

TRYDON SYSTEMS

Master Operational Manual & ISP Infrastructure Guide • Version 2026.1

TRYDON SYSTEMS - MASTER OPERATIONAL MANUAL & ARCHITECTURE GUIDE

Trydon Networks Ltd.

Comprehensive ISP Network Engineering, Dual-WAN PCC Load Balancing, FreeRADIUS AAA, WhatsApp Billing Engine, Hotspot Captive Portal & 24/7 Infrastructure Monitoring Guide

TABLE OF CONTENTS

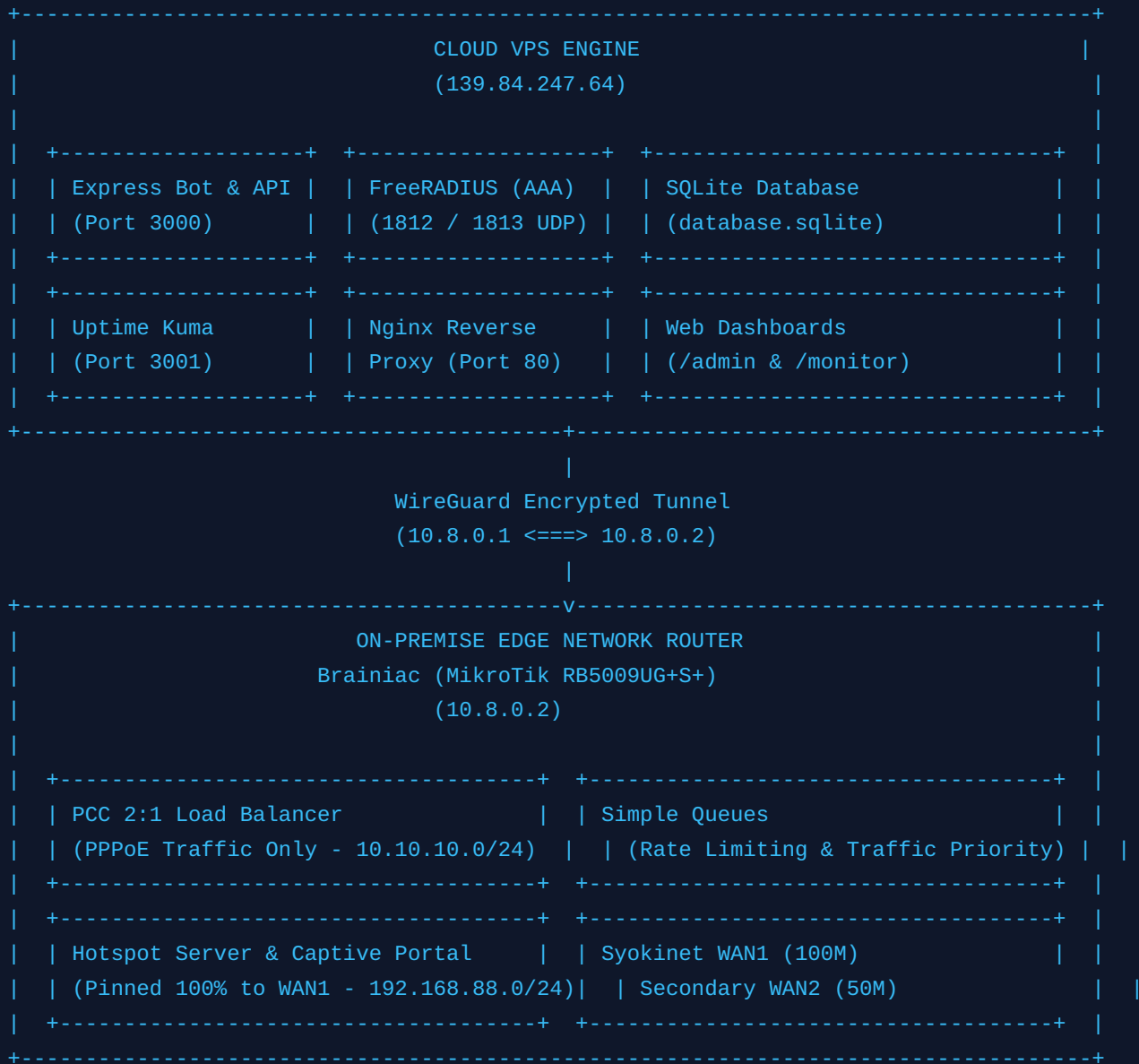
1. [Executive Summary & System Architecture](#1-executive-summary--system-architecture)
2. [Physical & Virtual Infrastructure Layout](#2-physical--virtual-infrastructure-layout)
3. [MikroTik RouterOS Configuration & Dual-WAN PCC](#3-mikrotik-routeros-configuration--dual-wan-pcc)
4. [Bandwidth Profiles, Queue Caps & Priorities](#4-bandwidth-profiles-queue-caps--priorities)
5. [FreeRADIUS AAA Server & Subscriber Sync](#5-freeradius-aaa-server--subscriber-sync)
6. [Trydon WhatsApp Bot & Billing Engine](#6-trydon-whatsapp-bot--billing-engine)
7. [Hotspot Captive Portal & "ALREADY PAID" Logic](#7-hotspot-captive-portal--already-paid-logic)
8. [Admin Dashboard (/admin) & Live Telemetry (/monitor)](#8-admin-dashboard-admin--live-telemetry-monitor)
9. [24/7 Infrastructure Monitoring (Uptime Kuma)](#9-247-infrastructure-monitoring-uptime-kuma)
10. [System Maintenance & Troubleshooting Playbook](#10-system-maintenance--troubleshooting-playbook)
11. [Master Access & Credentials Directory](#11-master-access--credentials-directory)

