

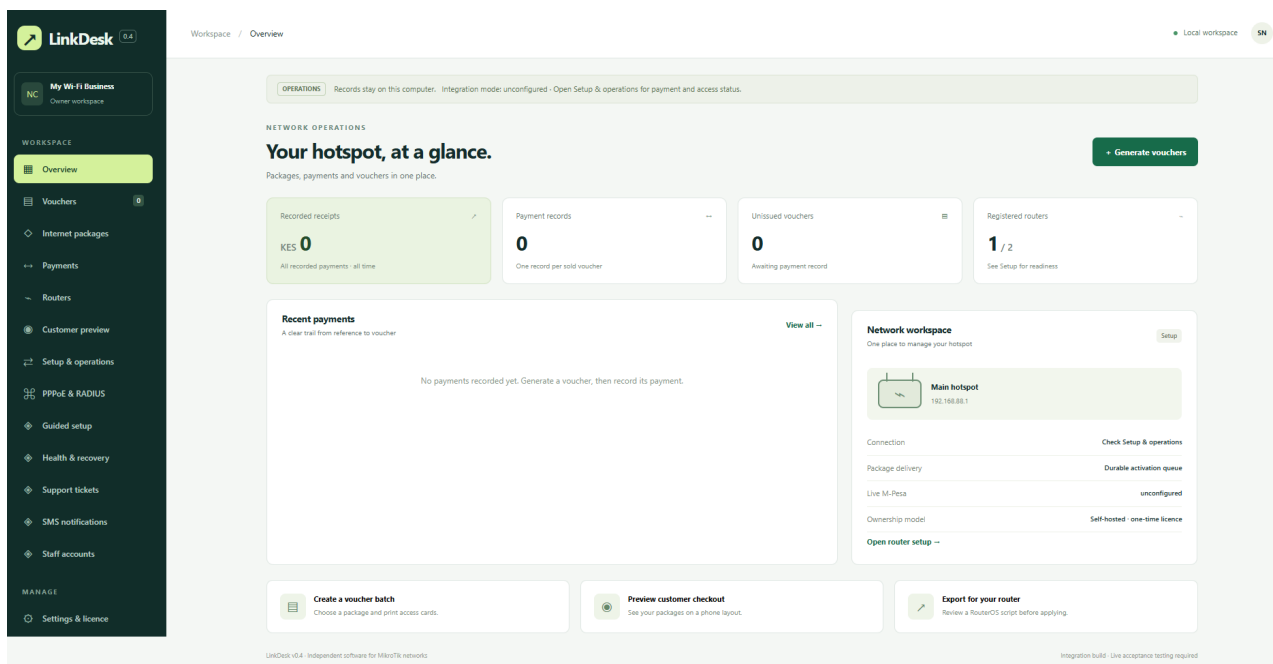
WayDesk

Powered by LinkDesk v0.5.0

Complete user manual

Buying, installation, activation, payments, vouchers, subscribers, diagnostics, backups, staff, SMS and customer support.

Customer and installer edition - 6 October 2026



Actual LinkDesk interface in an initial operations workspace. Empty figures are not evidence of live customer activity. The UI still displays a 0.4 badge in the v0.5 source package.

This replaces the abbreviated v0.4 installation guide for day-to-day reference. Product-key activation supersedes the old licence-JSON paste instructions. It documents included features and their actual prerequisites; it does not claim an unconfigured demo can operate a network.

Website: <https://mikrotikbilling.co.ke/> | Support: <https://wa.me/254722546300>

Find the feature you need

1. Buy and activate the software - page 3
2. Understand demo, preview and operations modes - page 4
3. Dashboard, packages, vouchers and recorded payments - page 5
4. Windows / Linux installation and public checkout - page 7
5. Product-key activation and equipment replacement - page 11
6. Guided setup, diagnostics, backups, plans, portal, SMS and staff - page 12
7. PPPoE, FreeRADIUS, renewals and accounting - page 16
8. Migration and customer cutover - page 20
9. Signed updates and recovery - page 21
10. Acceptance checklist and installation record - page 22

Use PDF search for a screen name or error. The detailed chapters below reproduce the customer-relevant release instructions and add a practical walkthrough. Technical sections apply only to the services you deploy.

1. Buy and activate the software

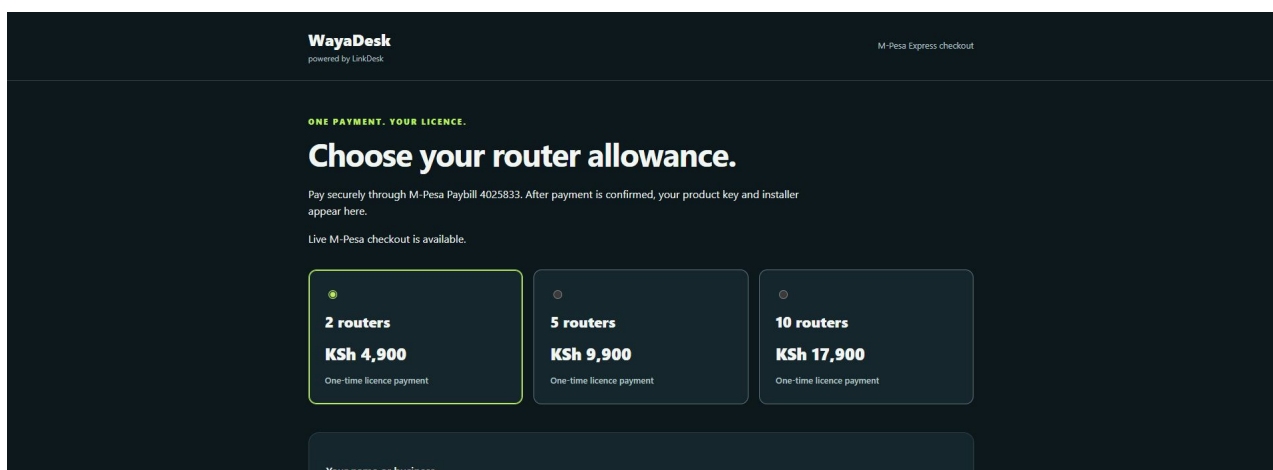
Buy your LinkDesk licence through WayaDesk

1. Open <https://mikrotikbilling.co.ke/checkout> and choose the router allowance that your installation needs. Read the current price on that page.
2. Enter the requested buyer information and your M-Pesa phone number. Check the amount before requesting the STK prompt.
3. Approve the payment on your phone. Enter the PIN only in the phone's M-Pesa prompt. Keep the order page open while payment is verified.
4. After payment is verified, copy your product key and download the seller-prepared customer installer. The private order link can reveal your key; share it only with the buyer.
5. Extract the complete ZIP. Install on Windows or Linux, sign in as owner, then use Settings & licence > Enter product key. Confirm active status and the purchased router allowance.

