IJMECIE International Journal of Mechanical, Electrical and Civil Engineering
E-ISSN: 3047-4531
P-ISSN: 3047-4523

Research Article

Implementation of Mikrotik Network Management Using Quality of Service Features with the Hierarchical Token Bucket Algorithm

Yuma Akbar [1*], Rizki Ananda Pratama [2], Sugiyono Sugiyono [3], Faris Jawad [4],

1 Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika (Stikomcki) Jakarta
2 Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika (Stikomcki) Jakarta
3 Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika (Stikomcki) Jakarta
4 Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika (Stikomcki) Jakarta
* Corresponding Author: e-mail: yumekhan@stikomcki.ac.id

Abstract: This research aims to address the issue of uneven bandwidth distribution in large organizational networks by implementing Quality of Service (QoS) using FIFO and the Hierarchical Token Bucket (HTB) algorithm on Mikrotik routers. Uneven bandwidth distribution can disrupt productivity and operational efficiency. This study creates a fair and efficient traffic management system, allowing bandwidth allocation according to user needs. The methodology involves detailed configuration of Mikrotik RouterOS to optimize QoS with adjusted HTB settings. Testing was conducted using IPerf3 to measure bandwidth variations received by clients in different conditions, including scenarios with two and three active clients. The results indicate that the HTB method provides more stable and consistent bandwidth distribution compared to FIFO. In the two-active client scenario, the unused bandwidth by the third client is allocated to higher priority clients, demonstrating HTB's effectiveness in managing traffic priorities. This research is expected to enhance user satisfaction by providing a network that is both stable and responsive to the needs of various operational applications, and contribute significantly to the development of best practices for bandwidth management in complex organizational environments.

Keywords: Bandwidth Allocation, Hierarchical Token Bucket (HTB), Mikrotik RouterOS, Network Performance, Quality of Service (QoS).

Received: August 15, 2024
Revised: September 12, 2024
Accepted: September 25, 2024
Published: October 15, 2024
Curr. Ver.: October 28, 2024

1. Introduction

The rapid growth of internet usage has made network performance and service quality critical factors in supporting various digital activities. Internet connectivity is no longer limited to communication purposes but has become an essential infrastructure for education, business, public services, and remote work environments. The increasing dependence on online applications requires network providers to ensure stable and equitable bandwidth allocation among users. In practice, however, many community-based internet service providers still face challenges related to bandwidth management, particularly when serving a large number of clients simultaneously. This condition can lead to network congestion, unequal bandwidth distribution, decreased application performance, and user dissatisfaction. A similar situation occurred at Hizam Net, an FTTH-based RTRW Net provider serving approximately 200 clients with a total bandwidth capacity of 400 Mbps. Despite adequate bandwidth resources, customers frequently experienced slow connections and unstable access due to inefficient bandwidth allocation. These circumstances indicate that effective bandwidth management is essential to maintain service quality, optimize resource utilization, and ensure fair internet access for all users within a shared network environment [1], [2].

Numerous studies have investigated bandwidth management and Quality of Service (QoS) implementation using MikroTik routers and the Hierarchical Token Bucket (HTB) algorithm. Previous research demonstrated that HTB can effectively regulate bandwidth allocation, prioritize traffic, and improve network efficiency compared with conventional queue management techniques. Several studies also reported improvements in QoS parameters, including throughput, delay, jitter, and packet loss, after HTB implementation in educational institutions, enterprise networks, and community internet services. Nevertheless, most existing studies primarily focus on technical implementation and comparative performance evaluation between HTB and other queue management methods such as PCQ or Queue Tree. Limited attention has been given to the practical challenges of bandwidth distribution in large-scale RTRW Net environments that serve hundreds of users with diverse traffic demands. Furthermore, the relationship between HTB-based bandwidth allocation and application access stability remains insufficiently explored. Therefore, this study offers novelty by examining HTB implementation within a real-world RTRW Net environment while emphasizing its impact on network performance and application stability under high user density conditions [3], [4].

Based on the identified problems and research gaps, this study seeks to investigate the implementation of bandwidth management using the Hierarchical Token Bucket algorithm on MikroTik routers to improve network performance and application access stability. The research is motivated by the need to establish a more balanced and efficient bandwidth distribution mechanism capable of accommodating increasing internet traffic demands. Specifically, this study addresses two primary research questions. First, how can HTB-based bandwidth management be implemented to improve network performance and maintain application stability in a multi-user environment? Second, what factors influence internet network performance, and how can bandwidth management mitigate performance degradation caused by traffic congestion and unequal resource allocation? To answer these questions, the study evaluates the configuration and application of HTB within an operational network environment and examines its effectiveness in controlling bandwidth consumption according to user priorities. Through this approach, the research aims to provide empirical evidence regarding the role of QoS mechanisms in supporting reliable internet services and improving overall user experience [5], [6].

This study argues that effective bandwidth management is not merely a technical requirement but a strategic necessity for maintaining service quality and customer satisfaction in modern internet-based services. The implementation of HTB enables network administrators to allocate bandwidth according to predefined priorities, prevent excessive resource consumption by individual users, and ensure fair access across the network. By optimizing bandwidth utilization and maintaining consistent QoS levels, network operators can reduce service disruptions and enhance application responsiveness. The contribution of this research lies in providing a practical reference for implementing HTB-based bandwidth management in RTRW Net infrastructures, particularly those operating under high user density and varying traffic patterns. In addition, the findings are expected to enrich the existing literature on MikroTik-based QoS implementation by offering evidence from a real operational environment. The results may also assist network administrators, practitioners, and researchers in designing more efficient bandwidth allocation strategies that support sustainable and high-quality internet service delivery [7], [8].

2. Literature Review

Systematic Literature Review Concept

Systematic Literature Review (SLR) is a literature review approach that is carried out in a structured manner to identify, evaluate, and interpret previous research that is relevant to a particular topic. In the context of research on the implementation of MikroTik network management using the Quality of Service feature with the Hierarchical Token Bucket algorithm, SLR is used to gain a comprehensive understanding of research developments, methods used, and important findings related to bandwidth management. This approach is important because computer network research is not enough to rely solely on technical assumptions, but also needs to be supported by scientific evidence from previous studies. Through SLR, researchers can trace how HTB is applied in various network environments, such as schools, RT/RW Net, wireless networks, and MikroTik-based internet services. In addition, SLR helps clarify the current position of research so that it does not repeat the same study, but rather makes new contributions based on the research gaps found [4], [9].