1. EXECUTIVE SUMMARY & SYSTEM ARCHITECTURE

Trydon Networks operates a hybrid cloud-and-edge ISP infrastructure designed to manage two primary subscriber types:

1. **PPPoE Home Fiber Subscribers:** Fixed-location residential and priority business subscribers authenticated via FreeRADIUS over encrypted PPPoE tunnels.
2. **Hotspot Wi-Fi Users:** Dynamic wireless clients authenticated through a custom MikroTik captive portal integrated with M-Pesa mobile money.

Core System Blueprint



2. PHYSICAL & VIRTUAL INFRASTRUCTURE LAYOUT

A. Cloud VPS Engine (139.84.247.64)

Located in a high-bandwidth data center acting as the central management plane:

- **Operating System:** Ubuntu Linux x86_64
- **Network Binding:** 139.84.247.64 (Public IPv4), 10.8.0.1 (WireGuard Tunnel Gateway)
- **Services Hosted:**

Node.js WhatsApp Billing Bot & Express REST API (<http://127.0.0.1:3000>)

FreeRADIUS RADIUS Server (Ports [1812/udp](#) , [1813/udp](#))

Uptime Kuma Infrastructure Monitor (<http://127.0.0.1:3001> proxied at [/kuma/](#))

Nginx Web Server (Proxying HTTP Port 80 to Node.js and Uptime Kuma)

B. Brainiac Core Router ([10.8.0.2](#) - MikroTik RB5009UG+S+)

Located on-site as the edge distribution router:

- **Hardware:** MikroTik RB5009UG+S+ (ARM64 Quad-Core CPU, 1GB RAM)
- **IP Address:** [10.8.0.2](#) (WireGuard VPN), [192.168.88.1](#) (Hotspot Gateway)
- **WAN Interfaces:**
 - [ether1](#) (WAN1 Syokinet 100M Fiber Modem: [192.168.100.1](#))
 - [ether2](#) (WAN2 Secondary 50M Fiber Modem: [192.168.200.1](#))
- **Tunnel Connection:** Persistent WireGuard tunnel [wg0](#) back to [139.84.247.64:51820](#) .

C. Thinker Secondary Router ([10.8.0.3](#))

Secondary expansion node connected via WireGuard VPN ([10.8.0.3](#)).