This website purchase pays for the software licence. Your customers' internet-package payments are a separate integration on your own installation. WayaDesk's seller Paybill is not automatically your business's merchant configuration. Licence checkout currently does not send an email or SMS containing the key.

Keep the two payment journeys separate

Software buyer: WayaDesk checkout > M-Pesa approval > payment verification > product key and installer > installation activation. Internet subscriber: your business checkout > package selection > M-Pesa approval > authenticated payment query > access job > router account > actual browsing. A successful licence purchase does not configure router access, internet packages, subscriber M-Pesa or SMS. Those features need the applicable setup chapters below.



The screenshot shows the WayaDesk checkout interface. At the top, it says 'WayaDesk powered by LinkDesk' and 'M-Pesa Express checkout'. Below this, a green banner reads 'ONE PAYMENT. YOUR LICENCE.' followed by the heading 'Choose your router allowance.' A note states: 'Pay securely through M-Pesa Paybill 4025833. After payment is confirmed, your product key and installer appear here.' Another note says 'Live M-Pesa checkout is available.' There are three selectable options for router allowances, each with a radio button and a 'One-time licence payment' label:

Router Allowance	Price (KSh)
2 routers	4,900
5 routers	9,900
10 routers	17,900

At the bottom, there is a text input field labeled 'Your name or business'.

Public WayaDesk software-licence checkout. Prices are a screenshot snapshot; use the live page for the current price. This is not a subscriber Wi-Fi checkout.

2. If the demo refuses an action

Identify what you opened

Preview.html: offline design preview. It does not make payments, save live records or configure a router.

Developer demo: started with `python server.py --demo --open`. Payments are simulated. Demo vouchers cannot be exported to a real router. Connection checks require an operations workspace; demo mode is not a live deployment. **Customer preview inside operations:** shows the checkout design. Its button does not send STK. Use the public customer checkout address in Setup & operations for actual M-Pesa.

Operations installation: the installed application with your own data. Available actions depend on licence status, role, configuration and the selected service.

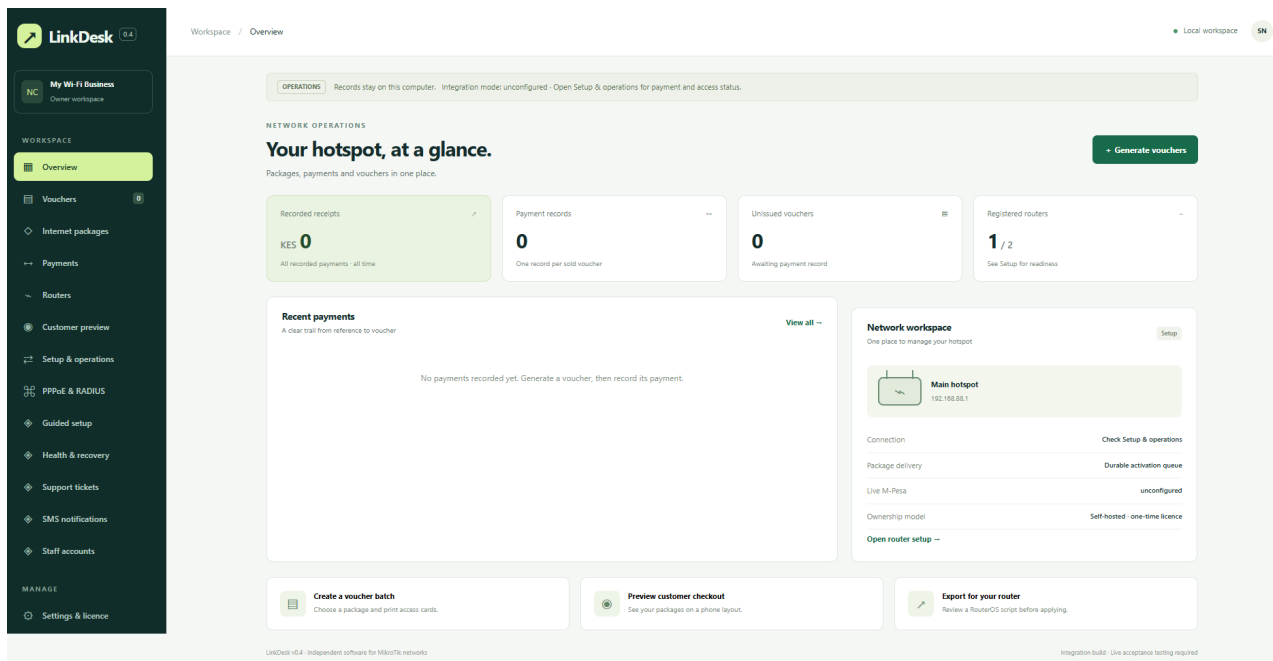
Match the message to the next step

Message or symptom	What to check
Checks require an operations workspace	You are running demo mode. Use a separate genuine operations installation for connection tests. Do not convert a demo database into customer data.
Use the public customer checkout address	You clicked Customer preview. Open the configured public HTTPS checkout instead.
Your staff role cannot perform this action	Sign in using an account with the permitted role. Owner-only integrations and licensing are blocked for cashier/technician.
Activate this installation before creating new sales	Check Settings & licence, genuine seller settings and the assigned product key. Existing paid access continues.
Router allowance reached	Compare registered routers with your purchased allowance. Ask the seller about an expanded licence.
No observations yet	Configure the applicable router, then Check routers now. A blank health screen does not confirm a live network.
RADIUS or PPPoE data cap rejected	These quota modes are not enforced by this release. Data caps apply to local HotSpot vouchers only.
Payment pending or access delayed	Diagnose payment, entitlement and connection separately. Recheck the existing order before initiating another payment.

Record the exact message, the screen, signed-in role and whether this is preview, demo or operations.

Without that information, the cause cannot be confirmed. Documentation does not remove these deliberate restrictions.

3. Dashboard and everyday sales



Overview: navigate to vouchers, packages, payments, routers and the additional operations tools. Zero records indicate the pictured workspace had no recorded sales.

Overview

Review recorded receipts, payment-record count, unissued vouchers and registered routers. In demo mode, simulated receipts are labelled separately. A registered router is a record, not proof that its connection is ready.

Internet packages

1. Open Internet packages > Add package.
2. Set a descriptive name, whole-KES price, intended service/audience, billing period and download/upload speeds.
3. For local HotSpot vouchers use fixed minutes; data cap 0 means unlimited. Choose 1-10 simultaneous HotSpot logins where applicable.
4. Use calendar-month plans only for persistent subscribers. Confirm the service/backend supports each setting before selling it.
5. Test the price, speed, expiry and device/data limits on the deployed network. Managed validity is elapsed calendar time; it does not pause when the phone disconnects.

