INTERNATIONAL JOURNAL OF SOCIAL SCIENCE, EDUCATIONAL, ECONOMICS, AGRICULTURE RESEARCH AND TECHNOLOGY (IJSET) | ISSN (E): 2827-766X
Website: https://ijset.org/ | Volume 5 No. 7 (2026)

IMPLEMENTATION OF LOAD BALANCING ON WIFI CORNER FOR RT/RW NET SERVER REDUNDANCY

Nurhidayat[1]*, Amuharnis[2]
[1,2]Informatics, University of Metamedia, Indonesia
E-mail: nurhidayat@metamedia.ac.id, amuharnis@metamedia.ac.id

Received: 02/06/2026 | **Revised** : 11/06/2026 | **Accepted:** 23/06/2026 | **Published** :25/07/2026

Abstract

As community-based networks increasingly rely on the internet, providing stable, reliable, and uninterrupted service has become essential. However, public internet hubs like "WiFi Corner" often suffer from network overloads and frequent disruptions due to the limited capacity of local RT/RW Net servers. This bottleneck degrades service quality, spikes latency, and increases the risk of downtime, ultimately disrupting user activities. To address this issue, this study implements a load-balancing technique at WiFi Corner to improve server redundancy and ensure continuous uptime. We utilized the Per Connection Classifier (PCC) method on MikroTik RouterOS, which dynamically distributes traffic based on active connections while providing an automatic failover mechanism if an ISP link drops. To evaluate performance, we compared three distinct scenarios: A single ISP setup, Dual ISPs without load balancing, and Dual ISPs optimized with PCC load balancing and failover. The performance metrics analyzed include throughput, delay, packet loss, jitter, and overall system uptime. The results demonstrate that implementing PCC load balancing successfully doubled the throughput compared to the baseline setup. Additionally, delay and jitter were significantly reduced, and packet loss dropped below 2%. Thanks to the integrated failover mechanism, network uptime reached nearly 100%. Ultimately, this study offers a cost-effective and highly efficient technical solution to boost the reliability and sustainability of community-based RT/RW Net internet services.

Keywords: Load Balancing, MikroTik RouterOS, PCC, RT/RW Net, WiFi Corner.

INTRODUCTION

The rapid evolution of information and communication technology has made fast, stable, and reliable internet access a necessity. In Indonesia, the RT/RW Net model has emerged as a popular community-based solution, allowing neighbors to pool resources and share internet access at an affordable cost. However, as more residents log on, these networks frequently suffer from severe traffic congestion, drastically reducing the overall quality of service. A textbook issue in RT/RW Net setups especially within "WiFi Corner" deployments—is the bottleneck caused by an uneven distribution of network traffic. This imbalance triggers high latency, spikes packet loss, and chokes overall throughput. To fix this, community networks need a mechanism that does more than just share the load proportionally; they need a system with built-in redundancy to keep users online if a primary connection goes down. MikroTik RouterOS offers a practical remedy through its Per Connection Classifier (PCC) load-balancing feature. By distributing traffic based on individual network connections, PCC balances workloads effectively while serving as a safety net to keep the network running if one ISP link fails.

This issue is highly evident in Parak Rumbio Village, Padang City, West Sumatra Province, where community internet usage has surged. Today, the internet is an essential tool for local students—from elementary school to university who rely on it for online learning, homework research, and accessing academic papers. Currently, there are seven WiFi Corner access points scattered across residents' homes, each serving about five active users a day, for a total of roughly 35 daily users. The dominant activities are high-bandwidth tasks like attending online lectures via Zoom and Google Meet, or streaming educational content on YouTube. Unfortunately, all seven access points are forced to share a meager 25 Mbps bandwidth pool, which quickly chokes during peak hours when everyone tries to get online at once.

As data demands climb, the central RT/RW Net server managing this traffic frequently overloads, causing slow speeds and frustrating network downtime. This leaves service providers in a tough spot: they need to optimize server performance without buying more bandwidth, especially since local economic constraints mean raising monthly user fees is out of the question.

Addressing this bottleneck is the core focus of this study. Specifically, this research aims to optimize a load-balancing system within the WiFi Corner network to improve overall performance, ensure the system can scale as user numbers grow, and reinforce server redundancy to prevent unexpected service drops. Ultimately, the goal is to develop a load-balancing approach that streamlines traffic across access points, keeps the network responsive and stable during peak usage, and provides a seamless failover mechanism to guarantee continuous internet access.

Technically speaking, load balancing works by spreading bandwidth demand across two or more connection paths. Its primary goal is to maximize network performance while preventing any single path from becoming a bottleneck. Among the various techniques available for networking hardware, the Per Connection Classifier (PCC) method stands out as one of the most widely adopted. Implemented directly within MikroTik RouterOS, PCC spreads out network traffic on a per-connection basis across multiple Internet Service Provider (ISP) links.

The underlying mechanism of PCC involves sorting traffic using specific parameters, such as the source address, destination address, and port numbers. By doing this, it ensures that a client's specific connection consistently travels down the exact same path. This behavior maintains connection stability, which is incredibly important for apps that are highly sensitive to sudden route changes—such as online banking, gaming, VPNs, and Voice over IP (VoIP) services.