In this study, SLR is variable through three main stages, namely planning, conducting, and reporting. The planning stage functions to formulate a research question that is the basis for searching for literature. The formulation of research questions is directed at the effectiveness of the HTB method in improving network performance, access stability, and bandwidth allocation efficiency. The conducting stage is carried out by determining search keywords, literature sources, and study selection criteria. The keywords used include "MikroTik", "Hierarchical Token Bucket", "HTB", "Quality of Service", and "bandwidth management". The reporting stage is carried out by compiling the results of the literature review in the form of an academic narrative that explains the relationship between the research problem, the method used, and the contribution produced. With this variabelization, SLR not only becomes a theoretical summary, but also becomes an analytical tool to connect previous research with the needs of bandwidth management implementation on real networks [3], [5].

Indicators in SLR in this study were carried out through the PICOC approach, namely Population, Intervention, Comparison, Outcomes, and Context. Population includes journals and books that discuss MicroTik, QoS, HTB, and bandwidth management. Intervention focuses on the implementation of HTB to optimize the network when many users are accessing the internet at the same time. Comparison is directed at comparing HTB with other methods, such as FIFO, PCQ, Simple Queue, and Queue Tree. Outcomes include improved network stability, bandwidth efficiency, access priority, and application and service performance. The research context is placed on a private network or RT/RW Net environment that has many users with different access needs. In the case of Hizam Net, this indicator is relevant because the network serves many clients with a certain total bandwidth, but still faces complaints of slow connections and uneven bandwidth distribution. Therefore, the PICOC indicator helps to compile the literature in a focused manner and in accordance with research needs [1], [10].

Bandwidth Management and HTB Concepts

Bandwidth management is the process of regulating, dividing, and controlling network capacity so that internet usage can run more efficiently, stably, and fairly for each user. In a network with many clients, large bandwidth does not always guarantee the quality of service if it is not well managed. Without bandwidth management, some users may overuse network capacity resulting in slow connections, high delays, or interruptions in application access. Quality of Service is an important concept in network management because it functions to maintain service quality based on certain parameters, such as throughput, delay, jitter, and packet loss. On MikroTik devices, the QoS feature can be applied through various queue methods, one of which is the Hierarchical Token Bucket. HTB allows administrators to create a tiered bandwidth allocation structure based on priority, so that network allocation can be adjusted to the needs of users and the type of service being run [11], [12].

The variability of bandwidth management in this study includes several key aspects, namely bandwidth allocation, access priority, connection stability, network performance, and user satisfaction. Bandwidth allocation is related to the distribution of internet capacity to each user or group of users so that unbalanced usage does not occur. Access priority relates to setting up certain traffic to keep critical services running even when the network is congested. Connection stability indicates the network's ability to maintain consistent internet access without excessive interruption. Network performance can be measured through QoS parameters, such as access speed, delay, jitter, and packet loss. Meanwhile, user satisfaction is the impact of successful bandwidth management. In the context of HTB, all of these variables are interrelated because the right configuration can limit overusage, prioritize services, and improve network efficiency. Thus, HTB can be a relevant method to overcome the problem of bandwidth distribution on the MikroTik network [6], [8].

Based on the results of the literature review, HTB has the advantage of being able to regulate bandwidth hierarchically through the division of main and subclass classes according to network needs. This mechanism allows administrators to set minimum and maximum bandwidth limits, so that each user still gets a proportionate allocation. In networks that have many users, such as RT/RW Net, HTB can help reduce bandwidth usage inequality and maintain stable application access. Several studies have shown that the implementation of HTB on MikroTik can improve service quality, especially when combined with traffic analysis and appropriate queue configuration. However, the effectiveness of HTB is greatly influenced by the accuracy of configuration, number of users, type of traffic, bandwidth capacity, and periodic network monitoring. Therefore, this study positions HTB not only as a technical feature, but as a network management approach that needs to be designed, implemented, and evaluated systematically in order to be able to improve the performance of internet services in a sustainable manner [7], [13].

Network Management Concepts

A router is a fundamental device in computer networking that functions to connect two or more networks and determine the most appropriate path for data transmission. It operates at the network layer of the OSI model and uses routing tables to forward packets from a source network to a destination network. In practical network environments, routers are not only used to connect local networks to the internet but also to manage traffic, apply security policies, and support service quality mechanisms. MikroTik routers are widely used because they provide flexible routing features, firewall configuration, Network Address Translation, and bandwidth management tools. These capabilities make MikroTik suitable for small, medium, and community-based networks such as RT/RW Net. In this context, routers become central components for ensuring connectivity, regulating traffic flow, and maintaining stable network performance when multiple users access the internet simultaneously [2], [11].

The concept of router functionality can be operationalized through several variables, including routing capability, traffic control, security configuration, and network service management. Routing capability refers to the ability of the router to select efficient paths for packet delivery using static or dynamic routing protocols. Traffic control is related to how the router manages data flow to prevent congestion and maintain network stability. Security configuration includes firewall rules, access control, and address translation mechanisms that protect internal networks from unauthorized access. Meanwhile, network service management involves the use of features such as Quality of Service, queue management, and monitoring tools. In MikroTik-based networks, these variables are important because administrators must configure routers according to user needs and traffic characteristics. Therefore, the effectiveness of a router is not only determined by its hardware capacity but also by the accuracy of configuration and continuous monitoring [1], [14].

Bandwidth refers to the maximum capacity of a communication channel to transmit data within a certain period. In computer networks, bandwidth determines how much data can be transferred between devices, users, or services at a given time. Higher bandwidth generally enables faster access, smoother streaming, and better application performance. However, bandwidth capacity alone does not guarantee good network quality if its distribution is not properly managed. In multi-user networks, unequal bandwidth consumption can cause some users to dominate network resources, while others experience slow connections and unstable access. Therefore, bandwidth must be understood not only as a technical measurement but also as a limited network resource that requires regulation. Bandwidth management is needed to allocate capacity fairly, prevent excessive use, and support service continuity. This is especially important in networks with many users and diverse traffic demands [4], [15].