Generate and print vouchers

1. Open Vouchers > Generate vouchers. Select the package and assigned router; choose the number of cards (the batch form permits 1-50).

2. Generate the cards, then Print cards. Inspect the username, password, package, price and assigned router before handing them out.
3. Generated cards are unissued. Printing alone does not activate access. Demo cards are marked demo-only and cannot connect a real customer.
4. After verifying an actual receipt, use Payments > Record payment. Enter the exact amount, customer, method and a unique reference for the selected voucher.
5. Check Setup & operations > Access & recovery for delivery. A recorded payment is separate from router activation and customer browsing.

Payments and CSV

Payments lists recorded transactions. Manual cash/M-Pesa entries remain operator-confirmed; authenticated gateway payments are distinguished. Internal STK references are not Safaricom receipt numbers. Export CSV for all payment records; the dashboard displays up to 1,000 recent records. Reconcile against the merchant statement.

Router export and revocation

Router export includes eligible operator-paid vouchers and revocations, excluding demo sales. Review the downloaded RouterOS script before importing it into the selected, already configured HotSpot router. Do not treat script export as network construction or confirmed delivery. Revoked vouchers should not be sold again; verify the resulting access state.

Daily checks

Review payment requests, delayed access jobs, current router observations, customer support tickets and backup failures. Confirm at least one actual customer can browse. Keep payment proof, account readiness and current connection as separate checks.

4. Installation and public payments

v0.5 update: customers can enter a product key or import a signed licence file. See `PRODUCT-KEYS.md`. Seller-prepared installers copy public activation settings automatically. Existing signing tools remain available for advanced use.

Install LinkDesk 0.5

This release is runnable source with guided installer scripts. It is not a signed Windows EXE. First installation downloads Python dependencies; internet access and Python 3.11+ are required. Use a dedicated, always-on computer or server for checkout. Sleeping laptops cannot process payments or provision accounts.

Windows

1. Install Python 3.11+ from <https://www.python.org/downloads/> with its py launcher.
2. Extract the ZIP to a normal folder. Run **Install-Windows.cmd**. It creates a private installation under `%LOCALAPPDATA%\LinkDesk`, a virtual environment, dependencies and an admin password.
3. Run `%LOCALAPPDATA%\LinkDesk\Start-LinkDesk.cmd`. Open <http://127.0.0.1:8787> and sign in.
4. Complete **Guided setup**. For public checkout, run the installed Python and launcher with `--public`, and configure an HTTPS reverse proxy as below. Use `autostart.py` to register current-user startup after Windows sign-in. An always-on Linux system service is the unattended boot deployment path.

To import a stopped v0.1 workspace during installation, run this instead of the wrapper:

```
py -3 install.py --target "$env:LOCALAPPDATA\LinkDesk" --existing-data "C:\path\to\old\data"
```

Do not point `--existing-data` at a demo database. The old folder is retained. Windows file ACLs should restrict the installation/data folder to the operator account; do not place it in a shared directory.

Linux

Install Python 3.11+, its venv package and pip using your OS package manager. Run:

```
sh Install-Linux.sh
$HOME/.local/share/linkdesk/start-linkdesk --open
```

For a server, install under a dedicated unprivileged account, adapt `deploy/linkdesk.service` to that account and installation path, then install and enable the unit using your normal systemd administration process. It restarts after failure and reboot. Do not run the application as root. The service starts both the worker and public listener.

To migrate a stopped old installation, use `python3 install.py --target /new/install/path --existing-data /old/data/path`. Keep the old folder and test the copy before retiring it.

Public HTTPS checkout

1. Point a domain you control to the always-on server. Install Caddy from <https://caddyserver.com/docs/install> using its official instructions.

2. Adapt `deploy/Caddyfile`, replacing `{$LINKDESK_DOMAIN}` with your actual hostname (or configure that environment variable for the Caddy service).
3. Start LinkDesk with `--public`. Caddy proxies **only port 8788**, the customer/callback listener. Both application ports bind to loopback; do not change that. Admin port 8787 must never be exposed by the public proxy.
4. Permit public HTTPS (and HTTP for certificate issuance/redirect) to Caddy. Keep router REST reachable only over the LAN or a private VPN. For remote admin use an SSH tunnel to local port 8787.
5. Set the exact `https://your-hostname` origin in M-Pesa setup. Do not include a trailing path. The callback URL is created per order; Caddy forwards `/callback/...` unchanged. Do not enable request-path logging because order URLs contain access tokens.
6. Add the checkout hostname to the router's HotSpot walled garden so unpaid devices can reach it. Keep required DNS/TLS reachable. The included `tools/make_hotspot_page.py` creates a login page pointing at your checkout, after you provide its real domain. Back up the existing hotspot HTML before replacing it.

The supplied proxy configuration uses a local trusted proxy. If adding more proxies, deliberately configure which forwarding headers are trusted and apply edge rate limits; do not expose the internal HTTP listener directly.

Merchant setup

Use your own Safaricom Daraja app and approved shortcode. Configure environment, consumer key/secret, STK passkey, business shortcode, receiving Party B and Paybill/Buy Goods transaction type. These merchant values must match Safaricom's provisioning; LinkDesk does not infer them. Start in sandbox. Switching to live requires typing `ENABLE LIVE MPESA`.

Credentials are encrypted in `data/settings.enc` using `data/secrets.key`. The dashboard never reads them back in plaintext. Use Settings > Complete encrypted backups to include these files and the ledger. Keep the passphrase separately and copy the encrypted archive off-device. The legacy database-only endpoint is not a complete backup. Never share private backups for support.

A callback triggers an authenticated STK Query; it does not prove payment. Successful queries are tied to the stored checkout and merchant IDs and to the price/phone sent by the server. Internal STK-... references are not M-Pesa receipt numbers. Reconcile merchant statements separately. An initiation timeout is held for operator review because blindly sending a second prompt can collect twice.

Router setup

Supported target: an existing IPv4 HotSpot on a physical MikroTik device with a serial number, RouterOS **7.12+**, working REST over HTTPS and synchronized NTP. CHR without a routerboard serial is not supported by this identity check. For PPPoE and FreeRADIUS-backed subscriber support, use the additional steps in `PPPOE-RADIUS.md`. Network/firewall creation remains an operator setup task.

Use a dedicated REST account with the permissions needed to read `system/routerboard/clock/NTP` and manage hotspot users/profiles, active sessions and schedulers. RouterOS REST uses the `rest-api` policy; the integration also needs appropriate `read/write/test` and sensitive access to verify its own created credentials. Restrict the account and HTTPS service by source address to the controller/VPN. Hardware acceptance must verify the exact policy set and script execution permissions on your chosen RouterOS version; do not use an unrestricted internet-facing admin account.