3. MIKROTIK ROUTEROS CONFIGURATION & DUAL-WAN PCC

A. Per-Connection Classifier (PCC) Load Balancing

To double network throughput, Brainiac balances **PPPoE subscriber traffic** ([10.10.10.0/24](#)) in a **2:1 ratio** across WAN1 (100 Mbps) and WAN2 (50 Mbps).

```
# Firewall Mangle PCC Rules on Brainiac (RB5009)
/ip firewall mangle

# 1. Accept Local & VPN Traffic (Bypass PCC)
add action=accept chain=prerouting dst-address=10.8.0.0/24 comment="Bypass VPN"
add action=accept chain=prerouting dst-address=192.168.100.0/24 comment="Bypass Modem 1"
add action=accept chain=prerouting dst-address=192.168.200.0/24 comment="Bypass Modem 2"

# 2. Pin Hotspot Traffic Strictly to WAN1
add action=accept chain=prerouting src-address=192.168.88.0/24 comment="Hotspot Pin to WAN1"

# 3. Classify PPPoE Subscribers for 2:1 PCC Load Balancing
add action=mark-connection chain=prerouting src-address=10.10.10.0/24 per-connection-classifier=both-addresses:3/0 new-connection-mark=WAN1_CONN passthrough=yes
add action=mark-connection chain=prerouting src-address=10.10.10.0/24 per-connection-classifier=both-addresses:3/1 new-connection-mark=WAN1_CONN passthrough=yes
add action=mark-connection chain=prerouting src-address=10.10.10.0/24 per-connection-classifier=both-addresses:3/2 new-connection-mark=WAN2_CONN passthrough=yes

# 4. Mark Routing Based on Connection Marks
add action=mark-routing chain=prerouting connection-mark=WAN1_CONN src-address=10.10.10.0/24 new-routing-mark=to_WAN1 passthrough=no
add action=mark-routing chain=prerouting connection-mark=WAN2_CONN src-address=10.10.10.0/24 new-routing-mark=to_WAN2 passthrough=no
```

B. Routing Table Configuration

- **Route Mark** `to_WAN1` : Distance 1 -> Gateway `192.168.100.1` (Syokinet)
- **Route Mark** `to_WAN2` : Distance 1 -> Gateway `192.168.200.1` (Secondary)
- **Default Failover:**

Main Default Route: Distance 1 -> `192.168.100.1`

Secondary Default Route: Distance 2 -> `192.168.200.1`

4. BANDWIDTH PROFILES, QUEUE CAPS & PRIORITIES

Bandwidth profiles are managed dynamically from the `/admin` **Bandwidth Speed tab**. When subscribers connect, FreeRADIUS returns exact rate-limits and queue priorities.

Active Package & Priority Matrix

tbody>

Package Name	Download / Upload Cap	RADIUS Rate-Limit (Mikrotik-Rate-Limit)	Priority	Representative Subscribers
30MBPS	30 Mbps / 30 Mbps	30M/30M	Priority 1 (Highest)	PITA
25MBPS	22 Mbps / 22 Mbps	22M/22M	Priority 2	REHOBOTH, TOT
20MBPS	17 Mbps / 17 Mbps	17M/17M	Priority 3	AIRFORCE
10MBPS	8 Mbps / 8 Mbps	8M/8M	Priority 5 (Standard)	WISECHOISE + 31 Standard Subscribers
Hotspot	4 Mbps / 4 Mbps	4M/4M	Priority 8	Temporary Voucher / Cash Clients

Priority Queue Behavior

During peak congestion hours, MikroTik's queue tree prioritizes packets based on queue priority:

- **Priority 1 (PITA - 30M):** Allocated bandwidth first with minimum queue delay.
- **Priority 2 (REHOBOTH, TOT - 25M):** Served second.
- **Priority 3 (AIRFORCE - 20M):** Served third.
- **Priority 5 (10M Users):** Balanced equally among standard home subscribers.

5. FREERADIUS AAA SERVER & SUBSCRIBER SYNC

FreeRADIUS handles central authentication, authorization, and accounting on the VPS.