Bandwidth management can be variabilized through allocation, limitation, prioritization, and performance stability. Allocation refers to the distribution of bandwidth among users, groups, or services based on network policy. Limitation is the process of setting maximum usage limits to prevent bandwidth monopolization by certain users or applications. Prioritization determines which traffic should receive higher service levels, particularly for critical applications such as video conferencing, online learning, or real-time communication. Performance stability reflects the ability of the network to maintain consistent access despite increased traffic load. In MikroTik RouterOS, these variables can be implemented through Simple Queue, Queue Tree, and Hierarchical Token Bucket methods. Simple Queue is commonly used for basic bandwidth control, while HTB provides more structured and hierarchical bandwidth allocation. Thus, bandwidth management becomes a measurable strategy for improving network efficiency, reducing congestion, and supporting user satisfaction [3], [12].

Quality of Service is a network management concept that focuses on maintaining the quality and reliability of data transmission by controlling key performance parameters. These parameters include throughput, latency, jitter, and packet loss. Throughput indicates the amount of data successfully transmitted over the network, while latency refers to the delay experienced by data packets during transmission. Jitter measures the variation in packet arrival time, which is critical for real-time applications such as voice and video communication. Packet loss indicates data packets that fail to reach their destination. In bandwidth management, QoS enables administrators to classify, prioritize, and control traffic according to service requirements. The Hierarchical Token Bucket algorithm is one of the methods used to implement QoS because it allows bandwidth to be distributed in classes and subclasses. This makes HTB relevant for maintaining fair bandwidth distribution and improving network performance [5], [16].

Supporting tools such as Winbox, BTest, GNS3, and VMware Workstation are also important in MikroTik network implementation and evaluation. Winbox provides a graphical interface that helps administrators configure MikroTik routers more efficiently, including routing, firewall, queue, and QoS settings. BTest is used to test network performance by measuring throughput, latency, and packet loss, making it useful for evaluating bandwidth management results. GNS3 supports network simulation by allowing users to design and test virtual network topologies before implementation in real environments. VMware Workstation complements this process by enabling virtual operating systems and isolated testing environments. These tools are relevant because network management requires not only configuration but also testing, simulation, and evaluation. Through the use of these supporting technologies, administrators can identify potential bottlenecks, verify configuration accuracy, and improve the reliability of MikroTik-based bandwidth management before applying it to operational networks [7], [17].

3. Materials and Method

Research Data

The data used in this study consist of quantitative network-related information associated with bandwidth management using the Hierarchical Token Bucket (HTB) method on MikroTik routers. The collected data are private and obtained from several reliable sources to support the research objectives. Secondary data were gathered through literature studies and textbook reviews, including scientific journals, academic publications, technical documentation, and reference books related to bandwidth management, Quality of Service (QoS), MikroTik RouterOS, and HTB implementation. These sources provide the theoretical foundation and empirical evidence required to understand the effectiveness of HTB in optimizing network performance. In addition, observational data were collected through direct monitoring of the network environment under study. The observations focused on bandwidth utilization, traffic distribution, and network performance indicators during the implementation of HTB. Quantitative measurements included the amount of bandwidth allocated and received by connected clients, enabling an objective evaluation of network efficiency. By combining secondary and observational data, this study aims to provide a comprehensive assessment of HTB's ability to improve bandwidth allocation, enhance network stability, and meet user bandwidth requirements effectively.

Methodology Implementation

This study employs an experimental research approach to investigate the implementation of bandwidth management using the Quality of Service (QoS) feature with the Hierarchical Token Bucket (HTB) algorithm on a MikroTik-based computer network. The methodology focuses on evaluating the effectiveness of HTB in optimizing bandwidth allocation and improving overall network performance. The implementation process begins with the collection and analysis of network traffic data to identify traffic characteristics, user behavior patterns, and bandwidth consumption trends. Based on this analysis, relevant network parameters are determined and used to design an appropriate bandwidth allocation strategy. The HTB algorithm is then configured on the MikroTik router to manage bandwidth hierarchically, allowing administrators to assign priorities and bandwidth limits according to network requirements. After the configuration stage, a series of performance tests are conducted to evaluate the impact of HTB on network efficiency, traffic distribution, and service stability. Through this experimental implementation, the study aims to demonstrate how HTB can enhance bandwidth utilization, reduce network congestion, ensure fair resource allocation, and maintain optimal performance in a multi-user network environment.

Testing Design

The testing design in this study aims to evaluate the effectiveness of the Hierarchical Token Bucket (HTB) method in managing bandwidth on a MikroTik-based network environment. The testing process begins with the collection of baseline network information, including the number of connected clients, internet access capacity, and bandwidth requirements for each user. This information is used to identify traffic characteristics and determine appropriate bandwidth allocation policies. The HTB algorithm is then configured on MikroTik RouterOS to distribute bandwidth hierarchically according to predefined priorities and user requirements. Several testing scenarios are conducted, including variations in the number of active clients, peak traffic conditions, and different bandwidth priority levels. To support the experiment, VMware Workstation is used to create a virtualized server environment, while iPerf3 is utilized to generate traffic and measure network performance. The proposed network architecture used during the experiment is illustrated in Figure 1, which presents the topology design, the interconnection between network components, and the placement of MikroTik RouterOS as the central bandwidth management controller implementing the HTB algorithm.

The following is the topology of the network infrastructure design using the HTB method:

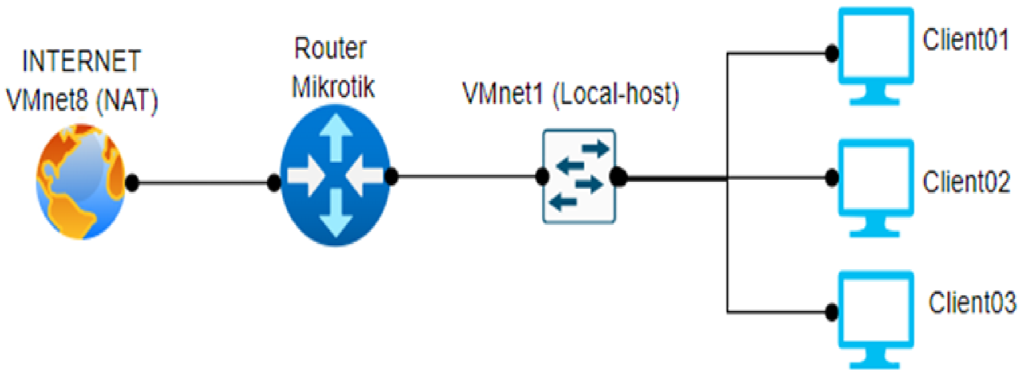

Figure 1. Topological Plan.