LITERATURE REVIEW

RT/RW Net Network

An RT/RW Net is a community-driven internet distribution model designed to make web access affordable by allowing neighbors to pool and share resources. This setup has become incredibly popular in residential neighborhoods, particularly in areas where major internet infrastructure is either lacking or too expensive. However, because these networks rely on shared infrastructure, small bandwidth pools and a growing number of simultaneous users often lead to severe bottlenecks—especially during peak evening hours when everyone logs on at once.

The introduction of "WiFi Corner" hubs within these RT/RW Net setups has been a game-changer for local communities, making it much easier for residents to get online for schoolwork, daily communication, and entertainment. That said, as the community's reliance on the internet deepens, simply setting up access points is no longer enough; it now requires smart, proactive network management strategies to keep the service stable and prevent the system from crashing under heavy loads.

Load Balancing

At its core, load balancing is a technique used to spread network traffic evenly across multiple internet connections or servers. By distributing the data load, the system maximizes resource utilization and prevents any single connection from becoming a frustrating bottleneck. Ultimately, implementing a solid load-balancing strategy is one of the most effective ways to boost network performance—driving up throughput, dropping latency, and ensuring services stay online and accessible.

Network administrators have several methods at their disposal to achieve this, including Equal Cost Multi-Path (ECMP), NTH, and Per Connection Classifier (PCC). Among these, PCC has become a go-to choice for MikroTik RouterOS environments. Unlike packet-based routing methods that can cause connection instability, PCC sorts traffic based on specific connection streams, offering end-users a much smoother and more reliable browsing experience [6]. This aligns with a growing body of research showing that strategic load balancing drastically improves a network's Quality of Service (QoS), particularly in multi-WAN setups facing heavy data demands.

Per Connection Classifier (PCC)

The Per Connection Classifier (PCC) is a specialized load-balancing method built into MikroTik RouterOS that categorizes network traffic using specific connection traits, such as source and destination IP addresses or port numbers. In practice, the PCC mechanism groups client connections into distinct traffic streams and maps each stream to a dedicated gateway. Because of this, a client's specific connection consistently relies on the exact same gateway, which keeps network sessions highly stable.

This consistency is where PCC truly shines compared to conventional routing methods. By keeping sessions persistent, it prevents disruptions in applications that are notoriously sensitive to sudden path changes—such as online banking, video conferencing, VPNs, online gaming, and Voice over IP (VoIP) services. This reliability is backed by a solid track record in networking research, where studies consistently show that PCC-based load balancing effectively slashes packet loss and latency while maximizing throughput across MikroTik infrastructures.

Failover and Redundancy

Failover is essentially a safety net that automatically shifts network traffic from a dropped connection to an active backup link, ensuring that users stay online without interruption. In community setups like an RT/RW Net, a reliable failover mechanism is a must-have; when an internet link goes down, it doesn't just affect one person—it disconnects an entire neighborhood at once. This brings us to redundancy, which is the practice of keeping backup systems or secondary connections on standby so the network can absorb hardware or ISP failures without breaking a sweat.

When you pair load balancing with an automatic failover mechanism, you get the best of both worlds: optimized daily speeds and an incredibly resilient network. Research consistently shows that combining failover with PCC load balancing yields far better network stability than traditional, single-gateway architectures. By taking human intervention out of the equation, automatic failover cuts recovery times down to seconds, keeping network hiccups practically invisible to the end user.

Quality of Service (QoS)

Quality of Service (QoS) is essentially a network's ability to prioritize specific types of traffic and deliver a smoother user experience. It achieves this through smart bandwidth allocation, latency management, traffic prioritizing, and minimizing packet drops.

To judge whether a network is actually performing well, engineers rely on four key QoS metrics: throughput, latency, packet loss, and jitter.

Throughput is the actual, real-world speed of successful data delivery across the network.

Latency measures the frustrating lag or delay it takes for a data packet to travel from the sender to the receiver.

Packet loss calculates the percentage of data packets that vanish entirely during transit.

According to the Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) standard, keeping latency and packet loss as low as possible is the gold standard for a healthy network. This is precisely why a thorough QoS analysis is so vital—it gives us concrete data to prove whether implementing PCC load balancing actually fixes the bottlenecks in an RT/RW Net or just masks the symptoms.

METHOD

This study utilizes an experimental approach paired with a comparative Quality of Service (QoS) analysis. Rather than being purely theoretical, this research is development-oriented and implementation-based, focusing on the real-world deployment of a WiFi Corner architecture and an RT/RW Net server system equipped with load balancing and failover capabilities. The entire deployment framework is built upon established networking theories and insights from previous literature on network optimization, workload distribution, and system redundancy.

An experimental method was chosen because this project required hands-on configuration, live implementation, and direct performance testing within an active, real-world RT/RW Net environment. To measure success, a comparative QoS analysis was used to benchmark network performance data, directly contrasting the baseline metrics against the results gathered after deploying the Per Connection Classifier (PCC) method on MikroTik RouterOS.