The HTTPS certificate must validate for the private management IP (IP subject alternative name). Paste a private CA's PEM certificate in Setup if needed. Certificate checking cannot be disabled. Enter the router serial and its **customer-facing HTTPS login URL ending** /login, using a certificate the customer's browser trusts. Configure the hotspot login method for HTTPS. Do not use plain HTTP PAP to send credentials.

Saving router setup checks identity, firmware and clock, then authorizes the worker. It creates only namespaced customer accounts/profiles/schedulers; conflicting objects are sent for review. It does not replace network settings or old-provider accounts. HotSpot plans configure 1-10 simultaneous logins; local PPPoE uses one CPE session. Each managed account consumes a user, profile and two scheduler entries; capacity/load testing on the actual router is required before scaling.

Customer access and time rules

The customer selects a package, approves an STK prompt and waits on the page. After authenticated payment verification, the worker enables the account and the browser submits its credentials to the configured HTTPS hotspot login. "Active" means the router confirmed the account was enabled; it does not prove the customer device completed login. The page provides credentials and a retry-connect button if browser captive-portal behavior interrupts the redirect.

For new paid accounts, calendar validity starts after the first successful router preflight, just before provisioning. That deadline is persisted before writes and never extended by retries; if a later provisioning step fails, some purchased time may elapse before access is delivered. Show that case to the operator for a service adjustment, rather than silently extending or charging again. A router offline before the initial preflight does not start the allowance.

Router-local guards check every 30 seconds, at startup and on login. They disable expired users and remove active sessions. Expected expiry tolerance is up to 30 seconds under normal scheduler operation, not a hard real-time guarantee. Unsynchronized NTP fails closed; the controller re-enables unexpired accounts after time recovers. A customer may lose access during clock recovery. Power failure itself means no service; after power returns, guards and controller reconciliation apply the saved deadline.

Acceptance before selling

Run the included tests first. Then use your Daraja sandbox, approved low-value live transactions and a physical test router to check successful/cancelled/lost-callback payments, network interruptions, router restart, controller restart, clock loss, exact expiry behavior, captive-portal auto-login, duplicate submission and a real backup restore. No live merchant/router validation has been performed for this package. Keep a signed release's previous version and a private backup available while piloting.

Subscriber services in 0.3

Open **PPPoE & RADIUS** for persistent account management. Local PPPoE can use the existing Windows/Linux controller. The supplied FreeRADIUS integration requires an always-on Linux host running both LinkDesk (- - radius) and FreeRADIUS 3.2.x. Configure, export and validate the private RADIUS bundle as described in PPPOE-RADIUS.md.

New setup in 0.4

Sign in with username owner and the existing administrator password. Staff sign in using their own credentials; only the owner can change integrations and licensing. Open **Guided setup**, configure the steps and choose **Run connection checks**. Router checks are read-only. The Daraja check uses OAuth only and

does not request payment. A "configured" step is not evidence of a successful real transaction.

The installer supports `--licence-key /seller/licence-public.pem` in addition to the separate `--trust-key /seller/update-public.pem`. These keys must come from the seller through a trusted channel. Activate the installation-specific signed licence in Settings before commercial deployment. Without a licence trust key the package runs in clearly labelled development mode; it is not an activated commercial installation.

Automatic startup

Stop a manually running instance before starting the service. From the installed release directory, using the installed virtual-environment Python:

Windows, current user:

```
& "$env:LOCALAPPDATA\LinkDesk\venv\Scripts\python.exe" autostart.py install --install  
"$env:LOCALAPPDATA\LinkDesk" --public
```

This starts after that Windows user signs in, not at unattended boot. Keep the computer awake. FreeRADIUS itself runs on Linux.

Linux, after installing as a dedicated non-root user:

```
sudo /absolute/install/venv/bin/python autostart.py install --install /absolute/install  
--public --radius  
sudo systemctl status linkdesk.service
```

The Linux helper installs a systemd boot service. It refuses to overwrite an existing unit. `autostart.py remove --install` ... removes its startup registration; on Linux this stops the service. These OS operations were not run on a customer machine during development.

Complete backup and restore

See PREMIUM-OPERATIONS.md. A restore goes into a **new** directory and creates a review gate that pauses background work, new sales and new RADIUS authorizations. Reconcile transactions after the backup, stop the old instance, and complete the review in Guided setup before resuming. A backup can be older than your merchant payments; simply restoring it must not cause you to collect again.

5. Product keys and equipment replacement

Activate LinkDesk with a product key

1. Install the prepared customer ZIP supplied by your seller. It automatically installs the seller's public verification settings.
2. Open **Settings & licence > Enter product key**.
3. Paste the five-part code from your purchase and click **Activate**.
4. Check that the licence status is active and the router allowance matches your order.

Internet is needed for this one activation. After a valid signed licence is saved, the purchased version keeps working without periodic activation-server checks. Hosting, M-Pesa, router management and SMS still have their own connectivity requirements.

If a response is interrupted, retry the same product key on the same installation. Its entitlement is returned again; it does not consume a second installation. A key already assigned to a different installation ID is rejected. Ask the seller to transfer the licence when replacing equipment. A complete restored backup preserves the installation ID; a new installation creates another ID.

Offline file option

Send the seller the installation ID shown in Settings & licence. The seller sends a signed LinkDesk-licence.json file. Choose **Import licence file**, select that file and choose **Verify & activate**. There is no need to copy JSON text. This requires the seller's public key already installed through the genuine prepared installer.

