

# Self-healing Docker host and an 8-machine lab managed entirely by remote automation

Hybrid physical/virtual fleet: domain controller, workstation VMs, Docker host, monitoring node · Role: everything

**Outcome:** Production host outages detect and alert within 5 minutes with state-change emails (no alert spam); the entire fleet deploys and heals over WinRM without touching a console.

## The Situation

A production Docker host running an internet-facing platform died silently one day - the container engine exited and nothing knew. Separately, an 8-machine lab (domain controller, five Windows workstation VMs across two hypervisors, a monitoring node) needed administration without ever sitting at a console - the machines span two networks joined by a mesh VPN.

## What Was Built

- **Watchdog service** (PowerShell, scheduled task) that checks engine and container health every 5 minutes, restarts containers when the engine is healthy, and emails on health-state *changes* only - one alert when it breaks, one when it recovers, never a flood. Notifications sent via Microsoft Graph with app-only certificate auth.
- **Remote deploy pipeline** over WinRM/pywinrm: code ships by git pull + container restart on a volume-mounted compose stack - no image rebuilds, no RDP sessions, fully scriptable.
- **Fleet automation** across all 8 machines: WinRM hardening quirks solved per-machine (auth modes, token-filter policy, local-account nuances on Windows 11), captured in scripts and a full-infrastructure test suite that verifies every machine end to end.
- **Root-cause discipline:** the original outage was traced to a session-scoped container engine - documented, mitigated with the watchdog, and the reboot-survival tradeoffs written up honestly instead of papered over.

## Results

- Outage detection went from “whenever a human notices” to under 5 minutes, with zero alert spam by design.
- Full-fleet health verification runs as one script: 8 machines, every test green.
- No-console administration: every routine operation on the fleet is a script, not a session.

## Stack & Methods

**Stack:** PowerShell · WinRM / pywinrm · Hyper-V · Windows Server (AD domain) · Docker Compose · Microsoft Graph (app-only auth) · Tailscale mesh VPN · scheduled tasks