DevCorp

MANAGED HOSTING

SINGLE-SERVER MANAGED HOSTING

Managed Hosting: Essential

A practical production foundation for smaller services, pilots, and controlled launches.

Essential keeps the architecture intentionally simple: one managed Linux server, Docker-based deployment, TLS, basic monitoring, and recoverable backups. It is a good fit when the main priority is getting a stable production environment online without adding unnecessary platform complexity.

Service model

Managed hosting

Package

Essential

Document

Version 1.0 - 2025

Package Overview

Essential keeps the architecture intentionally simple: one managed Linux server, Docker-based deployment, TLS, basic monitoring, and recoverable backups. It is a good fit when the main priority is getting a stable production environment online without adding unnecessary platform complexity.

Best fit

- Small public websites, portals, admin tools, and early SaaS products
- Predictable traffic with no strict high-availability requirement
- Teams that accept scheduled maintenance and manual scaling

Plain-English value

- The infrastructure is monitored and maintained by DevCorp.
- Operational responsibilities are defined before production use.
- Technical controls can be expanded as risk, traffic, or compliance needs grow.

Managed launch

A clean production environment with deployment, TLS, monitoring, and backup basics handled for you.

Lower complexity

Fewer moving parts mean easier troubleshooting, clear responsibilities, and faster onboarding.

Upgrade path

The design can grow into a larger VM, staging environment, or Professional package later.

Reference Architecture

The architecture is chosen to match the operational risk of the package: simple where simplicity is safer, layered where availability and control matter more.

Edge

DNS, TLS and restricted public ports

>

Server

Linux VM with nginx and Docker Compose

>

Recovery

Monitoring, database backups and updates

Technical foundation

- One EU-based cloud VM running Linux
- Host nginx for TLS termination and reverse proxying
- Docker Compose for application, database, and supporting services
- Local database on the same VM for simple operations
- Zabbix agent for host-level monitoring

Compute Platform

We choose the compute platform based on compatibility, stability, and cost/performance. AMD/x86 is usually the safest default; ARM is considered when dependencies and container images are validated.

Platform

AMD/x86 shared vCPU VM is the default. ARM can be considered only when all application dependencies and images are validated.

Sizing

Typical starting point: 2-4 vCPU, 4-8 GB RAM, NVMe storage. Final sizing depends on database and traffic profile.

Scaling

Vertical resize or migration. Some maintenance window should be expected.

AMD / x86 strengths

- Best default for PHP, Node.js, MySQL, Docker images, and broad package compatibility.
- Predictable choice for production systems with third-party binaries or legacy dependencies.
- Dedicated or performance-class AMD nodes are preferred for sustained production load.

ARM strengths

- Good cost/performance for compatible stateless services, workers, and staging systems.
- Requires image and dependency validation before production use.
- Usually introduced after the baseline environment is stable and measurable.

Security and Firewall Layers

Security is built in layers: public traffic controls, server firewalls, restricted operator access, secret handling, patching, and optional WAF or VPN controls.

Firewall and access controls

- Cloud firewall allows only HTTP/HTTPS and restricted SSH
- Host firewall mirrors cloud firewall rules for defense in depth
- SSH key-only access; no password login
- Optional basic CDN/WAF in front of public traffic
- Fail2ban or equivalent brute-force protection

Security baseline

- Least-privilege access for operators and deployment paths
- Secrets stored outside source control
- TLS certificates managed and monitored
- Security updates handled through planned maintenance
- Suspicious login and brute-force activity monitored where applicable

Monitoring, Backup, Operations

A managed service is only useful when it can be observed, recovered, and operated consistently. This package defines what is watched, what is backed up, and how routine operations are handled.

Monitoring

- CPU, memory, disk, network, and load average
- Public service availability and application heartbeat
- Container restart status and Docker daemon health
- SSL certificate expiry and domain reachability
- Basic email or chat alerts for urgent host/service issues

Backup and recovery

- Automated database backup
- Configuration and deployment metadata backup
- Short retention suitable for smaller environments
- Restore assistance during business hours

Operational support

- Business-hours support
- Scheduled maintenance windows
- Routine security updates
- Deployment assistance for planned releases
- Best-effort incident response within the support window

Support terms in brief

- Support is provided during agreed business days and business hours.
- Production-down issues are prioritised inside the support window.
- Planned maintenance is scheduled in advance where practical.

Add-ons and Boundaries

The package can be extended with add-ons. Boundaries are intentionally explicit so customers understand which risks are covered by the selected service level.

Available add-ons

- Separate staging environment
- Longer backup retention
- Cloudflare CDN/WAF
- Quarterly restore test
- Monthly operations summary

Important boundaries

- Single server means no automatic failover
- No dedicated staging unless added separately
- No guaranteed zero-downtime infrastructure maintenance
- No 24/7 incident coverage
- Limited historical logs and monitoring depth

Not included by default

- Application feature development is not included unless agreed separately.
- Third-party vendor fees, licenses, and account ownership remain with the customer.
- Legal, compliance, and security audits are not included in the base package.

Shared responsibility

- DevCorp manages the hosting platform, monitoring, backups, and agreed operational processes.
- The customer remains responsible for business content, third-party account ownership, and timely approval of changes.
- Final support windows, response targets, and legal commitments are defined in the service agreement.

Typical onboarding path

- Confirm application architecture, domains, DNS, secrets, integrations, and expected traffic.
- Select compute platform, region, backup target, firewall model, and monitoring scope.
- Deploy the environment, run smoke checks, validate backups, and document access.
- Agree support contacts, maintenance windows, incident priorities, and escalation path.