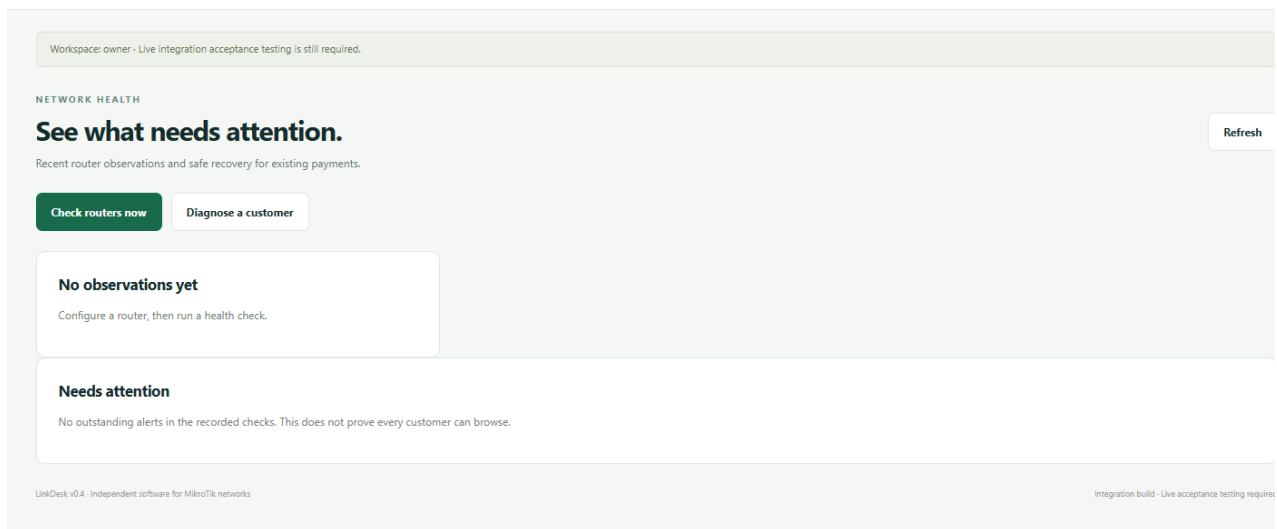
If activation fails

- **Service unavailable:** check internet and the seller's service; retry the same key later. An existing signed licence is left unchanged.
- **Key rejected:** check the code, its assigned installation and the purchased router allowance with your seller. Keys do not use I, O, 0 or 1.
- **Seller settings missing:** ask for the prepared customer installer. The generic developer source ZIP has no seller activation address or identity.
- **Invalid signature:** contact the seller. Do not accept a replacement verification key from an untrusted activation response.

One key binds to a LinkDesk installation ID. It is not a tamper-proof physical-device lock. Existing v0.4 signed licences remain valid with their original public key. Source distribution and offline operation cannot guarantee prevention of data-folder cloning or modified software.

The 17-page v0.4 installation guide remains useful for the existing network, payment and recovery flows, but its licence-JSON paste step is replaced by the product-key/file choices above. New installs use the release path releases/0.5.0 in startup commands.

6. Operations, backups, plans, portal, SMS and staff



Health & recovery in an initial workspace. Check routers now creates observations after router setup. Diagnose a customer separates payment, account and connection status.

Where to perform each task

Guided setup: follow the applicable business, licence, router, merchant, public checkout, RADIUS, startup and backup steps. Run connection checks after configuration; the Daraja check sends no payment prompt. Health & recovery: use Check routers now, review observation time, then Diagnose a customer. Choose retry/reconnect only for the existing paid account. Recovery does not collect again, extend expiry or remove suspension. Settings & licence: change your business name, activate/import your licence, request equipment transfer and manage full encrypted backups. Use Schedule backups, Run scheduled backup now and verify a restore into a new directory. PPPoE & RADIUS: create and manage persistent subscribers, record renewals, retrieve private credentials/links as owner, suspend/resume and inspect accounting. Configure the selected backend first. Support tickets: review the customer's thread, reply and close resolved tickets. Customers create and manage their tickets from the private account page. SMS notifications: configure Africa's Talking, callback and daily attempt cap; enable live sending explicitly. Persistent subscribers opt in; provider acceptance and handset delivery remain separate states. Staff accounts: the owner creates an individual username, 12+ character password and role. Use cashier for sales and technician for network recovery; never share the owner's credentials.

LinkDesk 0.4 operations guide

Included additions

1. Guided setup with business, licence, router, merchant, public checkout, RADIUS, startup and backup steps. Read-only router checks verify identity/time and read active sessions. Daraja OAuth check sends no payment prompt. Public HTTPS and FreeRADIUS daemon checks are explicit operator steps.
2. Payment / access / connection diagnostics. Readiness and live session observation are separate. Recovery reuses stored payment IDs and durable access jobs; it never sends a new STK request, resets expiry or lifts suspension.

3. Scheduled encrypted full backups, verification, retention and offline restore into a new directory.
4. Router availability and stale-accounting warnings, delayed payment/activation alerts and backup-failure alerts. Last successful router observations become unknown after 120 seconds. Router polling targets 60 seconds, subject to worker load and network response time.
5. A mobile-friendly private customer account page: paid time, payments, printable receipts, next-plan selection, scheduled periods, support tickets/replies, SMS preference and recovery. Existing private renewal links continue working.
6. Africa's Talking transactional SMS: payment confirmation, expiry reminder and account-enabled-again notice, opt-in preference, daily send-attempt cap, acceptance/delivery states and operator handling of uncertain sends.
7. Calendar-month or fixed-minute subscriber plans; local Hotspot data caps; Hotspot simultaneous-device limits; next-renewal plan changes with immutable paid-period snapshots.
8. Owner/cashier/technician accounts, server-enforced permissions, session revocation on account edits, and audit attribution.
9. Signed perpetual installation licences, expanded router limits and an equipment-transfer request workflow.

Restore a complete backup

Download from Settings using a passphrase of at least 16 characters, or configure an automatic destination outside the data directory. Encryption uses a random salt, scrypt key derivation and authenticated Fernet encryption. The full archive includes SQLite, settings, encryption key, administrator credentials and the installed licence. Staff credentials are in the database. Vendor trust keys and program files are separate installation inputs; do not accept a trust key from a backup or licence JSON.

For automatic scheduling, the passphrase is stored inside the encrypted integration settings so the worker can run unattended. It is not shown in dashboard state. Protect the host and keep a recovery copy of the passphrase away from it. A local disk backup alone does not survive loss of that disk. Retention deletes only this installation's own recorded backup files in its configured destination.

Inspect first:

```
python backups.py inspect /path/LinkDesk-full-backup.ldb
```

Restore to a directory that does not yet exist, under an existing private parent directory:

```
python backups.py restore /path/LinkDesk-full-backup.ldb --destination /private/recovered-data
```

The tool prompts for the passphrase. It verifies encryption, file hashes, database integrity, installation ID and decryption of saved settings before writing. It will not replace an existing directory. Stop the old worker. Start the new release with `--data-dir /private/recovered-data` (plus the appropriate `--public` and `--radius` flags). Inspect outstanding payment records and service dates against merchant records since the snapshot. In Guided setup, complete Restore review with reconciliation notes. Do not run both the old and restored controllers against the same routers. The review gate pauses new RADIUS logins; existing router-local sessions may remain until their prior guards/timeout expire.

Restore review records operator confirmation; it does not import a merchant statement or prove an external reconciliation was performed. No automatic merchant-statement reconciliation is claimed.

