed on a dedicated virtual machine with two network interfaces — WAN (NAT network) and LAN (host-only network). This setup enables separation between the internal and external networks, achieving higher levels of isolation and security. The host-only network connects pfSense with other machines (ESXi, Mailcow), while the NAT network provides WAN access.

After installation, the default LAN IP address was modified, and initial configuration was completed via the web interface, allowing centralized management of firewall rules and access control. Integrating pfSense into the system provided centralized traffic management, protection against unauthorized access, and filtering of data based on protocols and ports.

PfSense ensured stable and secure communication between the Mailcow server [1], ESXi hypervisor [4], and client machines, significantly improving the reliability and security of the virtual infrastructure.

5.2 Implementation of the Mailcow server

In the VMware [3] environment, a virtual machine named MailServer was created using the Linux Ubuntu 18.04 [12] operating system, where the Docker Mailcow server [1] was installed and configured.

This solution was chosen due to its simplicity, modularity, and high component isolation achieved through Docker containers. The MailServer machine was connected to the pfSense firewall [2], which provides network traffic control and access protection.

During initial configuration, hostname, FQDN, and static IP address were defined, ensuring stable and predictable communication. After installing Docker and Docker Compose, the Mailcow Dockerized package was downloaded and launched, containing all necessary services such as Postfix, Dovecot, Rspamd, MariaDB, Redis, and Nginx.

Each service runs in a separate container, simplifying maintenance, isolating resources, and enabling updates of individual components without affecting the entire system.

After installation, Mailcow was configured through a web interface. Domains and mailboxes were created, and internal communication between accounts was tested successfully. For additional protection, two-factor authentication (2FA) was enabled using Google Authenticator [13], ensuring a higher level of administrative and user security.

Integrating Docker Mailcow into this environment provided centralized user and domain management, enhanced communication security, scalability, reduced administrative costs, and simplified maintenance.

Such an implementation represents a reliable and flexible solution for managing electronic mail within a virtualized infrastructure, ensuring a high level of security, stability, and administrative control.

5.3 Implementation of the ESXi hypervisor

Within the VMware [3] environment, VMware ESXi 8.0.1 [4] type one hypervisor was installed and configured as the foundation for managing and running multiple virtual machines. ESXi was chosen for its reliability, efficient resource isolation, and centralized management capabilities.

The hypervisor was deployed within VMware Workstation, with allocated CPU, memory, disk, and network adapters enabling connectivity with pfSense [2] and Mailcow [1]. After setup, the ESXi Host Client was accessed via a web interface for system management.

Inside ESXi, two virtual machines were created — Windows 10 [11] and Linux Ubuntu 18.04 [12] — used for demonstrating e-mail communication via Mailcow. Each user was assigned specific privileges: Ksenija accessed the Windows machine, Marko the Linux one, while the administrator managed both.

After configuring DNS parameters, successful bidirectional communication between accounts marko@mailing.com and ksenija@mailing.com confirmed the system's full functionality.

The implementation of VMware ESXi enabled centralized management, user privilege segmentation, and the efficient simulation of enterprise-level e-mail and network systems. This architecture establishes a stable foundation for the development, testing, and enhancement of cloud and security solutions within corporate environments.

6 Results of the implementation of pfSense and Mailcow technologies

By implementing the proposed solution, which includes the Docker Mailcow server [1], VMware ESXi hypervisor [4], and pfSense firewall [2], a stable, efficient, and secure information system was achieved. The realization was carried out within the VMware environment [3], where virtual machines with different operating systems were created, while pfSense enabled centralized control of network traffic and provided an additional layer of protection.

6.1 Results achieved through the implementation of the ESXi hypervisor

The use of the VMware ESXi hypervisor [4] demonstrated significant advantages in the areas of virtualization and resource management.

Advantages of the VMware ESXi hypervisor:

1. Efficient virtualization: The ESXi hypervisor enables isolation and resource sharing on a physical server between multiple virtual machines. This means that several operating systems and applications can run on a single physical server without requiring dedicated hardware for each machine. This leads to more efficient utilization of hardware resources.
2. High reliability: The ESXi hypervisor is known for its high reliability. It is less prone to errors and security vulnerabilities. It also has the ability to perform critical tasks without the need for rebooting, which reduces downtime and increases availability.
3. Resource management: The ESXi hypervisor allows flexible resource management. It is possible to adjust the allocation of processors, memory, storage space, and network resources for each virtual machine according to needs. This enables optimal resource utilization, higher scalability, and better performance.
4. Centralized system management: The ESXi hypervisor can be integrated with centralized management systems such as VMware vCenter Server. This allows centralized management of virtual machines, access to management consoles, performance monitoring, high availability, machine migration, and more. Centralized management simplifies administration and increases efficiency.
5. Security and isolation: The ESXi hypervisor provides a high level of security and isolation between virtual machines. Each virtual machine is isolated from others, preventing issues on one from affecting the rest. It is also possible to apply security policies and access controls at the level of virtual machines and resources.
6. Support for various operating systems: The ESXi hypervisor supports a wide range of operating systems, including Windows, Linux, and others. This allows different applications and operating systems to run on the same server without conflict.