The empirical focus of this research is a case study on the WiFi Corner network in Parak Rumbio Village, Padang City, West Sumatra Province. By focusing on this specific location, the study aims to provide a concrete evaluation of how effectively PCC load balancing can rescue a strained RT/RW Net infrastructure—specifically by boosting network reliability, handling scalability as user numbers grow, and reinforcing server redundancy.

Research Stages

To ensure a completely objective and measurable evaluation of the PCC load-balancing method on MikroTik RouterOS, the research was structured into a series of systematic phases. The complete workflow and breakdown of these research stages are illustrated in Figure 1.

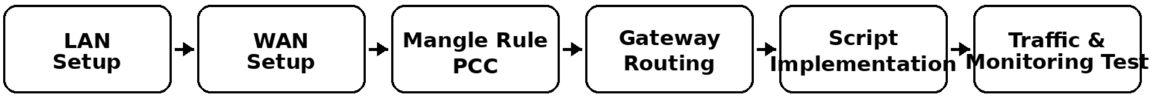

Figure 1. Research Stages

The research process followed a sequential framework to ensure a rigorous and structured implementation.

Phase 1: Literature Review & Theoretical Foundation: The study began with a comprehensive literature review to analyze previous work on load balancing, failover systems, RT/RW Net optimization, and Quality of Service (QoS). This phase was critical for identifying gaps in existing research and building a solid theoretical foundation for the project.

Phase 2: Topology and Interface Configuration: The second stage involved setting up the WAN and LAN interfaces to establish the underlying network topology. This architecture linked multiple internet gateways directly to the MikroTik RouterOS server, which then distributed the connection out to the various local WiFi Corner access points.

Phase 3: Firewall Mangle Rules & PCC Implementation: Next, firewall mangle rules were configured using the Per Connection Classifier (PCC) method. At this stage, incoming network traffic was sorted using specific connection traits—namely source and destination IP addresses alongside source and destination ports. Once classified, the traffic was balanced proportionally across the available internet gateways.

Phase 4: Routing and Session Persistence: Following the PCC setup, routing rules were defined for each gateway to guarantee that a client's specific connection would consistently travel down the same path. This session persistence is vital for preventing drops or hiccups in real-time applications like video conferencing, online learning platforms, and streaming services.

Phase 5: Automatic Failover Deployment: An automatic failover mechanism was then integrated to safeguard service continuity. If a primary gateway goes offline or experiences a disruption, the system instantly shifts traffic to an active backup link, making the transition virtually seamless for the end user.

Phase 6: Traffic Testing, Monitoring, and QoS Analysis: Once the network was fully deployed, live traffic testing and monitoring were conducted to evaluate performance under real-world conditions. Network behavior was analyzed during both normal and peak hours to see how well the system handled congestion. The evaluation relied on four core Quality of Service (QoS) parameters:

- **Throughput:** To measure data transmission speeds.
- **Latency:** To check for transmission delays.
- **Packet Loss:** To track data drops.
- **Server Uptime/Downtime:** To assess overall system reliability.

Ultimately, the gathered data underwent a comparative QoS analysis, directly contrasting the old network's performance against the new PCC-enabled infrastructure. These final results provided concrete proof of the system's ability to improve overall network performance, scalability, and redundancy.

Data Collection Method

Data collection was conducted through several stages. First, the existing conditions of the WiFi Corner network in Parak Rumbio Village were identified, including: the number of WiFi Corner access points, the number of active daily users, internet bandwidth capacity, and the existing network topology.

Second, documentation of the existing network configuration prior to the implementation of load balancing was collected as baseline data for comparative analysis.

Third, quantitative data were obtained by measuring network performance parameters before the implementation of PCC load balancing. The measured parameters included throughput, latency, and uptime performance. These measurements were carried out using network monitoring and traffic analysis tools integrated with MikroTik RouterOS. The measurement results before PCC implementation are presented in Table 1.

Table 1. Network Performance Measurement Results Before PCC Implementation

Internet Service Provider	Parameter	Value	Ideal Standard	Description
ISP Client	Throughput	12.92 Mbps	≥ 20 Mbps	Low bandwidth utilization and not optimal for multi-user environments
	Latency	28 ms	≤ 30 ms	Still within tolerance limits but increased significantly during peak hours
	Uptime	89%	≥ 95%	Frequent disconnections with downtime of approximately 2–3 hours per week
ISP Personal	Throughput	7.88 Mbps	≥ 20 Mbps	Download speed only reached 31% of total available bandwidth
	Latency	17 ms	≤ 30 ms	Network response was relatively good but unstable during high traffic loads
	Uptime	90%	≥ 95%	Service stability was not optimal due to periodic connectivity interruptions

Load Balancing Technique Using PCC

The Per Connection Classifier (PCC) method serves as the core mechanism for distributing network traffic proportionally across multiple internet gateways. Rather than routing data blindly, PCC analyzes specific connection parameters such as the source IP, destination IP, source port, destination port, or a combination of these fields to categorize incoming traffic.