Package rules

Package setting	Supported scope
Fixed minute validity	Existing local Hotspot and persistent subscriber workflows
Calendar month	Persistent PPPoE or RADIUS Hotspot subscribers; Africa/Nairobi anniversary
Data cap, total MB	Local Hotspot voucher only; RouterOS native limit-bytes-total; 1 MB = 1,048,576 bytes
1-10 concurrent logins	Local and RADIUS Hotspot, through profile shared-users / RADIUS Port-Limit
PPPoE CPE count	Local PPPoE uses only-one; RADIUS PPPoE has no strict global simultaneous-session enforcement
New package selection	Next unpaid invoice; existing unpaid invoice must be resolved before changing it
Paid future package	Starts after existing paid entitlement; current speed stays until that boundary

A Hotspot limit counts simultaneous logins, not permanent device identities. Router native data counters are not reset on provisioning retries. RADIUS data quotas and local PPPoE data quotas are intentionally rejected rather than displayed as enforced. RADIUS traffic totals in the customer page cover retained accounting sessions, not a calendar-month bill.

A calendar month preserves the local time and anniversary day. January 31 advances to February 28/29, then March 31 while the month sequence continues. An expired account starts a new anniversary from payment time. A period transition may disconnect an existing session briefly so the new rates and timeout take effect. Early renewal extends the overall deadline but does not apply a future package's speed early.

Imported customers retain their supplied expiry and credentials. Imported entitlement is not revenue. Old v0.3 invoices retain their existing paid time; all schema changes are additive. After this upgrade, do not run v0.3 against the upgraded data: it does not understand staff restrictions and future plan policies. The v0.4 updater blocks that rollback. Older updater scripts cannot enforce the new restriction; use the current updater only.

Customer privacy and support

The existing `/renew#PRIVATE_TOKEN` link now grants access to that subscriber's account history and support threads. Treat it like a password. The fragment is not in normal request URLs; browser storage still contains the link token/payment-resume state. Use private delivery, avoid shared browsers and rotate a link if exposed. Network passwords never appear on public payment or account endpoints. Staff who do not have owner permissions cannot reveal persistent subscriber credentials.

Customers can select the next package without making a payment, create up to five open tickets, reply, close a ticket and opt out of future SMS. Receipt printing uses the browser print/save-PDF dialog. Receipts distinguish internal STK references from Safaricom receipt numbers and operator-entered payments from authenticated gateway confirmation. They are ordinary payment receipts, not eTIMS tax invoices.

SMS setup

Use an Africa's Talking account with its own username, API key and approved sender ID. Start in sandbox. Save the generated secret delivery-report URL in that account's callback settings. Live sending requires explicitly enabling it in SMS notifications. Customers separately opt in on their account page. API calls are made only by the installed app after configuration; no SMS was sent during development.

The worker prepares payment, expiry and restoration events once. It distinguishes provider acceptance from callback-reported handset delivery. A send interrupted by a timeout or restart is held for review and never automatically repeated. After checking the provider dashboard, an owner can close it or deliberately queue a new tracked attempt. The daily cap counts send attempts, not SMS segments or shillings. Long/non-GSM messages can cost multiple segments. Provider fees and credentials are not included with the software.

Delivery callbacks use an unguessable secret URL and recorded provider message IDs; they are labelled provider reports. They never activate access or verify a payment. Expiry reminders and restoration notices depend on the worker being online; do not promise guaranteed delivery or exact timing. Current notifications cover persistent subscribers, not anonymous voucher purchasers. Router warnings are in the dashboard; outbound owner alerts are not included.

Staff roles

Action	Owner	Cashier	Technician
Integrations, licence, complete backups, staff administration	Yes	No	No
Persistent subscriber credentials / private links	Yes	No	No
Generate local Hotspot vouchers / record confirmed payments	Yes	Yes	No
Subscriber renewal invoice	Yes	Yes	No
Router checks / connection recovery	Yes	No	Yes
Suspend / resume subscriber	Yes	No	No
Retry / reconnect subscriber	Yes	No	Yes
Respond to support tickets	Yes	Yes	Yes
Payment CSV	Yes	Yes	No

UI hiding is supplemental; every protected endpoint checks the authenticated role. Edits to a staff account invalidate existing sessions. Administrator settings remain on the loopback listener, not the public customer site. For staff on other computers, use an authenticated SSH tunnel or an operator-managed secure access solution; do not expose port 8787 publicly.

7. PPPoE, FreeRADIUS and subscriber renewals

PPPoE and RADIUS - LinkDesk 0.3

LinkDesk now manages persistent subscriber accounts through either **local MikroTik PPPoE secrets** or **FreeRADIUS authorization for MikroTik PPPoE/HotSpot**. FreeRADIUS remains the RADIUS protocol/authentication server. LinkDesk supplies paid entitlement policy and stores accounting events through a private REST bridge.

Supported paths

Path	Authentication	Network credentials	Renewal
Existing local HotSpot vouchers	MikroTik user database	Existing voucher flow	Buy another voucher
PPPoE local subscriber	MikroTik PPP secret	Persistent username/password	Extend the same account
PPPoE RADIUS subscriber	FreeRADIUS PAP/CHAP/MS-CHAP modules	Persistent username/password	Extend the same account
HotSpot RADIUS subscriber	FreeRADIUS PAP/CHAP modules	Persistent username/password	Extend the same account

PPPoE customers enter the network credentials into their CPE/router once. LinkDesk does not remotely configure customer CPE, optical terminals, VLANs or a PPPoE access concentrator. The customer renewal page deliberately does not submit a HotSpot browser login for PPPoE.

1. Prepare the network

Use an existing configured PPPoE server, gateway and IPv4 address pool. For local PPPoE, the controller can be Windows or Linux. For the supplied RADIUS integration, run LinkDesk and FreeRADIUS **on the same always-on Linux host**. This version does not expose the bridge remotely.

Keep MikroTik REST management and RADIUS traffic on a private LAN/VPN. The supplied configuration uses RADIUS UDP 1812/1813 with a separate long shared secret for each NAS and mandatory Message-Authenticator. It is not a RadSec/TLS deployment. Do not expose raw RADIUS across the public internet; use an authenticated private VPN. Run a patched FreeRADIUS 3.2.x and RouterOS firmware supporting `require-message-auth=yes-for-request-resp`. Local expiry guards retain the 7.12+ minimum; the RADIUS security setting must also be supported and enabled on the actual router version.

Before configuration, back up the router and the current FreeRADIUS configuration. Do not remove a working authentication service as a first test.

2. Configure the router in LinkDesk

Open **PPPoE & RADIUS > Router setup**. Select PPPoE, HotSpot or both and provide the existing router REST credentials, serial number and trusted CA if needed. For PPPoE enter the exact configured

address-pool name and local gateway IP. Pool names currently use the same 3-64-character validation as usernames; rename/map unusual pool names explicitly.

For RADIUS additionally enter the server's private IPv4 address, the router's actual RADIUS source/NAS address, and a 32-128-character shared secret made from letters, digits, _ and -. These are stored encrypted. Each configured NAS address maps to one registered router.