6.2 Results achieved through the implementation of the Mailcow server

The implementation of the Docker Mailcow server [1] contributed to greater security and reliability in electronic mail exchange.

Advantages of the Docker Mailcow server:

1. Simple installation and management: Docker Mailcow simplifies the installation and configuration of an e-mail server. All required components are packaged within Docker containers, which eliminates the need for manual configuration of individual infrastructure components. Docker Mailcow

also provides simple command-line tools and utilities for managing the server.

2. Containerization: Docker Mailcow uses Docker containers to isolate and manage different components of the e-mail server. Containers ensure that each component has its own separate environment and dedicated resources, preventing potential interference between them. This makes the server stable and reliable, allowing for easy updates and scalability of individual parts of the infrastructure.
3. Flexibility and scalability: Docker Mailcow enables easy scalability of the e-mail server. Docker containers can be added or removed depending on resource needs. This allows for efficient load management and adaptation to changing organizational requirements.
4. Security: Docker Mailcow has built-in security features that help protect the e-mail server from attacks and malicious activities. Docker containers provide isolation between components, reducing the risk of spreading infections or attacks to other parts of the system. Additionally, Docker Mailcow offers tools for configuring and enforcing security policies, as well as mechanisms for protection against spam and malicious e-mails.
5. Automation and resource management: Docker Mailcow supports automation, allowing automated processes such as deployment, scaling, updating, and monitoring of the e-mail server. It also facilitates easy management of resources such as CPU, memory, and storage space across different containers.
6. Community and support: Docker Mailcow has an active user community and support network. Extensive documentation, guides, and resources are available to assist with setup and management of the e-mail server. The community also provides troubleshooting support and problem resolution through forums and communication channels.

6.3 Results achieved through the implementation of the pfSense software

The implementation of the pfSense software [2] as the central firewall significantly increases the level of network security and simplifies traffic management.

Advantages of the pfSense firewall/router server:

1. Enhanced network security: The implementation of pfSense firewall software considerably strengthens the security of the network infrastructure. pfSense provides advanced functionality with the ability to define detailed rules for each network segment. All traffic between the internal network and the Internet passes through pfSense, which blocks unauthorized access attempts and potentially harmful packets. This centralized approach to security reduces the risk of system intrusion and enhances the protection of business data.
2. Network traffic filtering: PfSense allows precise filtering of network traffic based on various criteria (IP address, port, protocol, domain, and others). In this way, both security and productivity are improved — by blocking

potentially malicious or phishing websites and by limiting employee access to social networks and other non-productive content during working hours.

3. Protection against network attacks: PfSense includes tools for defending against various types of Internet-based attacks. In addition to basic firewall filtering, it supports the integration of IDS/IPS systems (such as Snort or Suricata), which can be installed as additional packages within pfSense to detect and prevent intrusions. Furthermore, pfSense offers features such as rate limiting and protection against DoS/DDoS attacks at lower network levels. All suspicious activity is recorded in log files, allowing administrators to monitor events in real time and react promptly (for example, by updating rules to block specific IP addresses or address ranges).
4. Centralized management and network resources: The introduction of pfSense improved overall control of network traffic and simplified network administration. PfSense provides an intuitive web interface for configuring all aspects of the network — from DHCP and DNS services to VLAN creation and access control rules. Administrators can manage network resources from a single point of control. This greatly simplifies supervision — every service within the virtual infrastructure (Mailcow, as well as virtual machines hosted on ESXi) now communicates with the external network exclusively through pfSense, where parameters such as bandwidth, priorities (QoS), and time-based rule schedules can be precisely configured. This approach ensures optimal utilization of network resources and bandwidth, while maintaining a high level of security.

6.4 Results of the business solution implementation

Performance of the implemented business solution:

1. Resource efficiency: The use of Docker containers and virtual machines on the ESXi hypervisor [4] enables better utilization of hardware resources. Containers and virtual machines can be optimized for performance and flexibly distributed across available resources, resulting in more efficient use of hardware capacity.
2. Scalability: The Docker Mailcow server [1] and ESXi hypervisor support system scalability. It is possible to easily add or remove containers and virtual machines as needed, adapting to changes in e-mail demand. This allows for rapid adjustment to business growth or fluctuations in the number of users, without the need for large investments in new hardware.
3. Reduction of operational costs: The combined use of the Docker Mailcow server and virtual machines on the ESXi hypervisor contributes to reducing operational costs. Maintaining physical servers requires more resources, time, and financial investment, while virtual machines and containers offer greater cost-efficiency and simplified management.
4. Increased reliability and security: Docker containers and virtual machines enable application isolation, which enhances overall system reliability. Problems or incidents within one container or virtual machine do not affect

other parts of the system. In addition, Mailcow has built-in e-mail protection mechanisms and security standards that contribute to greater safety. The integration of the pfSense firewall [2] adds an additional layer of network defense, significantly reducing the risk of intrusion or unauthorized access to sensitive data. The combination of an isolated mail server and a robust firewall guarantees high operational reliability (the system remains functional even in the event of individual component failures) and multi-layered protection.