At its heart, PCC relies on a hashing mechanism driven by these connection parameters. By hashing the data, the router ensures that all packets belonging to a specific client session are consistently directed through the exact same gateway. This consistency is crucial; it prevents "route hopping" during active sessions, which would otherwise break secure connections or disrupt live applications. Ultimately, this approach achieves two goals at once: it maintains a balanced workload across all available gateways while seamlessly laying the groundwork for automatic failover if a primary link goes offline.

RESULTS AND DISCUSSION

This section details the overall system design and practical deployment of the PCC-based load-balancing architecture for the RT/RW Net WiFi Corner network. The framework is divided into two core parts: the physical hardware architecture and the underlying software configuration flow. On the implementation front, the process involves configuring the network hardware and deploying the load balancing and failover mechanisms directly within MikroTik RouterOS. The proposed topology for this PCC load-balancing setup is illustrated in Figure 2.

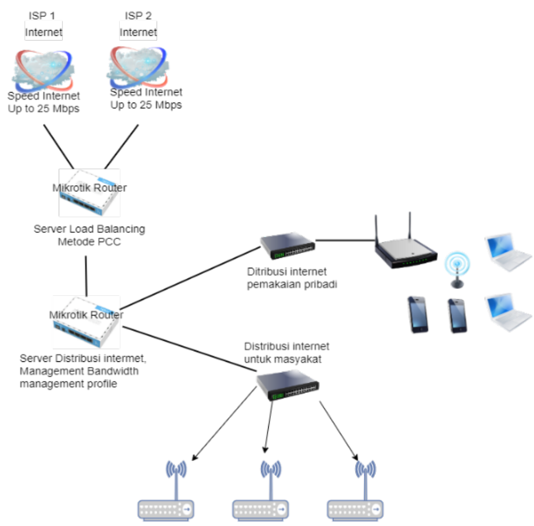

Figure 2. PCC Load Balancing Topology

As illustrated in the topology diagram in Figure 2, the network architecture is built around two Internet Service Providers (ISPs), a centralized MikroTik RouterOS device, and four downstream WiFi access points. In this setup, ISP-1 and ISP-2 serve as the dual internet entry points, feeding directly into the MikroTik RouterOS. This main router acts as the brain of the network, running the Per Connection Classifier (PCC) method to dynamically balance incoming traffic while managing the automatic failover system. From there, the optimized traffic is distributed out to the various WiFi Corner access points, delivering a stable and reliable connection to the end users.

Network Architecture Design

The PCC load-balancing architecture was specifically tailored to reinforce the RT/RW Net infrastructure powering the WiFi Corner network. To bring this design to life, the deployment was broken down into a series of precise configuration phases, beginning with the interface setup.

Ethernet Configuration

```
/interface ethernet set [find name="ether1"] name=ether1-isp-ke-1
/interface ethernet set [find name="ether2"] name=ether2-isp2-ke-2
```

Breaking down this syntax:

1. /interface ethernet directs the router to the specific Ethernet configuration menu.
2. The set parameter instructs the system to modify an existing interface's properties.
3. The conditional argument [find name="ether2"] targets the default port labeled ether2.
4. Finally, name=ether2-isp2-ke-2 applies the new, descriptive identifier to that port.

This renaming process ensures that the administrator can immediately distinguish between different ISP links during the later stages of the configuration. The finalized Ethernet interface layout within MikroTik RouterOS is illustrated in Figure 3.

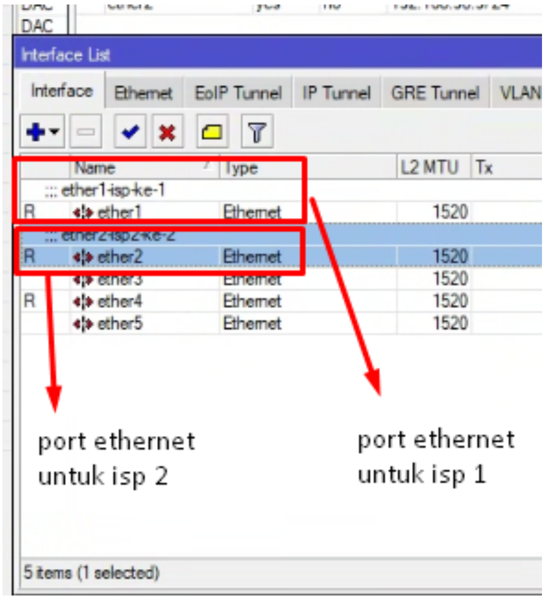

Figure 3. Ethernet Interface Configuration

DHCP Client Configuration

This step configures the MikroTik router to automatically lease IP addresses from both Internet Service Providers (ISPs). By enabling the DHCP Client feature on the ether1-isp-ke-1 and ether2-isp2-ke-2 interfaces, the router dynamically fetches its network settings without requiring manual IP assignments for the WAN side.