Router setup verifies HTTPS, serial, NTP time and the existing PPPoE server/pool. It does not silently change PPP AAA or existing network topology. Local PPPoE can proceed without a RADIUS server.

3. Install/configure FreeRADIUS

Install your distribution's patched **FreeRADIUS 3.2.x**, REST module package (commonly `freeradius-rest`) and utilities. Keep the distribution's `pap`, `chap`, `mschap` and `always` modules enabled. Do not use FreeRADIUS 4 configuration syntax with this export.

In **PPPoE & RADIUS**, enable the bridge with the Linux server's private address. Start/restart LinkDesk with `--radius` as well as `--public` when taking online payments. The supplied systemd unit includes both flags. Port 8789 is a separate loopback-only service; do not proxy it publicly. Port 8787 remains the admin UI and 8788 the public customer app.

Choose **Export configuration**. The downloaded `linkdesk-radius-private.zip` contains:

- `mods-available/linkdesk_rest`: the FreeRADIUS REST module instance.
- `sites-available/linkdesk`: dedicated authentication/accounting virtual server.
- `clients-linkdesk.conf`: exact permitted NAS source IPs and shared secrets.
- `router-radius-setup.rsc`: a reviewable router setup snippet.

The ZIP contains secrets. Do not distribute it with your product or send it to customers.

On a dedicated Debian/Ubuntu-style FreeRADIUS installation, the configuration root is usually `/etc/freeradius/3.0`; other distributions use `/etc/raddb`. Copy the module/site to the corresponding directories, include `clients-linkdesk.conf` from `clients.conf`, and enable the new module/site with symlinks. Preserve existing configuration and resolve duplicate listeners on UDP 1812/1813 before enabling this dedicated virtual server. Do not blindly disable an existing server carrying live users.

Run `freeradius -XC` (or `radiusd -XC`) to validate **your installed version's syntax and modules** before restarting. This validation was not executable in the development environment because the FreeRADIUS daemon is absent. A clean syntax check is a required installation gate.

Review/apply only the router's own section of the generated RouterOS script. If an equivalent RADIUS entry already exists, edit it instead of creating a duplicate. PPP AAA must enable RADIUS and accounting with interim updates. For HotSpot, enable RADIUS and accounting on the intended **server profile**, not unrelated profiles. Confirm Message-Authenticator compatibility in both directions. Restrict UDP ports to the configured NAS addresses.

4. Add a subscriber and collect payment

Create a subscriber with customer name, Kenyan phone number, persistent username, package, router, access service and authentication backend. Blank passwords are generated; supplied/imported passwords require 12-128 printable non-space characters. New accounts start unpaid. Choose fixed-duration minutes or a calendar-month subscriber package. Calendar months use Africa/Nairobi anniversary times and clamp short

months while retaining the billing-day anchor. See PREMIUM-OPERATIONS.md.

Choose **Credentials & link** to privately deliver the account's credentials and private renewal URL. Keep the URL private; it is a bearer capability. The token is in the URL fragment to avoid sending it in ordinary request paths. Renewals use the existing M-Pesa checkout and authenticated query workflow. The app does not expose the network password from a payment/order endpoint.

For cash or manually confirmed M-Pesa use **Record renewal**. Every invoice has one unique payment reference and can extend the account only once. An uncertain STK transaction is resolved in Operations after checking the merchant statement.

A paid period adds to $\max(\text{previous expiry}, \text{recorded payment time})$. A server/router outage does not reset this deadline, and recovery uses the stored payment timestamp. RADIUS/PPPoE subscriber time can therefore elapse while a paid service cannot be delivered; resolve service adjustments with the customer rather than charging again. A suspended account may renew, but payment does not lift the suspension.

The worker creates only owned profiles, secrets and scheduler entries. In RADIUS mode it **does not create local secrets**, because a matching local account would take precedence over RADIUS. Any such conflict goes to review. Active means the owned router configuration has been confirmed and policy is eligible; it does not prove a PPP client completed authentication.

5. Expiry, suspension and recovery

Local PPPoE uses a named profile, `service=pppoe` secret, single-session local profile, fixed calendar deadline and router-local 30-second/startup/on-up guards. Expiry disables the secret and removes active PPP sessions. Guards fail closed when router time is not synchronized.

RADIUS authorization requires: a trusted bridge request, configured NAS source, matching subscriber router/service, RADIUS backend, router readiness, paid unexpired entitlement and no suspension. It supplies the known-good password to FreeRADIUS's control list; **FreeRADIUS still verifies the presented PAP/CHAP/MS-CHAP password**. It never forces `Auth-Type = Accept`. Attribute values explicitly disable expansion, including imported passwords.

The RADIUS reply carries remaining `Session-Timeout`, MikroTik profile/rate limits and a 60-second accounting interval. Local guards and worker disconnection supplement the timeout. If LinkDesk is unavailable, new RADIUS authentication fails closed; an existing session retains its previously granted timeout and router guards. Resume/retry uses the same deadline.

Suspension immediately denies new RADIUS authorization and queues existing-session disconnection via **verified RouterOS HTTPS REST**. This release does not send UDP CoA/Disconnect-Request packets. After renewal, an existing session is disconnected once to reload timeout/rate settings; customers may see a brief reconnection. If router REST is down, delivery/disconnection retries rather than inventing a successful disconnect.

Local PPPoE uses `only-one=yes`; RADIUS PPP does **not** have a global simultaneous-session enforcement service in this release, because RouterOS ignores that local profile option for RADIUS. HotSpot RADIUS replies include the package device allowance as `Port-Limit` (default 1). Do not advertise strict single-session PPP RADIUS enforcement until a dedicated accounting/session-limit policy is implemented and tested.

6. Accounting

The bridge accepts Start, Interim-Update, Stop and NAS Accounting-On/Off events only from the mapped NAS. It stores cumulative input/output counters and handles gigawords. Duplicates are not added again, out-of-order packets cannot reduce counters or reopen a stopped session, and delayed restart reports do not close newer sessions. These are usage observations, not a billing balance or a data-quota enforcement engine.

The UI labels counters as **NAS input/output** because direction semantics need verification on the deployed RouterOS service. Reports older than three minutes are labelled stale rather than treated as proof of a live connection. Missing accounting packets can leave stale history. Inspect/reconcile on the router when troubleshooting; no periodic SQL/radacct synchronization is implied.

7. Migrate existing subscribers

Use docs/subscriber-import-template.csv: source_id, username, password, customer, phone, package_id, router_id, service, backend, expires_at. Include an ISO expiry with timezone, e.g. 2026-10-10T18:00:00+03:00, using the customer's actual future paid expiry. Preview is read-only, expires after one hour, and rejects duplicate IDs/usernames and invalid rows. Commit is atomic and idempotent. Passwords are encrypted in the database; no payment is collected or revenue booked for imported entitlement.

