HAMR-FORGE

FORGE OS · PUBLIC TECHNICAL WHITEPAPER

Platform Architecture & Operational Resilience

A self-hosted hub-and-spoke platform for persistent, policy-controlled communications

PUBLIC RELEASE

VALIDATED BASELINE V0.9.239 · SUPERSEDES V0.9.173 EDITION

PUBLISHED BY HAMR-FORGE, LLC · AUGUST 2026

Executive Abstract

FORGE OS is a self-hosted secure networking platform from HAMR-FORGE, LLC. It combines a centralized HUB control point with autonomous SPOKE appliances: encrypted multi-path transport, dynamic routing, and local fail-closed behavior on every site.

Around that core, the platform provides enterprise service integration, signed A/B firmware, and recoverable device identity. This paper explains how those elements form an operationally resilient system without requiring a vendor cloud in the data path or control path.

Intended audience. Network architects, security architects, NOC leadership, systems integrators, and technical evaluators.

> **Publication boundary: v0.9.239 capability is stated as current. Future work is labeled ROADMAP — NOT INCLUDED IN v0.9.239.**

1. Operational Problem

Remote and austere sites rarely fail in a tidy way. Individual carriers may degrade, disappear, return with a different address, or preserve reachability while delivering unusable latency. The central facility can also restart or become temporarily unreachable. An operational platform must distinguish those events, preserve a secure posture, and recover without turning every site into a network-engineering exercise.

FORGE OS approaches this as an appliance problem rather than a collection of unrelated Linux services. The operator works with explicit roles — HUB and SPOKE — while the platform coordinates transport, routing, isolation, service integration, telemetry, and recovery behind a consistent management interface.

- No mandatory HAMR-FORGE cloud dependency; the customer controls the HUB, SPOKE appliances, certificates, routes, logs, and recovery artifacts.
- Multiple WAN paths can contribute to one encrypted MPTCP session instead of waiting passively for a primary link to fail.
- A SPOKE that loses its overlay defaults to a contained posture rather than silently leaking protected traffic to an arbitrary WAN route.
- Routine control-plane and firmware operations are designed to preserve the independent Go data plane when its binary is unchanged.

2. System Roles and Responsibility Boundaries

Role	Primary responsibilities	Continues locally during management interruption
HUB	Terminates authenticated overlay sessions; coordinates site identity, routing, policy, enterprise services, audit collection, and recovery escrow	Existing data-plane process and established forwarding can remain independent of a web/API restart
SPOKE	Bonds WAN paths; originates the secure tunnel; provides LAN/VLAN services; enforces isolation; runs local routing and health logic	WAN health, isolation, routing, and reconnect logic operate on the appliance
Customer NOC	Owns placement, upstream routing, external identity, DNS/NTP, SIEM/NMS, backups, and recovery custody	Customer infrastructure remains the authority for its own continuity plan
HAMR-FORGE	Publishes signed software, appliance behavior, validation guidance, and product documentation	Does not require possession of customer traffic, private CA material, or operational logs

> **Design principle: FORGE OS provides connection resilience and secure integration. It does not replace the customer's NOC continuity, identity, SIEM, DNS, NTP, or upstream-routing architecture.**

3. Reference Architecture

Each SPOKE establishes an outbound TLS 1.3 connection to the HUB on the configured transport endpoint. MPTCP allows additional reachable WAN addresses to join the same logical connection as subflows. Inside the authenticated overlay, FORGE OS creates a point-to-point tunnel and establishes the selected routing protocol. BGP supports independently routed sites; OSPF supports sites that participate in a shared routing domain.

FIGURE PLACEHOLDER

platform architecture — customer NOC and enterprise services, HUB, spokes with bonded WAN paths, routed overlay

The public architecture intentionally separates transport identity from routed identity. WAN addresses are underlay reachability; tunnel addresses and dynamic-routing state describe the protected overlay. LAN networks are advertised only when the operator explicitly enables upstream advertisement.

A SPOKE's configuration carries an ordered list of HUB destinations rather than a single address. In a single-HUB region that list has one entry and behaves exactly as a single address did. Where a standby is assigned, order expresses operator intent: candidates are tried in assignment order, gated on reachability. They are never ranked by measured latency — a HUB is chosen because the operator assigned it, not because it answered fastest.

Assignments are documents signed by the home HUB and carry monotonic epochs. A SPOKE rejects an assignment that is stale, or that conflicts with one it already holds at the same epoch. SPOKEs take no part in selecting a standby: HUBs federate and compute, SPOKEs obey. Certificate status, revocation, and standby assignments replicate between federated HUBs; recovery escrow does not, and remains held by each spoke's home HUB.

