UPNIX
TECHNOLOGIES

INCIDENT REPORT

Ref UPX-IR-202607221927
Anonymized case study

SECURITY INCIDENT RESPONSE · DISCOVERED JULY 2026, REMEDIATED THROUGH OCTOBER 2026

Web-shell and backdoor campaign on a shared-hosting platform

WEB SHELLS | SUPPLY-CHAIN BACKDOOR | SHARED HOSTING | CMS / E-COMMERCE | ROGUE ADMINS | REMEDIATION AT SCALE

01 Executive summary

Between July and October 2026, the UPNIX team investigated and remediated a long-running web-shell and backdoor campaign across a client's shared-hosting fleet — hundreds of e-commerce shops and content-management sites spread over seven servers. The root cause was a backdoor baked into a cloned site template, compounded by a separate, actively self-healing CMS compromise that exploited an unpatched core vulnerability. The team mapped the full extent across the fleet, built a fully reversible clean-up pipeline, quarantined roughly 1,600 malicious files, removed several hundred rogue administrator accounts, and patched the vulnerable cores that were the live re-entry vector.

02 Environment

A multi-server shared-hosting platform (InterWorx control panel) hosting roughly 600 content-management installations and several hundred e-commerce shops for a client, distributed across seven servers.

03 Root cause

Two intertwined problems drove the campaign:

- **A template-baked backdoor.** A web-based file-manager/IDE component was present in a cloned e-commerce site template: around 300 shops carried it and about 217 held a backdoored configuration file with an HTTP-header-triggered `eval()` that granted arbitrary code execution. Timestamps showed it had been planted roughly twenty months earlier — built into the template rather than from a fresh break-in. Attackers used it to read each shop's database credentials, steal the mailbox passwords those shops had configured for outbound mail, and relay spam. The spam was only the visible symptom.
- **A separate, active CMS compromise.** A content-management campaign exploited an unpatched core vulnerability (CVE-2026-63030): self-healing scheduled tasks re-dropped shells every few minutes, machine-named rogue administrators were being created continuously, and a command-and-control beacon ran on fully-owned sites.

04 Campaign timeline

Nov 2024	Backdoor baked into the cloned site template; dormant.
May–Jul 26	Monetization begins: mailbox-credential theft and spam relay become visible.
22 Jul 26	The UPNIX team identifies the template backdoor as the root cause behind the spam — not independent password brute-force.
27 Jul 26	A fleet-wide re-scan shows the first clean-up was too narrow: a large pre-existing web-shell population remains, and one server is under an active content-management compromise.
1 Oct 26	Fleet-wide remediation completed and verified across all seven servers.

05 Findings

- Multiple web-shell families on the e-commerce servers: obfuscated single-file RCE droppers, a well-known PHP backdoor panel, an unauthenticated database manager, and shells uploaded through a file-manager plugin.
- A convenience admin extension that shipped a hard-coded credential — the mechanism by which the file-manager backdoor was re-deployed.
- On the content-management side: hundreds of machine-named rogue administrators, malicious scheduled tasks that re-dropped shells, a C2 beacon, and payload-bearing theme “changesets” stored in the database.

06 Remediation at scale (reversible pipeline)

Because the work touched hundreds of live client sites, the team built a repeatable, fully reversible pipeline: back up each site first (database, configuration and scheduled tasks); **quarantine** suspicious files (set to non-executable with a SHA-256 manifest, never delete) instead of removing them; replace vulnerable cores with patched releases of the same branch; strip injected include lines; remove rogue administrators by a narrow machine-name signature and reassign their content to the real owner; demote rogue self-registrations; clean malicious changesets, options, scheduled tasks and orphaned metadata; and rotate secrets where accounts were removed. Scope: roughly 600 installations scanned, 228 sites cleaned, about 1,600 files quarantined, several hundred rogue accounts removed, and vulnerable cores reduced from dozens to zero — confirmed by an independent re-scan. Every change left a backup and an audit trail for rollback.

07 Lessons learned

- **Audit vendor and template code.** A backdoor baked into a cloned template infects every site built from it, silently, for years.
- **Do not stop at the symptom.** A clean-up scoped only to the visible problem (here, spam) can leave the underlying intrusion intact; map the full extent before closing an incident.
- **Patch content-management cores promptly.** An unpatched core vulnerability was the live re-entry vector behind the continuous re-infection.
- **Remove convenience tools that ship hard-coded credentials.** They become a reliable re-entry mechanism.
- **Make large-scale remediation reversible.** Quarantine with a manifest, back up before every change, and keep an audit trail; at fleet scale this is the difference between recovery and new outages.
- **Constrain outbound traffic from site users** so a single compromised shop cannot relay spam or reach command-and-control infrastructure.

This report has been fully anonymized. All client identifiers — organization names, hostnames, domains, IP addresses, account names and storage locations — have been removed. Technical findings, methods and recommendations are retained unchanged so the case remains useful to other operators.

About UPNIX Technologies. The UPNIX team builds, monitors, protects and keeps running the infrastructure businesses depend on — security, 24/7 monitoring and operations for clients worldwide, alongside software and AI engineering.