After the implementation phase, network performance is evaluated by comparing conditions before and after HTB deployment. The assessment focuses on bandwidth allocation consistency, fairness of bandwidth distribution, traffic prioritization effectiveness, and the ability of the network to maintain stable performance under varying workloads. Measurements obtained through iPerf3 are analyzed to determine whether HTB successfully improves bandwidth utilization and reduces network congestion. In addition, the evaluation examines how the hierarchical bandwidth allocation mechanism affects different categories of users and applications. The overall testing procedure is summarized in Figure 2, which presents the flowchart of the experimental stages, beginning with network analysis and data collection, followed by HTB configuration, traffic generation, performance measurement, and result evaluation. This structured testing framework enables a systematic assessment of HTB performance and provides empirical evidence regarding its effectiveness in improving network management efficiency and ensuring more balanced bandwidth distribution in multi-user environments.

The following is a flowchart in the testing stage:

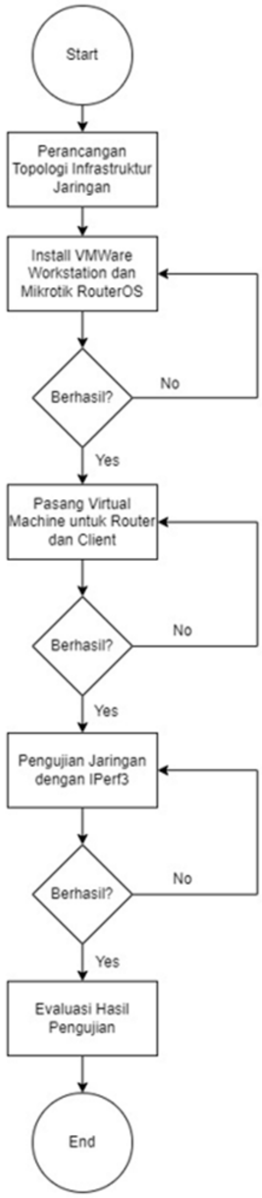

Figure 2. Test Stage Flowchart.

4. Results and Discussion

Research Tools

This study utilized a combination of hardware and software components to support the implementation and evaluation of MikroTik network management using the Quality of Service (QoS) feature with the Hierarchical Token Bucket (HTB) algorithm. The primary physical device used in the experiment was a Lenovo X1 Carbon laptop equipped with 16 GB of RAM, a four-core processor, a 512 GB SSD, an Ethernet network interface, and the Windows 10 operating system. The laptop served as the host machine for running a virtualized network environment through VMware Workstation. Within the virtualization platform, several virtual machines were deployed, including a MikroTik RouterOS instance configured with 1 GB RAM, one CPU core, and 2 GB storage, as well as three Ubuntu 24.04 client machines, each allocated 4 GB RAM, two CPU cores, and 20 GB of storage. The virtual network environment enabled the simulation of a multi-user network scenario and provided a controlled platform for bandwidth management experiments using the HTB algorithm.

To support network configuration, monitoring, and performance evaluation, several software applications were employed throughout the research process. Windows 10 functioned as the host operating system for virtualization activities, while MikroTik RouterOS served as the primary network operating system responsible for implementing QoS policies and HTB-based bandwidth allocation. Winbox was used as the graphical management interface for configuring and monitoring MikroTik RouterOS. The MikroTik Queue feature was utilized to implement hierarchical bandwidth management policies and traffic prioritization mechanisms based on HTB principles. Additionally, iPerf was installed on the Ubuntu client machines to generate network traffic and measure bandwidth performance through TCP and UDP testing. The integration of these hardware and software components provided a comprehensive experimental environment for evaluating bandwidth allocation efficiency, network stability, and the overall effectiveness of HTB-based QoS implementation in a simulated multi-client network infrastructure.

Table 1. Hardware Specifications.

No.	Device	Hardware Type	Specification
1	Lenovo X1 Carbon Laptop (Physical Host)	RAM	16 GB
		CPU	4 Cores
		Storage	512 GB SSD
		Network Interface	Ethernet LAN
		Operating System	Windows 10
2	MikroTik RouterOS Virtual Machine	RAM	1 GB
		CPU	1 Core
		Storage	2 GB
		Network Interface	NAT, Local Host
		Operating System	MikroTik RouterOS
3	Ubuntu Client 1	RAM	4 GB
		CPU	2 Cores
		Storage	20 GB
		Network Interface	Local Host
		Operating System	Ubuntu 24.04
4	Ubuntu Client 2	RAM	4 GB
		CPU	2 Cores
		Storage	20 GB
		Network Interface	Local Host
		Operating System	Ubuntu 24.04
5	Ubuntu Client 3	RAM	4 GB
		CPU	2 Cores
		Storage	20 GB
		Network Interface	Local Host
		Operating System	Ubuntu 24.04

Table 2. Software Specifications.

No.	Software/Application	Description
1	Windows 10	Host operating system running on the physical laptop and used to execute virtualization software.
2	MikroTik RouterOS	Network operating system used for router configuration and implementation of Quality of Service (QoS) with the Hierarchical Token Bucket (HTB) algorithm.
3	Winbox	Graphical management application used to configure, monitor, and manage MikroTik RouterOS devices.
4	MikroTik Queue	Built-in RouterOS feature used to implement QoS policies and hierarchical bandwidth allocation using HTB.
5	iPerf3	Command-line network testing utility installed on Ubuntu clients to measure bandwidth, throughput, and network performance using TCP and UDP traffic.

