



Appliance, Device, Container and VM Go-Live Checklist

Before your test box becomes production

Diversified Tech Solutions

Use this before a test system is exposed to the internet or anyone depends on it. If you cannot say when a test system became production, run this checklist now.

Tick each box as you go. Anything you cannot tick is a decision to make on purpose, not a step to skip.

Scope What this applies to

Run this for any of the following (and anything similar):

- Monitoring and observability servers
- Firewalls
- VPN concentrators
- Switches and routers
- Wireless controllers and access points
- Load balancers
- UPS and PDU network management cards
- IP cameras and network video recorders (NVRs)
- Access control and door entry systems
- HVAC and building management controllers (BMS)
- PBX systems and VoIP phones
- Print servers and multifunction printers (MFPs)
- NAS and SAN storage
- Backup appliances
- Hypervisor management and out-of-band management consoles (IPMI and BMC style)
- KVM-over-IP devices
- Log collectors and SIEM collectors
- DNS and DHCP appliances
- Email gateways
- Web filters
- IoT hubs
- Digital signage players
- POS terminals
- Environmental sensors
- Lab containers and VMs spun up from templates

1 Treat every test as future production

- ☐ Named owner assigned on day one.
- ☐ Added to asset inventory, marked "test," with a planned review or teardown date.
- ☐ Baseline configuration exported or recorded at first setup.

2 Network exposure (internal until go-live)

- ☐ During testing, reachable only via VPN, internal network, or IP allowlist. Not on the open internet.
- ☐ Every listening port listed and justified (`ss -tlnp`, firewall rules, port forwards, cloud security groups).
- ☐ Management consoles and APIs never exposed directly. Use VPN, an authenticated reverse proxy, or an allowlist.
- ☐ Agent and management ports restricted to the specific hosts that need them.
- ☐ "Temporary" port forwards and firewall exceptions from testing removed.

3 Accounts (audit every one)

- ☐ All accounts listed: built-in, initial, admin, service, API, and OS-level.

- ☐ Default or initial admin account disabled, renamed, or given a strong unique password. Adding a new admin does not count.
- ☐ Groups and roles still held by the default admin reviewed.
- ☐ No default or shared credentials anywhere in the stack (application, database, OS, web server, SNMP communities).
- ☐ Every admin is a named person. Accounts nobody claims are removed.
- ☐ Guest or anonymous access disabled.

4 Config drift and change control (the go-live gate)

- ☐ Current configuration compared to the baseline from section 1. Every difference reviewed and either kept on purpose or reverted.
- ☐ Features enabled "to try something" during testing reviewed and disabled if unused.
- ☐ Debug, verbose, or sample settings and content removed.
- ☐ Patched to a current supported release.
- ☐ Backups configured and a restore tested.
- ☐ A named person signs off, with a date, that the system is production. Inventory updated from "test" to "production."
- ☐ Going forward, every change goes through change control (at minimum a written record of who, what, when, and why).
- ☐ Periodic drift review scheduled (configuration compared to the approved baseline).

5 MFA

- ☐ MFA required for every admin-level account on web consoles and APIs where supported.
- ☐ If the device does not support MFA, the console is reachable only through an MFA-protected VPN or jump host.
- ☐ SSH uses keys, with password login disabled where possible.

6 Remote script and command features

- ☐ Inventory every feature that runs commands, pushes configuration, or executes scripts on other systems.
- ☐ Disable remote command or script execution features you do not use, both on the appliance and on any agents it controls.
- ☐ Where needed, restrict to specific commands, specific hosts, and a small set of users or roles.
- ☐ Script creation and editing rights limited to the fewest people possible (treat as root on reachable hosts).
- ☐ Unused scripts deleted.
- ☐ Logging of remote command execution enabled on agents where available.
- ☐ API tokens: dedicated least-privilege users, expiration dates, regular rotation, unused tokens removed.

7 Audit logging and alerting

- ☐ Admin actions logged and forwarded off the box.
- ☐ Alerts on successful admin logins (odd hours, unfamiliar sources), not just failed logins.
- ☐ Alerts on creation, change, deletion, or execution of anything that can run code.
- ☐ Alert when a script or job is created and deleted within a short window.
- ☐ Alerts on configuration changes outside approved change windows.

8 Host persistence checks

- ☐ Crontabs for all users: `crontab -l -u <user>`, `/var/spool/cron/`, `/etc/crontab`, `/etc/cron.d/`, `/etc/cron.*`.
- ☐ systemd units and timers: `/etc/systemd/system/`, `/lib/systemd/system/`, `~/.config/systemd/user/`, `systemctl list-timers --all`.
- ☐ `/etc/ld.so.preload`, unexpected kernel modules (`lsmod`), and process lists that do not match CPU usage.
- ☐ Unexplained sustained CPU usage and unfamiliar outbound connections.
- ☐ File integrity monitoring on cron and systemd directories and `/etc/ld.so.preload`, with alerts on change.

9 If abuse is suspected

- ☐ Isolate affected hosts and capture evidence before cleanup.
- ☐ Rotate passwords and API tokens; end active sessions.
- ☐ Remove attacker-created scripts; disable remote commands pending review.
- ☐ Remove persistence and verify after reboot, or rebuild where rootkit-style persistence is found.
- ☐ Sweep every host the appliance could reach.



Want a second set of eyes before go-live?

Contact Diversified Tech Solutions.

diversifiedtechsolutions.com

423-888-0252 | avery@diversifiedtechsolutions.com

This checklist is general guidance and not a substitute for a review of your specific environment. Free to share with attribution to Diversified Tech Solutions.