5. Improved scalability and agility: The transition to a Docker Mailcow server and virtual machines enables faster and easier adaptation to changes in the business environment. The system's capacity can be expanded or reconfigured without physical intervention or hardware modification. Moreover, centralized network management through pfSense allows instant modification of security rules or network configurations as required by business operations.

7 Conclusion

This paper presents the implementation of the Docker Mailcow server [1] and pfSense firewall [2] within a VMware virtual environment [3] as a cost-effective, efficient, and reliable solution for improving e-mail systems and network security.

The installation and configuration of the pfSense firewall/router software enabled centralized management of network traffic and enhanced overall system security. By configuring WAN and LAN interfaces, defining access control rules, and activating filters to block unwanted content, a stable foundation was created for the secure operation of the Mailcow server. This integration provides complete control over network flows and an additional layer of protection against potential threats.

The implementation of the Docker Mailcow server facilitated efficient management, simplified installation and administration, easier scalability, and faster service updates. The use of Docker containers within the virtualized VMware environment enabled more flexible resource management and more efficient utilization of hardware capacity.

The results of the analysis indicate that the proposed solution has proven to be a practical model for building a modern and scalable IT system that ensures reliable communication, simplified administration, and a high level of security. Due to its modern and adaptable structure, this solution provides a foundation for further development and integration of additional services in line with the organization's growth and evolving needs.

The proposed model can serve as a best-practice example for other organizations seeking to modernize their IT systems, particularly in educational and business environments where security, reliability, and cost-efficiency are critical factors. Future research may focus on implementing automated monitoring and management, integrating with large-scale cloud infrastructures, and applying real-time anomaly detection systems.

References

1. Mailcow Team, "Mailcow: dockerized documentation," Information and Support, [Online]. Available: https://docs.mailcow.email. [Accessed: 2023/06/11].
2. Netgate, "Download pfSense," pfSense.org, [Online]. Available: https://www.pfsense.org/download/. [Accessed: 2024/06/14].
3. VMware Inc., "Download VMware Workstation Pro Evaluation," VMware Official Website, [Online]. Available: https://www.vmware.com/products/desktop-hypervisor/workstation-and-fusion. [Accessed: 2023/06/10].
4. VMware Inc., "Download VMware vSphere Hypervisor (ESXi)," VMware Official Website, [Online]. Available: https://www.vmware.com/products/cloud-infrastructure/vsphere. [Accessed: 2023/06/10].
5. Oracle, "What is Cloud Computing?" [Online]. Available: https://www.oracle.com/europe/cloud/what-is-cloud-computing/. [Accessed: 2025/10/01].
6. IT and General, "About pfSense," [Online]. Available: https://www.itandgeneral.com/about-pfsense/. [Accessed: 2024/06/12].
7. Mailcow Project, "Mailcow: dockerized," GitHub, [Online]. Available: https://github.com/mailcow/mailcow-dockerized. [Accessed: 2023/06/10].
8. Mailcow Project, "Sponsor @mailcow on GitHub Sponsors," GitHub, [Online]. Available: https://github.com/sponsors/mailcow. [Accessed: 2023/06/10].
9. Mailcow Team, "Spamfilter – Mailcow: dockerized documentation," [Online]. Available: https://docs.mailcow.email/manual-guides/mailcow-UI/u_e-mailcow_ui-spamfilter/. [Accessed: 2023/06/10].
10. Microsoft, "Download Windows 11," Microsoft Software Download, [Online]. Available: https://www.microsoft.com/en-us/software-download/windows11. [Accessed: 2023/06/10].
11. Microsoft Corporation, "Download Windows 10 Disc Image (ISO File)," Microsoft, [Online]. Available: https://www.microsoft.com/en-us/software-download/windows10. [Accessed: 2023/06/11].
12. Canonical Ltd., "Ubuntu Desktop – Download Ubuntu 18.04," Ubuntu Official Website, [Online]. Available: https://ubuntu.com/download/desktop. [Accessed: 2023/06/11].
13. Google LLC, "Google Authenticator," Google Play Store, [Online]. Available: https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2. [Accessed: 2024/06/15].