CHECKLIST · IT ASSET READINESS

The Zero-Gap Asset Readiness Checklist

24 checks and two scorecards to know what you run before the next zero-day lands, and to switch IT asset tools without losing sight of it.

Oct 21
PHP2Shell fixes released, one day after details leaked early

8 days
From fix release to the CISA patch deadline for US federal agencies

~1.7M
Upper-bound estimate of domains on PHP branches with no official fixed release

Inside

00 What Black October taught IT teams
01 Know what you run
02 Track lifecycle and end-of-life
03 Respond in hours, not weeks
04 Switch tools without a gap
05 Score your readiness
06 Vulnerability fix scorecard
07 IT tool visibility scorecard

Who it's for

IT directors, IT operations and help desk leads, and IT asset managers at growing organizations where a small team covers a large estate.

Use it alone or with your security team. Tick each check you can prove today, then total your score on the last page.

© 2026 EZO. All rights reserved.

LESSONS FROM BLACK OCTOBER

The fix was ready in a day. Finding what needed it took longer.

In October 2026, two critical flaws in the PHP engine, known as PHP2Shell (CVE-2026-109618 and CVE-2026-109619), let attackers run code on servers without logging in. Fixes shipped on October 21. Within days, CISA confirmed active exploitation and gave US federal agencies until October 29 to patch. Some governments and organizations took public sites offline while they patched.

LESSON 1

The flaw sat below the application

Updating WordPress or a plugin did not help. The risk lived in the runtime underneath, which many inventories never record.

LESSON 2

Your version decided your options

PHP 8.2 to 8.5 got fixed releases. 7.4 to 8.1 got source-only backports. Older branches got nothing. End-of-life software turned a patch into a project.

LESSON 3

The response window shrank

AI models helped researchers turn a flaw that had been flagged years earlier into working exploits in days. A monthly inventory report can't keep up with that pace.

LESSON 4

Devices count, not just servers

Industrial vendors issued advisories for web interfaces built into cameras, access-control panels and control units, where firmware fixes are slow to arrive.



The same gap opens when you change tools

Your IT asset system is the map you reach for in a crisis. That map is at its weakest during a tool switch, and many teams switch under pressure they didn't plan for:

- A licensing or renewal change forces a decision on a deadline.
- A client, auditor or security team asks you to review the tools you run.
- The service desk moves to a new platform and the asset records need a new home.

Sections 01–03 keep you ready for the next zero-day. Section 04 makes sure a tool change doesn't leave you blind in the meantime.

1 Know what you run

You can't patch what you haven't mapped.

- [ ] **Every endpoint, server and VM reports into one inventory**
  Records come from an agent, MDM or network scan, not a spreadsheet someone updates by hand.
- [ ] **Software inventory captures runtimes and exact versions**
  Interpreters and frameworks such as PHP, Java, Python and Node.js are recorded with their version, not just a list of "installed apps".
- [ ] **Duplicate records are reconciled**
  When an agent, MDM and a scan all see the same device, it ends up as one record, matched by serial number or another unique ID.
- [ ] **Network-attached devices are in scope**
  Cameras, access-control panels, NAS units and printers are tracked with vendor, model and firmware version.
- [ ] **Discovery gaps are known and covered**
  You know which device types or networks your discovery can't reach, and how those assets are tracked instead.
- [ ] **Inventory is searchable by software name and version**
  One query returns every asset running a given component and version.

2 Track lifecycle and end-of-life

EOL software is where a patch turns into a project.

- [ ] **Every OS and runtime version has an end-of-support date**
  Dates are stored on the software record, so nobody has to look them up mid-incident.
- [ ] **A standing report flags EOL and soon-to-be-EOL software**
  It covers anything already unsupported or reaching end of support in the next six months.
- [ ] **Upgrades are scheduled against EOL dates**
  Example: PHP 8.2 security support ends December 31, 2026. Upgrade work is in the plan and the budget before that date.
- [ ] **Unused software is retired**
  Every install nobody uses is one more thing to patch. Usage data drives removal.
- [ ] **Vendor contracts and support terms are linked to assets**
  You know which vendor ships the firmware or patch, and when coverage runs out.
- [ ] **Unsanctioned software is discovered**
  Shadow IT is found and either brought under management or removed.