Plan the old-account cutover explicitly. LinkDesk will not adopt or delete an existing foreign-owned local secret. For local-to-local cutover, retire/rename the old secret after a backed-up test so LinkDesk can create its owned replacement with the chosen persistent credentials. For local-to-RADIUS cutover, remove/rename the shadowing local secret only when the replacement has been prepared. Preserve rollback access and coordinate the short interruption with the customer.

Live acceptance still required

Test on an actual MikroTik and the installed FreeRADIUS daemon: config syntax, PAP/CHAP/MS-CHAP correct and incorrect passwords, unknown/wrong NAS, no local-account bypass, first payment, duplicate requests, renewal/reconnect, expiry, suspension, restarts, NTP loss, bridge outage, NAS accounting and old-provider cutover. Packet-level RADIUS, PPPoE client authentication and physical-router behavior were not run in this development environment. The included tests simulate router/provider transport and exercise the real LinkDesk policy/accounting code locally.

Customer portal and future packages in 0.4

The private renewal link now opens account history, printable receipts, support tickets, opt-in SMS controls, connection diagnosis and next-renewal package selection. It remains a secret bearer link. Rotate it from the owner dashboard if exposed. A next-plan choice is not payment; paid future periods apply at their recorded start, without immediately replacing the current speed. Data-capped packages are limited to local Hotspot vouchers in 0.4; RADIUS counters remain observations, not a quota balance.

8. Migration and cutover

Move existing customers

The migration tool handles **paid HotSpot entitlements via CSV**. It generates new LinkDesk usernames and passwords; it does not copy or overwrite an old billing provider's router accounts, PPPoE users, quotas or RADIUS records.

1. Export a customer list from the current provider. Map package and router IDs to those shown in LinkDesk. Give every customer a stable, unique `source_id` across all imports.
2. Prepare columns `source_id`, `customer`, `package_id`, `router_id`, `expires_at`. `expires_at` is an ISO date/time with timezone, such as `2026-10-10T18:00:00+03:00`. Replace the example date with the actual, future paid expiry. Limit: 1,000 rows / 500 KB per file. Past or more-than-366-day expiries are rejected.
3. In Setup & operations, choose Import customers and preview the file. Row errors block the whole import. Preview changes no customer records and expires after one hour.
4. Verify count, package mapping and original deadlines, then type `IMPORT PAID CUSTOMERS`. This queues access without sending a payment request or booking new revenue. Duplicate file commits return the existing outcome; duplicate source IDs are rejected.
5. Check Access & recovery for activation. Print new credentials from Vouchers and have each customer reconnect. Retire the old accounts in the old system after successful switching; LinkDesk does not delete them automatically. Coordinate the cutover to avoid two active accounts for one customer.

Keep the original provider export privately until reconciliation is complete. New-account usage is bounded by the remaining calendar validity; historical usage and accounting are not imported. Imported deadlines never reset on retries.

For existing LinkDesk v0.1 users, the installer can copy the stopped data folder. Legacy exported vouchers retain their original cumulative-uptime rules and are not silently converted to managed accounts. Operator-paid, unexported vouchers are eligible for queued managed activation once router setup is enabled. Test this transition on the copied workspace first.

Persistent PPPoE / RADIUS accounts

Version 0.3 has a separate subscriber importer that preserves supplied usernames/passwords and fixed paid expiry. Use `subscriber-import-template.csv` and the **PPPoE & RADIUS** screen. Follow `PPPOE-RADIUS.md` for the old-router account cutover; foreign-owned local accounts are not overwritten or silently adopted.

9. Signed updates and recovery

Updates require an authentic seller-signed release and the trusted update public key. Take a full encrypted backup first and retain the running version. Stop the application/service before using the currently installed trusted updater. Do not install an old source ZIP over customer data. For a v0.5 Windows installation, use the installed virtual-environment Python and `releases/0.5.0/updates.py`; on Linux use the equivalent paths. The apply command is `updates.py apply --install YOUR_INSTALL --archive SELLER_SIGNED_ZIP`. Substitute actual paths; the update key is separate from the licence key. Read the seller's release-specific dependency and compatibility instructions, restart, then check login, payment/access queues and an actual customer connection. The updater verifies signatures and hashes; that is not a promise the new release has no bugs. Code rollback does not restore older payment data. Do not downgrade a v0.5 workspace to old versions that do not understand staff restrictions or future paid-plan policies. Disaster recovery uses a complete backup plus merchant reconciliation and the restore review gate described in the operations chapter.

10. Acceptance and handover

Confirm your deployment, not just a feature list

The supplied package's automated tests use simulated router/payment/SMS transports. This manual does not certify your deployment. The WayaDesk licence-purchase test shown earlier does not validate customers' internet payment and router flows. Record a result for each applicable item:

1. Owner login, active licence, purchased router allowance and staff restrictions.
2. Public HTTPS checkout from mobile data and unpaid Wi-Fi.
3. Approved controlled M-Pesa payment, cancellation, duplicate submit, delayed/lost callback and uncertain-payment review.
4. Paid account enabled and the actual customer phone/CPE browsing.
5. Expiry, supported speed/device/data rules, router restart, controller restart and NTP recovery.
6. Windows sign-in startup or Linux boot service, as selected.
7. Full encrypted backup and restore on a separate test installation; reconcile post-backup payments.
8. If deployed: local PPPoE or actual FreeRADIUS authentication, wrong-password rejection, accounting, renewals and suspension.
9. If deployed: private customer portal, printable receipts, support tickets and real SMS delivery.
10. Actual customer load and a documented old-provider cutover/rollback plan.

Keep the handover record privately

Business/site; version/date; installation ID/router allowance; seller/installer contact; agreed support hours and update terms; public checkout hostname; backup location and last restore test; outstanding acceptance items. Never write passwords, Daraja secrets, M-Pesa PINs, private account links or backup passphrases into a shared handover PDF. Hardware, domain/hosting, power, internet, merchant fees and SMS charges are separate unless your agreement includes them.

Ask for help with a specific failure

Provide version, operating system, screen/action, role, exact error, time in EAT and whether one or all customers are affected. Redact credentials, product keys and customer links. Keep private backups private.

Sources and version notes

Based on the local LinkDesk-v0.5.0 release: README.md, INSTALLATION.md, PRODUCT-KEYS.md, PREMIUM-OPERATIONS.md, PPPOE-RADIUS.md, MIGRATION.md, UPDATES.md and the dashboard/server action checks. Some included documents retain older chapter headings because those features carry forward into v0.5; the product-key chapter supersedes legacy activation wording. Screenshots are real supplied interface captures and a public WayaDesk licence checkout capture. Initial workspace screenshots do not show completed network deployment. This manual includes no seller signing keys or private customer records.