Implementation and Testing

At this stage, the study implemented a virtualization-based environment to simulate MikroTik network management using the Quality of Service (QoS) feature with the Hierarchical Token Bucket (HTB) algorithm. Virtualization was used to create independent virtual machines on a physical host, allowing each virtual machine to run its own operating system, applications, and network configuration without affecting other machines. The physical laptop running Windows 10 served as the host system, while VMware Workstation was used as the virtualization platform. In this environment, one MikroTik RouterOS virtual machine was configured as the main router and gateway, while three Ubuntu virtual machines were used as clients for network performance testing. This virtualized infrastructure provided a controlled environment for implementing QoS rules and evaluating bandwidth allocation. The testing focused on measuring network speed, traffic distribution, and the effectiveness of HTB in managing bandwidth fairly among multiple clients. Through this implementation, the study aimed to assess whether MikroTik QoS configuration could improve bandwidth control and maintain stable network performance.

Here's an overview of virtual machine virtualization using VMware Workstation software:

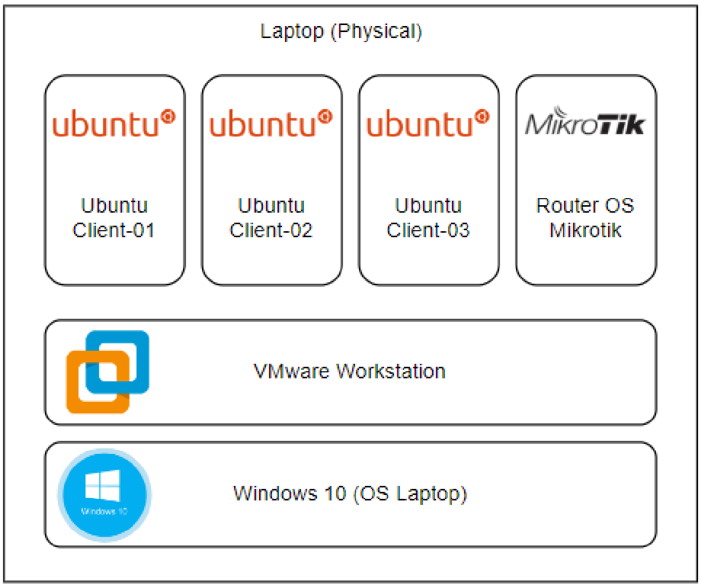

Figure 3. VMware Workstation Virtualization.

Virtual Server Simulator Installation

The initial stage of implementation involved preparing the virtualization platform by installing VMware Workstation 17 Pro version 17.5.2. This software was used to create and manage virtual machines required for the experimental network environment. Before installing the virtual machines, the virtual network infrastructure was configured through the Virtual Network Editor in VMware Workstation. Two virtual network interfaces were prepared, namely VMnet8 and VMnet1. VMnet8 was configured as a NAT interface with DHCP enabled and a subnet address of 192.168.113.0/24, allowing the MikroTik virtual router to access the internet through the physical host. Meanwhile, VMnet1 was configured as a host-only interface with DHCP disabled and a subnet address of 192.168.115.0/24. This interface was used to connect the MikroTik router with the Ubuntu client machines in an isolated local network. This configuration ensured that each virtual machine could communicate according to the designed topology while maintaining separation between internet-facing and internal testing networks.

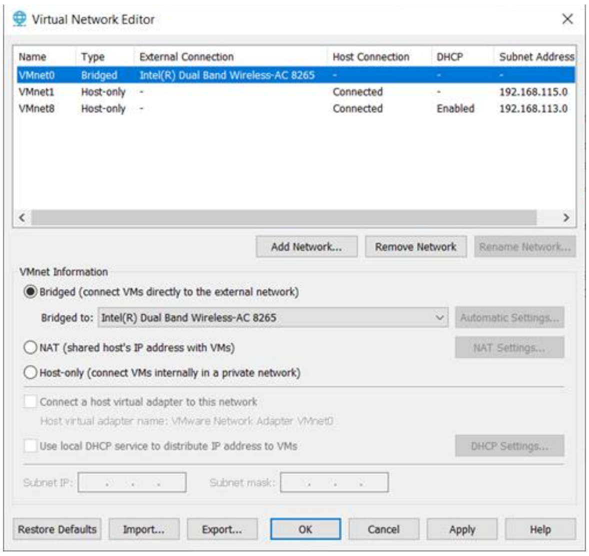

Figure 4. Virtual Network Editor.

On the Virtual Network Editor page, you can see several parameters such as the name of the virtual interface, the type of virtual interface, and the type of interconnection that exists in each virtual interface. There are three types of virtual interfaces that can be used when adding new interfaces or customizing existing ones. Each of these types of virtual interfaces has different functions.

Here is the configuration of the Virtual Network used on the VM host to be installed:

Tabel 1. Virtual Network VMWare.

No	Parameter	Keterangan
1	Name	VMnet8
	Type	NAT
	Host Connection	*Enabled*
	DHCP	*Enabled*
	Subnet Address	192.168.113.0/24
2	Name	VMnet1
	Type	Host-only
	Host Connection	*Enabled*
	DHCP	*Disabled*
	Subnet Address	192.168.115.0/24

Virtual Operating System Installation

After the virtual network infrastructure was prepared, the next stage was installing the operating systems on each virtual machine. MikroTik RouterOS was installed as the main virtual router responsible for gateway functions, Network Address Translation, and QoS configuration using the HTB method. In addition, three Ubuntu virtual machines were installed as client nodes to generate traffic and evaluate bandwidth distribution. The MikroTik virtual machine was imported using an OVA file, then its network adapters were adjusted so that the first adapter used NAT for internet access and the second adapter used VMnet1 for local client communication. The Ubuntu virtual machines were created using the "Typical" configuration option in VMware Workstation, with Ubuntu ISO files mounted during installation. Each Ubuntu client was assigned sufficient virtual resources, including processor cores, memory, and disk capacity. The installation process included selecting the language, keyboard layout, disk configuration, user profile, and timezone. Once installed, all virtual machines were integrated into the experimental topology for QoS testing.