4. Transport and Routing Convergence

The transport process is implemented separately from the Python management plane. When a firmware package changes only management components, the installer preserves the existing role-specific data-plane PID and live sessions. If the deployed Go binary changes, the installer performs a controlled data-plane restart and verifies that the active process is executing the newly installed binary.

After boot, the SPOKE retry loop continues while a HUB endpoint remains configured. If the appliance starts before DHCP has produced a gateway, the route guard — the gate that installs the narrowly scoped HUB route only once a WAN gateway is resolvable — later reconciles the HUB pinhole when reachability appears. This closes the historical gap in which a cold-booted SPOKE could remain disconnected until an operator clicked reconnect.

- MPTCP preserves one logical transport across multiple underlay paths.
- Path self-healing can rotate a missing endpoint without discarding the surviving session token.
- Dynamic routing reconverges after tunnel recovery and removes unavailable overlay paths from the routing information base.
- Traffic continuity depends on application behavior, routing timers, underlay conditions, and whether the data-plane binary itself must restart; FORGE OS does not claim zero packet loss under every failure.

5. Fail-Closed Site Behavior

A SPOKE is intended to be safe when unattended. Its isolation controller begins from a restrictive posture and relaxes only the rules required by the live overlay state. If the tunnel is unavailable, protected LAN forwarding does not automatically fall through to a commodity Internet default route. A narrowly scoped HUB pinhole remains available so the appliance can re-establish the authenticated tunnel.

Internet breakout is a separate, explicit posture. Where authorized, an administrator can enable NAT egress on a selected interface; the interface and dashboard show that state. It is not implied merely because a WAN has a default gateway.

> **A fail-closed claim is bounded by the FORGE OS forwarding configuration. Physical bypass wiring, external switches, hypervisor networking, or customer-installed routes remain outside the appliance boundary.**

6. Firmware Resilience and Rollback

FORGE OS firmware is signed and installed into alternating application slots. The inactive slot is staged and verified before activation. A rollback returns the application payload to the prior slot while persistent configuration, recovery escrow roots, and the security-audit chain remain outside the replaceable application slot.

The firmware chain also maintains a monotonic root-boundary manifest (introduced at v0.9.118): a protected record of the privileged dispatchers, wrapper tools, rollback logic, and firmware verification key that may advance with new releases but can never silently regress. A signed downgrade may restore older application code, but it may not replace that protected root boundary with a weaker historical copy. Forward installation and downgrade/rollback are exercised on both SPOKE and HUB roles as part of the release gates.

Layer	Update behavior	Continuity objective
Data plane	Preserved if hash is unchanged; controlled restart if changed	Avoid unnecessary tunnel disruption while preventing stale-binary execution
Application slot	Alternates A/B	Recover to a known signed application payload
Persistent state	Stored outside application slots	Retain configuration, audit identity, and escrow roots
Root boundary	Monotonic protected manifest	Prevent an older signed package from weakening the privileged execution boundary

7. Recovery and Operator Burden

Fresh encrypted installations create a TPM-bound protector and an independent LUKS recovery credential. The setup workflow also generates the appliance break-glass credential. Both values can be copied during setup for immediate troubleshooting. After a SPOKE authenticates to its HUB, its recovery record is encrypted for HUB custody and removed from normal SPOKE recovery custody according to the escrow workflow — the platform's rule that site recovery credentials live encrypted at the HUB, not on the field appliance.

The HUB exposes recovery material only to an authorized administrator after password step-up. Reveal and export operations are security-audited. The HUB disaster-recovery bundle is separately passphrase-sealed and contains the identity material needed for a replacement HUB to adopt the recovered identity during first-run setup.

- Configuration backup and disaster recovery are different artifacts and are labeled separately.
- A diagnostic bundle is secret-free by design and suitable for support workflows; it is not a recovery bundle.
- Recovery success still depends on protected off-box custody of the HUB recovery bundle and its passphrase.

8. Monitoring and Operational Truthfulness

The management interface reports observed state rather than treating saved configuration as proof of health. Examples include live DNS resolution, NTP synchronization, routing-neighbor state, MPTCP subflow counts, syslog delivery state, audit-chain verification, and active firmware-slot health. The design deliberately distinguishes configured, connected, synchronized, forwarding, degraded, and unavailable.