A. Core Database Tables (database.sqlite)

- **users** : Stores subscriber credentials, plan, fee, **dueDate** , connection type (**pppoe** or **hotspot**), and **paused** status.

- `packages` : Defines plan names, monthly fees, speed limits, and connection types.
- `radcheck` : Maps subscriber password (`Cleartext-Password :=`).
- `radreply` : Maps subscriber attributes (`Mikrotik-Rate-Limit :=` , `Mikrotik-Group :=`).
- `radacct` : Records active session accounting start/stop times and total uploaded/downloaded bytes.

B. Subscriber Synchronization Workflow (`syncRadiusUser`)

When a user's plan is updated or cash payment recorded in `/admin` :

1. `syncRadiusUser(username)` is called in `utils/db.js` .
 2. Reads subscriber plan from `users` table and matches with `packages` table.
 3. Updates `radcheck` and `radreply` attributes in SQLite.
 4. If subscriber account is expired, `syncRadiusUser` removes their `radreply` entry and issues a PoD (Packet of Disconnect) to kick active sessions on Brainiac.
-

6. TRYDON WHATSAPP BOT & BILLING ENGINE

The Node.js Express application running in `/opt/trydon-whatsapp-bot` handles automated billing, payment webhooks, and WhatsApp customer messaging.

Key Workflows:

- **Automated Expiry Checker:** Runs on startup and periodically. Suspends overdue accounts, notifies subscribers via WhatsApp, and updates RADIUS tables.
 - **M-Pesa / IntaSend Webhook (`/api/payment-webhook`):**
 1. Receives payment notification from IntaSend.
 2. Matches phone number or subscriber account number in `users` table.
 3. Records entry in `payments` table.
 4. Extends subscriber `dueDate` by 30 days.
 5. Calls `syncRadiusUser(username)` to immediately unblock subscriber.
 6. Sends automated WhatsApp payment receipt to subscriber.
-

7. HOTSPOT CAPTIVE PORTAL & "ALREADY PAID" LOGIC

The Hotspot captive portal allows guest clients to purchase internet access or reconnect existing payments via M-Pesa.

A. Landing Page Architecture

- Hosted on Brainiac at `/file/hotspot/login.html`.
- Frontend logic in `/opt/trydon-whatsapp-bot/public/js/main.js` and `scratch/login.html`.
- API Communication: Communicates directly with VPS endpoint `http://139.84.247.64/api/hotspot/reconnect`.

B. "ALREADY PAID" 1-Click Reconnect Algorithm

1. Client connects to Wi-Fi SSID and opens landing page.
2. Clicks **"ALREADY PAID"** and enters M-Pesa receipt code (e.g. `RAB1234567`) or phone number.
3. Landing page sends POST request to `/api/hotspot/reconnect`.
4. Server searches `payments` table for matching M-Pesa code or recent payment within valid timeframe.
5. If payment exists, server extends user account, logs user into MikroTik Hotspot via API (`/ip/hotspot/active/login`), and redirects client to internet.

8. ADMIN DASHBOARD (/admin) & LIVE TELEMETRY (/monitor)

A. `/admin` Administration SPA

Protected single-page administrative dashboard (`https://trydon.net/admin`):

- **Subscriber Management:** Add, edit, extend duration, or delete subscribers.
- **Cash Recharge:** Record manual cash payments and reactivate accounts immediately.
- **Hotspot Hub:** View live online hotspot users, expired accounts, and process maintenance compensation claims.
- **Bandwidth Control:** Edit package speed limits and fee amounts.
- **Payments & Expenses Ledger:** Complete financial revenue and expenditure tracking.