The design of the virtualization topology used can be seen in the following image:

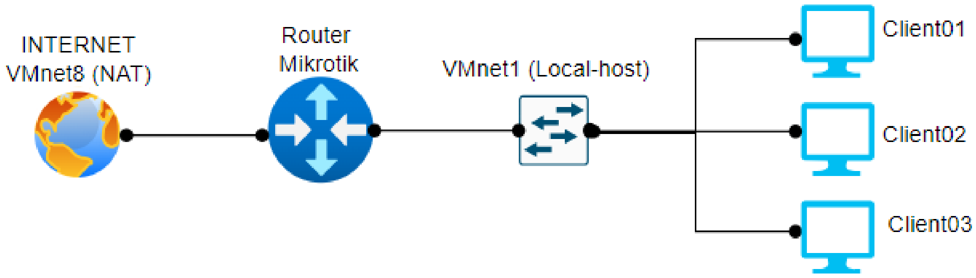

Figure 1. Installation Topology.

MikroTik QoS and Client Network Configuration

After the virtual machines were installed, network configuration was performed on both the Ubuntu clients and the MikroTik router. The Ubuntu clients were initially configured using DHCP, but static IP addresses were later assigned to ensure stable communication during testing. Client01 used 192.168.115.1/24, Client02 used 192.168.115.2/24, and Client03 used 192.168.115.3/24, with 192.168.115.254 as the gateway. The network interface was identified using the ifconfig command, and static IP settings were applied through the Netplan configuration file. On the MikroTik router, ether1 was configured as a DHCP client connected to the NAT network, while ether2 was assigned the static IP address 192.168.115.254/24 as the gateway for the Ubuntu clients. A NAT masquerade rule was then added to allow the clients to access the internet through ether1. After connectivity was verified, QoS rules were created using FIFO and HTB methods. The FIFO scenario limited each client to 2 Mbps, while the HTB scenario applied 1 Mbps minimum bandwidth and 2 Mbps maximum bandwidth with different priorities.

Here is the IP Address table for the Ubuntu host:

Tabel 4. IP Client Ubuntu.

Hostname	IP Address	Prefix	Gateway
Ubuntu-Client01	192.168.115.1	/24	192.168.115.254
Ubuntu-Client02	192.168.115.2	/24	192.168.115.254
Ubuntu-Client03	192.168.115.3	/24	192.168.115.254

Name	Target	Upload Max Limit	Download Max Limit	Upload Limit At	Download Limit At	Upload Priority	Download Priority
Cond3-Parent-QOS	192.168.115.1, 192.168.115.2, 192.168.115.3	3M	3M	unlimited	unlimited	8	8
Cond3-Client03	192.168.115.3	2M	2M	1M	1M	3	3
Cond3-Client02	192.168.115.2	2M	2M	1M	1M	2	2
Cond3-Client01	192.168.115.1	2M	2M	1M	1M	1	1

Figure 6. HTB QOS Configuration.

Quality of Service Testing Using FIFO and HTB

Quality of Service testing was conducted after the FIFO and HTB configurations were applied on the MikroTik router. The first scenario tested three active clients using the FIFO method, where all clients were assigned a maximum upload and download limit of 2 Mbps. In this scenario, iPerf3 was executed on each Ubuntu client to generate traffic and measure the bandwidth received. The results showed fluctuating bandwidth values among clients, indicating that FIFO processed traffic based on arrival order without structured priority control. The second scenario tested three active clients using HTB, with each client assigned 1 Mbps minimum bandwidth and 2 Mbps maximum bandwidth. The results were more stable, with each client receiving bandwidth close to the guaranteed minimum allocation. The third scenario tested two active clients using HTB, showing that bandwidth could be distributed more flexibly when fewer clients were active. Overall, the HTB method demonstrated better bandwidth consistency, fairer allocation, and stronger priority control compared with FIFO in the simulated MikroTik network environment.

Final Test Results

After implementing each stage starting from simulation installation, OS installation, then continued with QoS configuration using FIFO and HTB methods, the following are the results of each test that has been carried out.

Hasil Pengujian Quality of Service (QoS) Metode FIFO dan HTB

Quality of Service (QoS) testing was conducted to evaluate the performance and effectiveness of two bandwidth management methods, namely First In First Out (FIFO) and Hierarchical Token Bucket (HTB). These methods were applied in several testing scenarios involving multiple clients within the network to determine how each approach affected bandwidth distribution and service consistency. FIFO is a simple queueing method that processes data packets based on their arrival order, where the first packet received is the first packet served. In this study, FIFO was tested with three active clients to observe bandwidth variation and fluctuation among users. In contrast, HTB is a hierarchical bandwidth management method that allows traffic to be classified and processed based on predefined priorities. HTB provides greater flexibility in allocating minimum and maximum bandwidth limits to each client. Therefore, HTB testing was conducted in two scenarios, involving three active clients and two active clients, to evaluate its performance under different network conditions and assess its ability to maintain more stable bandwidth distribution.

The following is a comparison table of QoS test results using FIFO and HTB methods:

Tabel 5. Hasil Pengujian FIFO dan HTB.

Testing Aspects	FIFO	HTB
Client01	1.78, 0.429, 0.359, 1.37	0.961, 0.927, 1.07, 0.973
Client02	0.961, 0.938, 1.30, 1.07	0.985, 0.996, 0.984, 0.997
Client03	0.232, 1.09, 1.78, 0.382	0.973, 0.996, 0.985, 0.973

The table above shows the comparison of the bandwidth performance received by each client when using the FIFO and HTB methods with three active clients. This data provides an overview of the differences in bandwidth distribution that occur in each method when applied under the same network conditions.

Table 6. Test Results 3 Conditions.

Metode	Client	Bandwidth (Mbits/sec)
FIFO	Client01	1.78, 0.429, 0.359, 1.37
	Client02	0.961, 0.938, 1.30, 1.07
	Client03	0.232, 1.09, 1.78, 0.382
HTB (Tiga Client)	Client01	0.961, 0.927, 1.07, 0.973
	Client02	0.985, 0.996, 0.984, 0.997
	Client03	0.973, 0.996, 0.985, 0.973
HTB (Dua Client)	Client01	1.89, 1.83, 1.90, 1.84
	Client02	1.09, 1.08, 1.09, 1.07

To further clarify the test results, the following is a graph showing the bandwidth variation for each method and client that has been tested. These graphs provide a more understandable visualization of the bandwidth performance comparison between FIFO and HTB methods.