At fleet scale, the HUB pairs an attention-oriented dashboard — site health with an explicit needs-attention count, live per-WAN and tunnel throughput, and recent operator-impacting activity — with an offline geographic Fleet Map served entirely from signed, customer-installed map packs. No external mapping service ever enters the data path or control path.

FORGE OS also exports TLS-protected syslog, SNMPv3 polling and traps, and scheduled diagnostic audit bundles. These integrations let the NOC use its existing SIEM and NMS as the durable system of record instead of requiring a proprietary cloud console.

> **The local audit journal is hash chained and permission restricted, but a root administrator can ultimately rewrite local storage. The stronger durability boundary is prompt forwarding to a separately administered collector. Public claims therefore use "local integrity chain plus off-box durability," not "locally tamper-proof."**

9. Validated v0.9.239 Baseline

This paper describes the publicly releasable behavior of FORGE OS v0.9.239, the current validated pre-release baseline. Validation evidence cited here comes from the project's automated release gates, virtual-appliance exercises, and physical hardware testing. The term validated does not imply a third-party certification, government approval, product-level FIPS validation, or general-availability designation.

Area	Validated evidence	Qualification
Virtual fleet	HUB plus multiple BGP and OSPF SPOKEs; MPTCP subflows; firmware forward/rollback; audit and escrow continuity	Lab validation, not a capacity certification
Physical appliance	UEFI Secure Boot, physical TPM 2.0, LUKS2 enrollment, PCR reading, auto-unlock, recovery-key rotation and HUB escrow	Hardware-specific qualification remains required
Cold boot and outage	SPOKE restart, delayed WAN lease, HUB restart, and extended HUB power-off recovery	Observed lab outcomes; site timers and carriers vary
Endurance	72-hour firmware-lock soak: one HUB, three spokes (two BGP, one OSPF), continuous bidirectional 100 Mbps iperf3 per spoke toward the HUB. No errors, drops, or faults recorded.	Sustained-load endurance at a fixed offered rate. Not a capacity result, not a scale result, and not a multi-HUB result — the soak ran single-HUB.
Release controls	Behavioral CI gates, migration tests, signed firmware, privileged-helper validation, rollback boundary verification	Supports release confidence; does not replace independent assessment

10. Roadmap — Not Included in v0.9.239

Roadmap items are architectural direction, not v0.9.239 capability. They are visually labeled and must not be used as procurement or deployment claims.

- Independent third-party penetration test and remediation cycle.
- Customer-specific STIG/FIPS control mapping where contractually required.

Earlier roadmap items from the v0.9.118 edition of this paper — the attention-oriented fleet dashboard, the geographic Fleet Map, and the AES-256-GCM storage-format migration — shipped between v0.9.119 and v0.9.173 and are described above as current capability. Multi-HUB geographic resilience, with deterministic per-SPOKE failover, replicated readiness state, and operator-controlled failback, was carried as roadmap in the v0.9.173 edition and has since shipped; it is described in Sections 3 and 4.

> **Roadmap designs require implementation, migration testing, operational validation, and documentation before becoming product claims.**

11. Deployment Considerations

- Treat HUB placement, upstream routing, power, hypervisor HA, backup, NTP, DNS, identity, SIEM, and NMS as customer-owned infrastructure decisions.

- Qualify each physical platform for UEFI, Secure Boot, TPM 2.0 behavior, storage controller support, NIC behavior, thermals, and power restoration before approval.
- Size WAN paths and QoS from measured service behavior, not carrier branding or radio indicators.
- Preserve an offline HUB recovery bundle and test restoration on a replacement system before production dependence.
- Use the release gate and site acceptance procedure for every approved firmware and hardware baseline.

12. Conclusion

FORGE OS v0.9.239 demonstrates a coherent architecture for customer-controlled, hub-and-spoke communications across unreliable underlays. Its resilience does not come from a single feature. It comes from the combination of independent transport, authenticated multi-path sessions, dynamic routing, fail-closed site behavior, truthful health reporting, protected recovery, signed A/B firmware, and explicit ownership boundaries.

The result is an appliance model intended to reduce field-operator burden without hiding consequential state from the NOC. Remaining roadmap work is deliberately separated from the validated baseline described in this paper.

Document Control

Field	Value
Publisher	HAMR-FORGE, LLC
Platform	FORGE OS
Validated baseline	v0.9.239
Classification	Public release
Release date	22 August 2026
Supersedes	v0.9.173 edition (August 2026)
Claims boundary	No third-party certification or general-availability claim