Crucially, the command includes the parameter add-default-route=no. Disabling the automatic default route is a deliberate choice here; it prevents the router from generating its own fallback paths, which would conflict with our custom routing scripts. By setting this to "no," we ensure that traffic distribution remains entirely under the control of our manual load-balancing and failover rules. The finalized DHCP Client setup in MikroTik RouterOS is illustrated in Figure 4

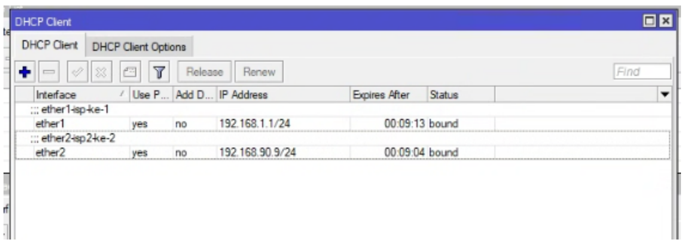

Figure 4. DHCP Client Configuration

DNS Server Configuration

To handle domain name resolution, the DNS configuration on the MikroTik router was set up by enabling its built-in DNS resolver feature. This allows the router to act as a local DNS proxy, intercepting and forwarding domain requests from local clients to speed up web browsing.

For this deployment, Google's Public DNS servers (8.8.8.8 and 8.8.4.4) were configured as the upstream forwarders. Utilizing these global servers ensures fast, stable, and highly reliable domain translation for all connected users on the network. The complete DNS Server setup within MikroTik RouterOS is displayed in Figure 5.

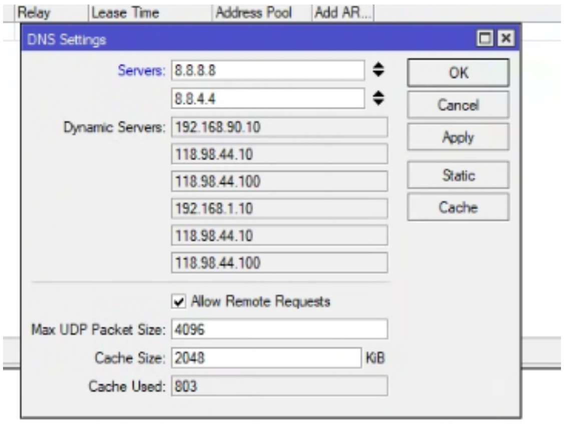

Figure 5. DNS Server Configuration

NAT (Masquerade) Configuration

Network Address Translation (NAT) was deployed using the masquerade method. This configuration is essential for translating local, private IP addresses from the WiFi Corner clients into routable public IP addresses before the packets are pushed out across the internet.

Crucially, the rule includes the parameter place-before=*0. This ensures that this specific NAT rule is pushed to the absolute top of the srcnat chain, giving it maximum processing priority. By placing it at index zero, the router guarantees that address translation occurs immediately before any secondary NAT policies can interfere. The complete firewall NAT configuration within MikroTik RouterOS is illustrated in Figure 6.

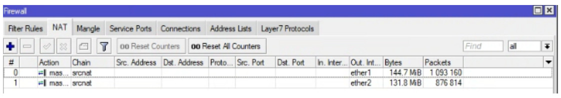

Figure 6. NAT Configuration

Address List Configuration for Local Networks

Before moving forward, it is important to check that all local networks are accounted for in the router's database. If any internal subnet is missing, it must be added manually to the network group.

In this setup, an address list labeled LOCAL is used to define all private IP address ranges. This list acts as a central reference point, allowing the router to easily identify exactly which traffic is originating from the internal network before applying load-balancing or firewall rules.

ISP Routing and Gateway Configuration

Routing rules were defined to ensure that traffic is explicitly steered toward its designated ISP gateway. To prevent network dropouts, a recursive failover mechanism was implemented using the stable public DNS addresses 1.0.0.1 and 1.0.0.2. This setup acts as a continuous ping test, allowing the router to actively monitor gateway availability beyond just the physical link status. Additionally, custom routing marks—specifically via-ISP1 and via-ISP2—were applied to differentiate the paths between the two ISPs. These marks are the crucial link that ties our routing table to the PCC-based load balancer, ensuring that traffic is split accurately and predictably. The finalized static routing configuration within MikroTik RouterOS is shown in Figure 7.

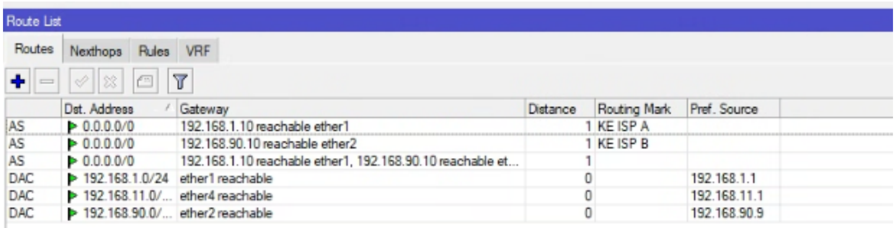

Figure 7. Routing Configuration

PCC Load Balancing Configuration

The following script details the MikroTik RouterOS configuration used to deploy the core PCC load balancer. This setup utilizes firewall mangle rules to handle connection and route marking, creating the operational foundation for our traffic distribution system.