Graph 1: Bandwidth variation for three clients using the FIFO method.

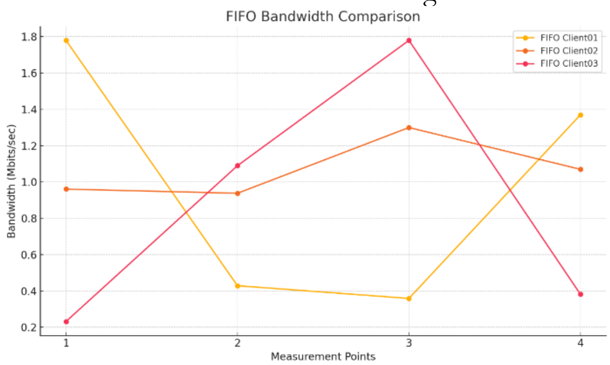

Figure 7. Graph 1 : FIFO Summary.

The graph illustrates the bandwidth distribution received by three clients (Client01, Client02, and Client03) when the First In First Out (FIFO) bandwidth management method was applied. FIFO processes network traffic strictly according to the order in which packets arrive, without considering traffic priority or service requirements. The results indicate significant variations in bandwidth allocation among the clients during the testing period. Client01 experienced bandwidth values ranging from 0.359 Mbps to 1.78 Mbps, while Client02 received bandwidth between 0.938 Mbps and 1.30 Mbps. Client03 showed the largest fluctuation, with bandwidth values varying from as low as 0.232 Mbps to as high as 1.78 Mbps. These variations demonstrate that bandwidth distribution under FIFO is highly dependent on traffic arrival patterns and network congestion levels. As a result, some clients may receive substantially lower bandwidth than others at certain times, leading to inconsistent network performance. Overall, the graph confirms that the FIFO method does not provide stable or predictable bandwidth allocation, making it less effective for maintaining consistent Quality of Service in a multi-user network environment.

Graph 2: Bandwidth variation for three clients using the HTB method with three active clients.

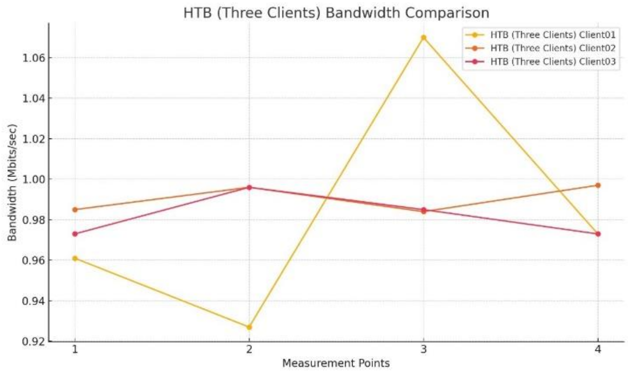

Figure 8. Graph 2: HTB 3 Client Comparison.

The graph presents the bandwidth distribution received by three clients (Client01, Client02, and Client03) when the Hierarchical Token Bucket (HTB) method was implemented with all three clients active simultaneously. Unlike FIFO, HTB provides a hierarchical bandwidth allocation mechanism that enables administrators to define bandwidth priorities, minimum guaranteed rates, and maximum bandwidth limits for each client. The testing results demonstrate a more balanced and consistent distribution of network resources among users. Client01 received bandwidth values ranging from 0.927 Mbps to 1.07 Mbps, while Client02 maintained bandwidth between 0.984 Mbps and 0.997 Mbps. Similarly, Client03 received bandwidth values ranging from 0.973 Mbps to 0.996 Mbps. The relatively small variation between the minimum and maximum bandwidth values indicates that HTB successfully maintained stable bandwidth allocation throughout the testing period. This consistency ensures that each client receives a fair share of the available network capacity while preventing excessive fluctuations caused by traffic contention. Overall, the graph demonstrates that HTB provides superior bandwidth stability and more predictable Quality of Service compared to the FIFO approach in a multi-client network environment.

Graph 3: Bandwidth variation for two clients using the HTB method with two active clients.

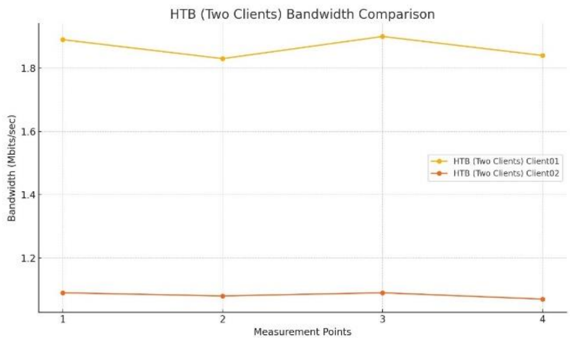

Figure 9. Graph 3: HTB 2 Client Comparison.

The graph illustrates the bandwidth allocation achieved using the Hierarchical Token Bucket (HTB) method when only two clients, Client01 and Client02, were active on the network. The results show that Client01 received bandwidth ranging from 1.83 Mbps to 1.90 Mbps, while Client02 maintained bandwidth between 1.08 Mbps and 1.09 Mbps. In this testing scenario, Client03 was inactive, leaving approximately 1 Mbps of available bandwidth unused by the predefined allocation. HTB dynamically redistributed this remaining bandwidth according to the configured priority levels assigned to each client. Since Client01 was configured with a higher priority than Client02, it received a larger share of the available excess bandwidth. Despite the redistribution process, both clients experienced highly stable and nearly constant bandwidth values throughout the test period. The minimal variation observed demonstrates the effectiveness of HTB in maintaining predictable traffic management while maximizing bandwidth utilization. These findings indicate that HTB not only ensures fair resource allocation but also efficiently distributes unused bandwidth based on priority policies, resulting in the best overall performance and network stability among all tested scenarios.