3 Respond in hours, not weeks

When a deadline is days away, ownership is the bottleneck.

- [ ] **Every asset has a named owner and business service**
  A vulnerable host leads straight to a person, not to a ticket queue.
- [ ] **Someone watches advisories for the software you run**
  CISA KEV and vendor advisories are checked against your inventory, not just read in the news.
- [ ] **Internet-facing assets are flagged**
  You can list every publicly reachable system before an advisory lands.
- [ ] **The emergency path is agreed in advance**
  You know who can approve taking a service offline, and an emergency change template is ready.
- [ ] **There's a plan for when no patch exists**
  For end-of-life software, the fallback is agreed: a configuration change, isolation, or taking the service offline.
- [ ] **You've run a drill in the last quarter**
  Take a recent critical CVE and time how long it takes to go from disclosure to a full list of affected assets and owners.

4 Switch tools without a gap

Replacing your help desk or ITAM tool shouldn't leave you blind in between.

- [ ] **The cutover date works back from your licence expiry**
  Build in buffer, so the new system is live before the old one switches off.
- [ ] **The full export is pulled and checked early**
  Assets, custom fields, contracts, licences, payment history and attachments. Confirm what the export actually includes.
- [ ] **Old history has a fallback**
  If some records won't move, arrange read-only access to the old tool for the period you need.
- [ ] **Discovery runs in parallel before cutover**
  Devices report into the new system, and counts match the old one, before you retire it.
- [ ] **Workarounds are listed, not copied over**
  Note every process that exists only because the old tool couldn't do something. Then design it properly.
- [ ] **A security review pack is ready for the new vendor**
  Cover what the agent can access, where data is hosted, certifications, SSO and MFA, and audit logs.

YOUR SCORE

Count your ticks in each section

01 · Know what you run	02 · Lifecycle & EOL	03 · Respond fast	04 · Switch without a gap
/ 6	/ 6	/ 6	/ 6

0–11 · Exposed
The next zero-day starts with a hunt for affected assets. Start with Section 01.

12–19 · Partial
The basics are there. Close lifecycle and ownership gaps before your next renewal or audit.

20–24 · Ready
You can find, prioritize and prove a fix fast. Keep running drills.

HOW ASSETSONAR HELPS

One connected record of what you own

Discovery and reconciliation

The ITAM Agent, MDM integrations such as Intune, Jamf and Kandji, and network discovery feed one inventory, with duplicate records merged by serial number.

Software and licences

Software normalization, licence and renewal tracking, and Shadow IT discovery in one place.

Workflow Automations

Routing, approvals and escalations run as automated workflows instead of manual hand-offs.

IT Graph

Devices, software, contracts, licences and tickets connected through one relationship graph.

Sources: PHP fixed releases and backports: Pentest-Tools.com, "PHP2Shell: what it is, how to detect, how to patch" (Oct 2026). CISA KEV listing and Oct 29 federal deadline: Nextgov/FCW and BleepingComputer (Oct 2026). Estimate of domains on end-of-life PHP (upper bound): TechnologyChecker.io (Oct 2026). Industrial device advisories: SecurityWeek (Oct 2026). PHP 8.2 end of support: php.net supported versions. AssetSonar Patch capabilities: AssetSonar Patch Management datasheet (2026).

DETECT → PRIORITIZE → DEPLOY → PROVE

How fast can you go from CVE to fixed?

Score each check: **Yes = 2**, **Partly = 1**, **No = 0**. The right-hand column shows how AssetSonar Patch handles each step.