Internal Traffic Exemption: Initially, a set of rules was established to accept all traffic moving purely between internal local IP addresses. By applying these rules across the primary chains—including prerouting, postrouting, forward, input, and output—the router ensures that local, internal communication bypasses the marking engine entirely, preventing unnecessary processing overhead and internal lag.

Inbound Connection Marking: Next, connection marking (mark-connection) was configured based on the ingress WAN interface. Traffic entering through ether1-isp-ke-1 is tagged as via-ether1-isp-ke-1, while incoming packets on ether2-isp2-ke-2 receive the tag via-ether2-isp2-ke-2.

Route Marking and PCC Classification: These connection marks directly feed into the route marking (mark-routing) process, mapping traffic to either the via-ISP1 or via-ISP2 routing tables. To actually split the workload, the Per Connection Classifier (PCC) is activated using the parameters both-addresses-and-ports:2/0 and both-addresses-and-ports:2/1.

By analyzing both the source/destination IPs and ports, this mechanism mathematically divides outbound traffic from local clients and spreads it proportionally across the two ISP gateways. The beauty of this approach is its dual functionality: it maintains a perfectly balanced traffic load during normal operations, and seamlessly acts as a redundancy safeguard if either ISP drops offline. The final mangle rule architecture within MikroTik RouterOS is displayed in Figure 8.

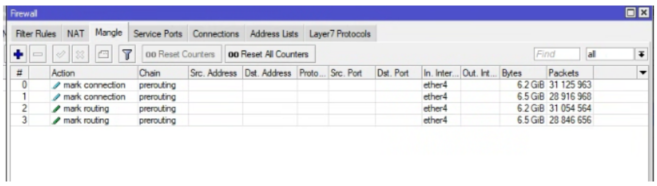

Figure 8. PCC Load Balancing Configuration

Client Connectivity Testing

Client connectivity testing was conducted to verify that the PCC load-balancing implementation on MikroTik RouterOS met its core objectives: distributing the network workload evenly while keeping user sessions perfectly stable. The testing phase systematically evaluated how the system handled live user traffic under varying network loads.

Client Connectivity Test Results

The empirical results demonstrated a clear improvement in network behavior; client connectivity remained highly stable even as the volume of active users scaled up.

With the PCC method active, the network achieved several key performance upgrades:

- **Reduced Latency:** Average delay dropped to approximately 65 ms.
- **Optimized Bandwidth:**
- **Traffic distribution became noticeably more balanced across the WAN links. Minimized Data Drops:** Packet loss was significantly reduced compared to the baseline, single-gateway conditions.

During the live trial, several client devices were connected to the local network to stress-test internet accessibility. Observation of the live traffic logs confirmed that client connections were successfully and dynamically routed across the different ISP gateways in strict accordance with our PCC hashing rules. This proof of alternating gateway routing is illustrated in Figure 10.

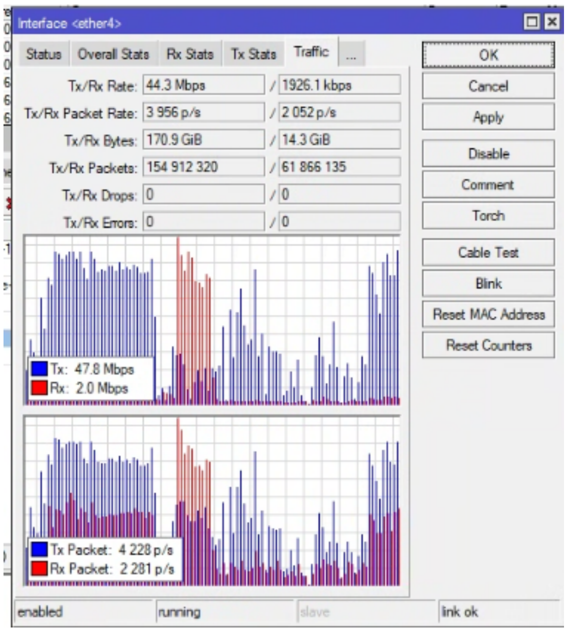

Figure 10. Internet Connection Speed Test Results

Traffic Distribution Testing

Traffic distribution testing was carried out to verify that the outbound internet workload from multiple active clients was being evenly split across our ISP links, exactly as dictated by the load-balancing algorithm.

To execute the test, we generated simultaneous, high-volume internet traffic from a cluster of client devices connected to the WiFi Corner network. We then tracked the real-time distribution of these data streams using the IP > Firewall > Connections tracking menu within MikroTik RouterOS. The test was deemed a success when the monitoring logs confirmed a balanced and proportionate allocation of connections across both ISP gateways simultaneously, as illustrated in Figure 11.