6. Conclusion

This study investigated the implementation of Quality of Service (QoS) using the Hierarchical Token Bucket (HTB) algorithm on a MikroTik-based network environment and evaluated its effectiveness through several bandwidth management scenarios. The experimental results demonstrated that HTB provides a more efficient and reliable approach to bandwidth allocation compared to the First In First Out (FIFO) method. While FIFO exhibited significant bandwidth fluctuations among active clients, HTB maintained a more balanced and consistent distribution of network resources, ensuring that each client received bandwidth according to the configured allocation policies. This capability contributed to improved network stability and a more predictable user experience, particularly in multi-client environments where bandwidth contention frequently occurs.

The findings also revealed that network performance is strongly influenced by factors such as the number of active clients, traffic demand, and bandwidth prioritization policies. HTB effectively managed these factors through its hierarchical bandwidth allocation mechanism, allowing administrators to define minimum guaranteed bandwidth, maximum bandwidth limits, and client priorities. When all three clients were active, HTB ensured fair bandwidth distribution and significantly reduced the fluctuations observed under FIFO. Furthermore, when only two clients were active, HTB dynamically redistributed the unused bandwidth according to the configured priority levels, resulting in more efficient utilization of available network resources. This adaptive behavior highlights the flexibility of HTB in handling changing network conditions while maintaining service quality.

Overall, the implementation of HTB successfully improved bandwidth management, enhanced network performance, and provided greater stability in comparison with conventional queue management methods. The results indicate that HTB is a suitable solution for networks requiring controlled bandwidth allocation, fair resource distribution, and consistent Quality of Service. By ensuring stable connectivity and optimizing bandwidth utilization, HTB can support better application performance and improve user satisfaction in environments with varying traffic demands and multiple concurrent users.

References

[1] J. D. Santoso, "Analisis Perbandingan Metode Queue Pada Mikrotik," *Pseudocode*, vol. 7, no. 1, pp. 1–7, 2020, doi: 10.33369/pseudocode.7.1.1-7.

[2] H. Kurniawan, "Pembangunan Jaringan RT-RW Net Berbasis Mikrotik di Desa Sukodono Kecamatan Dampit Kabupaten Malang," *J. Artik.*, vol. 9, no. 1, p. 9, 2018.

[3] T. O. Sidqi, I. Fitri, and N. D. Nathasia, "Implementasi Manajemen Bandwidth Menggunakan Metode HTB (Hierarchical Token Bucket) Pada Jaringan Mikrotik," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 6, no. 1, pp. 132–138, 2021, doi: 10.29100/jipi.v6i1.1927.

[4] B. Widodo, "Manajemen Trafik Menggunakan HTB untuk Meningkatkan Kualitas Layanan IP Network," *J. Sains Terap.*, vol. 11,

no. 1, pp. 88–103, 2021, doi: 10.29244/jstsv.11.1.88-103.

[5] G. Feoh and P. W. Gunawan, "Implementation of Bandwidth Management with Hierarchical Token Bucket (HTB), Per Connection Queue (PCQ) and Layer 7 Protocol Methods Using a Mikrotik Router," vol. 1, no. 1, pp. 65–84, 2022.

[6] H. Alamsyah and S. Somantri, "Perancangan dan Implementasi QoS di Mikrotik Menggunakan Metode HTB (Studi Kasus SMP MBS Al Karimah Cibadak)," *J. Tek. Inform. UNIKA St. Thomas*, vol. 7, pp. 14–22, 2022, doi: 10.54367/jtiust.v7i1.1648.

[7] A. Ramadhan, D. I. Mulyana, and Y. Akbar, "Implementasi Manajemen Jaringan MikroTik Layanan Port API Dengan Algoritma Hierarchial Token Bucket," *Progresif J. Ilm. Komput.*, vol. 19, no. 1, pp. 425–435, 2023.

[8] P. Haryani and S. Raharjo, "Manajemen Pada Jaringan Mikrotik Menggunakan Metode Hierarchical Token Bucket (HTB) dan Keamanan Firewall Intrusion Detection System (IDS)," *J. Jarkom*, vol. 9, no. 1, pp. 1–9, 2021.

[9] B. Kitchenham and S. Charters, "Guidelines for Performing Systematic Literature Reviews in Software Engineering," United Kingdom, 2007.

[10] W. A. Hidayatulloh and H. Setiawan, "Implementation of the RT/RW Net Network Using the IP Bindings and Hierarchical Token Bucket (HTB) Methods for Micro, Small, and Medium Enterprises," *J. Inf. Technol. Comput. Sci.*, pp. 1–9, 2023.

[11] T. Rahman, S. Sumarna, and H. Nurdin, "Analisis Performa RouterOS MikroTik pada Jaringan Internet," *INOVTEK Polbeng Seri Inform.*, vol. 5, no. 1, pp. 178–187, 2020, doi: 10.35314/isi.v5i1.1308.

[12] D. A. Nur Soleh and A. T. Arsanto, "Analisa QoS (Quality of Service) Menggunakan Simple Queue dan Queue Tree Menggunakan Mikrotik," *J. Explor. IT*, vol. 11, no. 2, pp. 38–44, 2019.

[13] S. Budin and I. Riadi, "Traffic Shaping Menggunakan Metode HTB (Hierarchical Token Bucket) pada Jaringan Nirkabel," *Bul. Ilm. Sarj. Tek. Elektro*, vol. 1, no. 3, pp. 144–151, 2019, doi: 10.12928/biste.v1i3.1100.

[14] D. Danang and K. Setiawan, "Pengaturan Billing Hotspot Pada Sistem Jaringan RT/RW Net Dengan Mikrotik Router OS," *J. Publ. Tek. Inform.*, vol. 1, no. 1, pp. 12–22, 2022.

[15] P. Pangestu and Desmira, "Analisis Optimalisasi Kinerja Jaringan MAN pada Layanan Internet Berbasis Mikrotik di PT. Bina Technindo Solution," *J. PROSISKO*, vol. 8, no. 1, pp. 8–17, 2021.

[16] J. Ruskin, "Quality of Service and High Availability," in *Systems and Network Infrastructure Integration*, 2020, pp. 113–156. doi: 10.1002/9781119779964.ch8.

[17] O. Reboul, "Simulating Network Architectures with GNS3," in *Systems and Network Infrastructure Integration*, 2020, pp. 9–25. doi: 10.1002/9781119779964.ch2.