B. `/monitor` Network Telemetry Board

Real-time infrastructure operations board (<https://trydon.net/monitor>):

- **VPS Resource Telemetry:** Live CPU load, RAM usage, disk usage, and server uptime.
 - **MikroTik Router Hardware:** Live RB5009 CPU temperature (`54°C`), RAM usage, board model, RouterOS version.
 - **Bandwidth Gauges:** Live combined WAN Rx/Tx throughput counters.
 - **Latency Telemetry:** Real-time ping latency to Google (`8.8.8.8`), Brainiac (`10.8.0.2`), and Thinker (`10.8.0.3`).
-

9. 24/7 INFRASTRUCTURE MONITORING (UPTIME KUMA)

Uptime Kuma runs on port `3001` (proxied via Nginx at `/kuma/`) and executes 19 continuous automated health checks:

tbody>

#	Monitor Name	Type	Target	Frequency
1	Core Router (Brainiac)	PING	10.8.0.2	Every 20s
2	Secondary Router (Thinker)	PING	10.8.0.3	Every 20s
3	Brainiac API Port	PORT	10.8.0.2:8728	Every 30s
4	Brainiac Winbox Port	PORT	10.8.0.2:8291	Every 30s
5	Brainiac Webfig Interface	PORT	10.8.0.2:80	Every 30s
6	Syokinet Modem Gateway	PING	192.168.100.1	Every 20s
7	Secondary Modem Gateway	PING	192.168.200.1	Every 20s
8	Hotspot Reconnect API Gateway	HTTP	http://127.0.0.1:3000/api/hotspot/reconnect	Every 30s
9	IntaSend M-Pesa Webhook	HTTP	https://trydon.net/api/payment-webhook	Every 30s
10	FreeRADIUS Auth Service	HTTP	https://trydon.net/api/health/radius	Every 20s
11	Trydon SQLite Database Engine	HTTP	https://trydon.net/api/health/db	Every 30s
12	Trydon Net Billing & API	HTTP	https://trydon.net	Every 30s
13	Trydon VPS Health API	HTTP	https://trydon.net/api/admin/vps-health	Every 30s
14	Trydon Express Backend	PORT	127.0.0.1:3000	Every 20s
15	VPS SSH Access Port	PORT	127.0.0.1:22	Every 60s
16	Olympus Fitness Web Site	HTTP	http://olympus.trydon.net	Every 30s
17	Syokinet Primary DNS	PING	8.8.8.8	Every 15s
18	Secondary DNS WAN	PING	1.1.1.1	Every 15s
19	Cloudflare Backup DNS	PING	9.9.9.9	Every 15s

10. SYSTEM MAINTENANCE & TROUBLESHOOTING PLAYBOOK

Playbook 1: Subscriber Paid but Account Still Inactive

1. Open `/admin` dashboard.

2. Go to **Subscribers** tab and search for subscriber.
3. Verify if `DueDate` is in the future.
4. Click **Cash** or edit subscriber plan and save to trigger immediate `syncRadiusUser(username)`.

Playbook 2: Hotspot Client Reconnect Issue

1. Verify client is connected to `Trydon-Hotspot` Wi-Fi.
2. Open captive portal `http://192.168.88.1/login.html`.
3. Enter M-Pesa code under **"ALREADY PAID"**.
4. If M-Pesa code was entered manually via M-Pesa Till/Paybill without account number, open `/admin` -> **Hotspot Hub**, find user, and click **Cash Recharge**.

Playbook 3: Restarting Core Services on VPS

If any web service requires a manual restart, execute via SSH:

```
# Restart WhatsApp Billing Bot Engine
pm2 restart trydon-whatsapp-bot

# Restart Uptime Kuma Monitoring Engine
pm2 restart uptime-kuma

# Reload Nginx Web Server
systemctl reload nginx
```

11. MASTER ACCESS & CREDENTIALS DIRECTORY

tbody>

System Service	URL / Path	Username	Password
Admin Dashboard	https://trydon.net/admin	admin / pita	watersheep
Telemetry Board	https://trydon.net/monitor	admin / pita	watersheep
Uptime Kuma Panel	https://trydon.net/monitor (Uptime Tab)	pita	watersheep69
MikroTik Winbox	10.8.0.2:8291	admin	(Set on router)
VPS Remote SSH	139.84.247.64:22	root	(SSH Key Authentication)

Manual compiled and validated live for Trydon Networks Ltd.