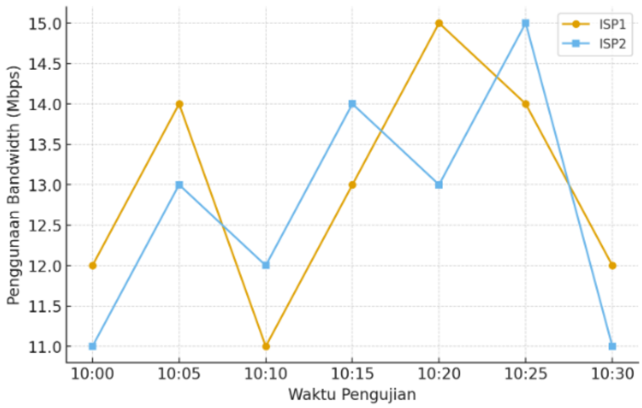

Figure 11. Bandwidth Usage Graph

The test results confirmed that the PCC mechanism effectively balanced the network workload, distributing traffic proportionally between ISP1 and ISP2. By dynamically spreading the data streams, the system prevented either link from bottlenecking, ensuring that neither internet connection path was pushed into an overload state.

Failover Testing

Failover testing was conducted to verify that our MikroTik RouterOS architecture goes beyond simple traffic distribution to actively safeguard service continuity. The primary goal of this phase was to evaluate how effectively the system detects a WAN blackout and reroutes traffic to the surviving ISP path without inflicting noticeable downtime on active users.

To simulate a real-world outage, we intentionally pulled the plug on one of the active ISP links. The system responded instantly: internet connectivity remained fully operational as the traffic was automatically and seamlessly redirected through the secondary gateway. Ultimately, the failover mechanism maintained unbroken client connections with zero manual intervention required by a network administrator—a successful transition captured in the interface logs shown in Figure 12.

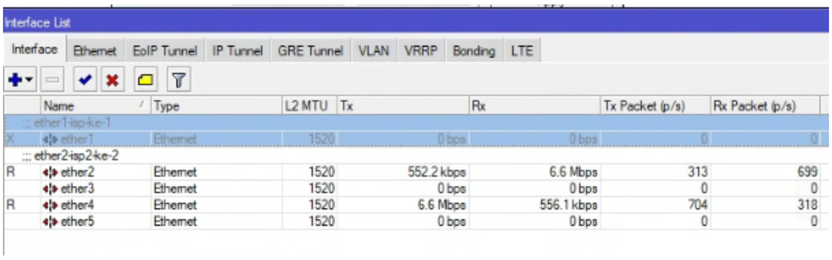

Figure 12. Network Performance Testing Using a Single ISP Path

The empirical results confirmed that the recursive failover configuration on MikroTik RouterOS successfully maintained uninterrupted network availability. By actively monitoring upstream stability, the system significantly enhanced overall service reliability, ensuring a stable and resilient connection environment for the RT/RW Net community.

Monitoring and Evaluation

Following deployment, we systematically monitored and evaluated the infrastructure to quantify how effectively the PCC load-balancing and failover mechanisms improved network performance and uptime. To stress-test the system's redundancy, we intentionally disabled one of the active ISP connections. As expected, the architecture passed the test seamlessly: outbound network traffic was instantly rerouted to the remaining active ISP link, preventing any perceptible disruption or service dropouts for the connected clients.

Overall, the data confirms that implementing PCC on MikroTik RouterOS successfully achieved our core objectives. Traffic loads were split proportionally across both ISP gateways during normal operation, and the recursive failover mechanism successfully insulated the network from standalone ISP outages. The final quantitative metrics captured during these performance tests are aggregated and presented in Table 2.

Table 2. Network Performance Testing Results

Parameter	**ISP 1**	**ISP 2**	**Combined (After PCC)**	**Description**
Throughput	12.92 Mbps	7.88 Mbps	30–55 Mbps (avg. 42.5 Mbps)	Significant bandwidth utilization improvement approaching maximum capacity
Latency	28 ms	17 ms	11 ms	Lower and more stable latency supporting real-time applications
Packet Loss	8.5%	5.6%	1.7%	Significant reduction in packet loss
Delay	205 ms	130 ms	85 ms	Delay decreased substantially
Uptime	89%	90%	99%	Service availability increased significantly with downtime below 5 minutes per week

Analyzing the post-implementation testing results reveals several critical enhancements brought about by the PCC load-balancing architecture. Most notably, network throughput experienced a massive surge, climbing from a baseline of 12.92 Mbps to a robust average of 42.5 Mbps. This jump demonstrates that the system is successfully utilizing the full combined capacity of our ISP links rather than letting bandwidth sit idle.

Beyond raw speed, the user experience saw a dramatic upgrade. Latency and delay spikes were drastically minimized, translating to a much faster, more responsive network environment. This stability is further backed by a substantial reduction in packet loss. By intelligently spreading the traffic workload, the PCC algorithm successfully insulated the network from data bottlenecks, preventing any single gateway from becoming overwhelmed.

Finally, total network uptime soared to nearly 100%. Thanks to the automated failover mechanism, the infrastructure can now survive sudden ISP dropouts by shifting traffic seamlessly behind the scenes, effectively eliminating unexpected downtime for the WiFi Corner community.