Stage	Check	Score	How AssetSonar Patch does it
01 Detect	**Endpoints are scanned for known CVEs continuously**, not only when a new CVE makes the news.	○ Y ○ P ○ N	AssetSonar agents scan endpoints and map results to the NIST National Vulnerability Database.
	OS and third-party app vulnerabilities appear in one view.	○ Y ○ P ○ N	Open CVEs across Windows, Mac, Linux and third-party apps in one place.
02 Prioritize	**Priority uses business context, not CVSS alone.**	○ Y ○ P ○ N	Risk-based prioritization factors in asset ownership, licence status and business usage.
	Low-priority noise is filtered out, so the team works the most critical assets first.	○ Y ○ P ○ N	Focuses patching on the most critical assets and reduces noise from low-priority vulnerabilities.
03 Deploy	**Patches go out to Windows, Mac and Linux from one console.**	○ Y ○ P ○ N	Cross-platform patch deployment from a single platform.
	Emergency and scheduled patching both work, and failures are retried.	○ Y ○ P ○ N	Apply patches right away or schedule them, with built-in retry logic for failed patches.
04 Prove	**Patch SLAs and coverage are tracked.**	○ Y ○ P ○ N	SLA dashboards and patch coverage metrics.
	Audit evidence can be exported on demand.	○ Y ○ P ○ N	Exportable audit trails and audit-ready patch reports.

Your total

/ 16

- 🔴 **0–7 · Reactive** — Fixes start with a manual hunt. Start with Detect.
- 🟠 **8–12 · Managed** — Detection works. Prioritization and proof slow you down.
- 🟢 **13–16 · Ready** — Your patch process holds up under a deadline.

SELF-ASSESSMENT · ONE TOOL AT A TIME

Could you identify every affected IT tool?

Assess one IT tool or application at a time. Find the gaps that could slow an urgent software security review, even before a patch is available.

IT TOOL / APPLICATION

ASSESSMENT DATE

SERVICE OR APPLICATION OWNER

DEPLOYMENT / ENVIRONMENT

A Set the review priority

Check every statement that applies. These flags set the review order. They don't change the readiness score.

- ☐ Can execute commands or install software
- ☐ Has privileged access or handles sensitive data
- ☐ Operates across many managed devices
- ☐ Supports critical production services

B Answer the eight questions

Select one response per question: **Yes = 2 · Partly = 1 · No = 0**

#	Assessment question	Yes 2	Partly 1	No 0
01	Can you identify the managed devices where this tool is installed?	☐	☐	☐
02	Do you have installed-version details for the tool and any separately installed runtimes it uses?	☐	☐	☐
03	Can you compare discovered installations with the versions and deployments IT approved?	☐	☐	☐
04	Can you tell when each relevant device or software record was last updated?	☐	☐	☐
05	Can you link the tool to the devices and IT services that depend on it?	☐	☐	☐
06	Is there a named person or team accountable for reviewing and maintaining the deployment?	☐	☐	☐
07	Are relevant support deadlines and lifecycle details documented and accessible?	☐	☐	☐
08	After an update, can you verify the installed version on affected managed devices?	☐	☐	☐

SCORING RULE "Yes" means you can show it with current evidence, not just that a process exists. If you're unsure, select "Partly".

Turn your answers into a plan

Add your eight responses for a total out of 16. A higher score means stronger evidence for scoping a software advisory. It doesn't mean the tool is free of vulnerabilities.

My readiness score

/ 16

IMPACT FLAGS: ____ of 4 selected

Review high-impact tools first when their readiness scores are low. Impact flags show review priority, not a calculated risk rating.

C Interpret your score

0–6 · Limited visibility

Start with software discovery, version evidence and accountable ownership. An advisory may set off a manual search across devices.

7–11 · Partial readiness

Fix stale records, mismatched installations and undocumented dependencies before the next review.

12–16 · Strong foundation

Test a representative deployment and confirm owners can pull accurate records quickly. Keep the evidence current as software changes.

D Choose your next three actions

Start with the questions you answered "No", beginning with tools that have wide access or critical dependencies.

- [ ] **Identify devices that aren't covered, or are missing version evidence**
- [ ] **Assign an owner to investigate unapproved or unsupported installations**
- [ ] **Refresh records and confirm the fixed build after the next change**

See how AssetSonar helps IT teams track software and versions →

Close the detect-to-fix gap

AssetSonar Patch brings vulnerability detection, risk-based prioritization, cross-platform deployment and audit-ready reporting into one platform, so IT teams can patch smarter and prove progress faster.

Talk to an Expert

Scope note: This self-assessment measures visibility and response readiness, not CVSS severity or overall product security. AssetSonar does not guarantee complete discovery of bundled components, patch availability, engineering coverage or incident escalation.