The deployment of the PCC load-balancing framework on MikroTik RouterOS successfully resolved the core vulnerabilities of the Parak Rumbio Village WiFi Corner. Prior to this implementation, the network's traffic was bottlenecked through a single ISP link. This single-point-of-failure model caused severe bandwidth congestion, high latency, and frequent service drops during peak hours when users simultaneously accessed online learning, video streaming, and web applications.

By transitioning to the PCC method, traffic is now dynamically and proportionally distributed across multiple ISP gateways. The Quality of Service (QoS) data confirms that this shift eliminated bandwidth hogging and drastically reduced response times. Ultimately, the system successfully transformed a fragile, single-link setup into a resilient and balanced multi-WAN infrastructure capable of handling high concurrent user demands.

CONCLUSION

The implementation of PCC-based load balancing on MikroTik RouterOS successfully optimized the WiFi Corner network, which serves an average of 35 daily active users on a constrained 25 Mbps bandwidth limit. By proportionally distributing traffic and integrating a recursive failover mechanism, the system mitigated server overload risks and established reliable WAN redundancy—ensuring uninterrupted client connectivity even during sudden ISP outages.

Post-implementation metrics confirmed significant upgrades across all QoS parameters, including increased throughput, minimized latency, reduced packet loss, and near-100% network uptime. These results prove that PCC load balancing is an exceptionally viable, scalable solution for multi-user environments.

Crucially, this architecture offers a highly cost-effective blueprint for budget-constrained, community-based networks. It proves that network reliability and service quality can be dramatically improved through smart traffic engineering alone, without requiring expensive bandwidth upgrades or increasing subscription costs for the community.

REFERENCES

[1] D. K. S. a. R. Rahmadani, "Analysis of Network Performance on RT/RW Net Using MikroTik RouterOS," Journal of Network and Computer Applications, vol. 5, no. 2, pp. 45-52, 2022.
[2] M. M. a. A. S. Fitrani, "Configuring Load Balancing and Failover Using a MikroTik Router on RT RW NET," Procedia of Engineering and Life Science, vol. 2, no. 2, pp. 1-8, 2022.
[3] T. B. A. a. A. Hidayat, "Network Performance Optimization Using Load Balancing Techniques in Wireless Networks," International Journal of Computer Networks, vol. 14, no. 1, pp. 33-41, 2023.
[4] R. K. a. A. S. H. Pratama, "Implementation of Server Redundancy Using Load Balancing on MikroTik Networks," Journal of Information Technology and Networking, , vol. 8, no. 3, p. 120–128, 2024.
[5] F. A. N. a. D. P. M. S. Alam, "Failover Mechanism for Reliable Internet Connectivity Using PCC Method,," International Journal of Advanced Networking and Applications, , vol. 15, no. 2, p. 88–96, 2024.
[6] D. M. a. D. Ramayanti, "Load Balancing Techniques for Bandwidth Optimization in Multi-ISP Networks,," Journal of Computer Engineering and Networking,, vol. 11, no. 1, p. 55–63, 2023.
[7] A. N. H. a. W. Sulistyo, "Performance Analysis of PCC Method on MikroTik Router for Multi-WAN Networks," Journal of Telecommunications and Information Technology, vol. 9, no. 4, p. 210–218, 2024.
[8] J. W. a. A. Fathoni, "Implementation of Per Connection Classifier Method on MikroTik RouterOS for Internet Traffic Distribution," Journal of Computer Systems and Informatics, vol. 6, no. 1, p. 75–83, 2025.

[9] D. Mulyana, "Traffic Classification Mechanism in MikroTik PCC Load Balancing," International Journal of Network Security and Engineering, vol. 7, no. 2, p. 98–106, 2023.
[10] A. I. a. A. S. Anton, "Stability Analysis of PCC-Based Load Balancing for Real-Time Applications," Journal of Internet Technology and Smart Systems, vol. 12, no. 1, p. 44–51, 2024.
[11] A. S. T. a. D. Wetherall, Computer Networks, 6th ed, Boston, MA, USA: Pearson,, 2021.
[12] C. Systems, High Availability and Failover Networking Concepts,, USA: Cisco Networking Documentation,, 2022.
[13] A. K. a. S. Gupta, Network Redundancy and Reliability Engineering, New York, NY, USA: Springer, 2021.
[14] R. S. a. Y. Firmansyah, "Analysis of Failover and Load Balancing Performance in Multi-WAN Networks," Journal of Information Systems Engineering,, vol. 10, no. 3, p. 101–110, 2023.
[15] ETSI, elecommunications and Internet Protocol Harmonization Over Networks (TIPHON); General aspects of Quality of Service (QoS), ETSI TR 101 329, 2020.
[16] I.-T. R. G.1010, End-user multimedia QoS categories,, Geneva, Switzerland: International Telecommunication Union,, 2021.
[17] A. Forouzan, Data Communications and Networking, 6th ed., New York, NY, USA: McGraw-Hill, 2022.
[18] W. Stallings, Data and Computer Communications, 11th ed., London, U.K: Pearson, 2021.