WARATEK

Securing Oracle E-Business Suite
The 2026 Guide to EBS Threats, Best Practices, and Automated Runtime Protection

Waratek | July 2026

Table of Contents

Executive Summary

Oracle E-Business Suite (EBS) consolidates your organization's most critical operations (finance, HR, supply chain, payments) onto a single platform. That consolidation delivers enormous efficiency, but it also concentrates your crown jewels behind a single, highly complex attack surface.

The past year has proven the point: between August 2025 and June 2026, EBS was hit by two mass-exploited critical vulnerabilities (CVE-2025-61882 and CVE-2026-46817, both CVSS 9.8) and a third actively exploited flaw (CVE-2025-61884). The Cl0p extortion group used CVE-2025-61882 as a zero-day for nearly two months before a patch existed, stealing data from dozens of major organizations.

This guide combines Waratek's Oracle EBS security best-practices handbook with practical, up-to-date threat intelligence. It covers:

- The 2025–2026 EBS threat landscape, including the CVEs currently being exploited in the wild and what they teach us about patch-lag risk.
- The vulnerability classes that most commonly affect EBS (SQL injection, insecure deserialization, XSS/CSRF, information exposure, and insecure file handling), with concrete mitigations.
- Hardening best practices across configuration, authentication, network security, and monitoring, updated for Oracle's new monthly Critical Security Patch Update cadence, plus a consolidated checklist.
- Why perimeter and scanning tools (WAF, SAST, DAST) fall short for EBS, and how Waratek's runtime platform (RASP for production protection and the newly launched Waratek IAST for pre-production testing) closes the gap with zero false positives, no code changes, and no perceivable performance impact.

The core lesson of the last twelve months is simple: attackers now weaponize EBS flaws before or within days of a patch. Patching remains essential, but it is no longer sufficient on its own.

Organizations need protection that works at runtime, inside the application, from day zero.

The 2025–2026 EBS Threat Landscape

EBS has always attracted attackers. It is internet-reachable in many deployments, Java-based, and directly connected to money movement and sensitive personnel data. But the recent wave of exploitation is unprecedented in scale and speed.

CVE-2025-61882: The Cl0p Zero-Day (CVSS 9.8)

In October 2025, Oracle issued an emergency Security Alert for CVE-2025-61882, an unauthenticated remote code execution flaw in the BI Publisher Integration component of Oracle Concurrent Processing, affecting EBS 12.2.3 through 12.2.14. No credentials were required: an attacker with HTTP access could execute arbitrary code on the EBS server.

The timeline is the alarming part. Mandiant and Google Threat Intelligence found exploitation by the Cl0p extortion group as early as August 9, 2025, roughly eight weeks before Oracle's patch was published on October 4, 2025. The exploit chained five separate bugs to achieve pre-authentication RCE.

Victims received mass extortion emails claiming theft of sensitive EBS data; confirmed victims included large enterprises and universities such as the University of Pennsylvania, and Harvard and others were also listed by Cl0p. A working exploit had reportedly been offered for sale on dark-web forums as early as June 2025.

CVE-2025-61884: Unauthenticated SSRF in Oracle Configurator (CVSS 7.5)

Days after the 61882 alert, Oracle published a second out-of-cycle advisory for CVE-2025-61884, a server-side request forgery flaw in the Oracle Configurator Runtime UI (the /configurator/UiServlet endpoint), affecting EBS 12.2.3–12.2.14. It allows a remote, unauthenticated attacker to access sensitive configuration data.

CISA added it to the Known Exploited Vulnerabilities catalog after confirming in-the-wild use; the same endpoint had been abused in the July–August 2025 campaign that preceded the Cl0p attacks.

CVE-2026-46817: Oracle Payments File Transmission Flaw (CVSS 9.8)

The pattern repeated in 2026. Oracle fixed CVE-2026-46817 (an improper privilege management and authentication vulnerability in the File Transmission component of Oracle Payments, affecting EBS 12.2.3 through 12.2.15) in its May 28, 2026 Critical Security Patch Update, notably the very first release in Oracle's new monthly patching program (see Patch Management below). An unauthenticated attacker with HTTP access can send a crafted request to the /OA_HTML/ibytransmit endpoint and fully compromise Oracle Payments, the module that moves your money.

On June 27, 2026 (roughly six weeks after the patch and before any public proof-of-concept), honeypot sensors recorded the first in-the-wild exploitation. Real-world breaches followed, including a compromise of Nissan payroll data. Shadowserver tracks hundreds of EBS instances still exposed on the internet, nearly half of them in the United States and Europe.

Recent and Notable EBS CVEs at a Glance

CVE	Component	CVSS	Impact	Status
CVE-2026-46817	Oracle Payments: File Transmission (/OA_HTML/ibytransmit)	9.8	Unauthenticated takeover of Oracle Payments	Actively exploited since June 2026; patched in the May 2026 Critical Security Patch Update
CVE-2025-61882	Concurrent Processing: BI Publisher Integration	9.8	Unauthenticated remote code execution	Exploited as zero-day by Cl0p from Aug 2025; emergency patch Oct 2025

CVE-2025-61884	Configurator Runtime UI (/configurator/UiServlet)	7.5	Unauthenticated SSRF; sensitive data access	In CISA KEV; exploited in the wild
CVE-2022-21500	User Management: Manage Proxies	7.5	Insecure deserialization; mass PII exposure without credentials	Widely exploited 2022
CVE-2020-2586 / -2587 (“BigDebIT”)	General Ledger	9.9	XSS/CSRF enabling alteration of financial reports, full EBS takeover	Patched; historical
CVE-2019-2638	Consolidation Hierarchy Viewer (GL)	9.9	Access-control bypass; attacker-directed electronic fund transfers	Patched; historical
CVE-2017-3506	WebLogic (EBS stack)	7.4	OS command injection; leveraged by 8220 Gang for cryptomining	Still exploited on unpatched systems

Three lessons stand out.

- First, attackers are exploiting EBS flaws before patches exist (61882) or within weeks of release, before public PoCs (46817).
- Second, the targets are the modules that matter most: payments, general ledger, concurrent processing.
- Third, unauthenticated, network-reachable flaws keep recurring, so any EBS instance with internet-facing components must assume it is being scanned today.

Apply Oracle's quarterly Critical Patch Updates, monthly Critical Security Patch Updates, and out-of-cycle Security Alerts immediately, and pair them with runtime protection for the window when no patch exists.

Understanding Oracle EBS and Why It's Hard to Secure

Oracle EBS is a comprehensive, integrated suite of business applications (ERP, CRM, and SCM on one platform) used by over 21,000 organizations globally. Its interconnected modules all operate on shared, real-time data, which reduces errors and latency and scales with organizational growth.

Module	Primary Applications	Why It Matters (and Why Attackers Care)
Financial Management	General Ledger, Payables, Receivables, Fixed Assets, Payments	Manages financial transactions and reporting, the direct target of CVE-2019-2638, BigDebIT, and CVE-2026-46817
Human Capital Management	HR, Payroll, Self-Service HR	Centralizes employee PII and payroll, the data stolen in recent extortion campaigns
Supply Chain Management	Inventory, Order Management, Purchasing	Drives procurement and order-to-cash; disruption halts operations
Project Management	Project Costing, Billing, Resource Management	Controls project financials and invoicing
CRM	Sales, Service, Marketing	Holds customer data and interactions

The same qualities that make EBS valuable make it difficult to secure. Its many interplaying components and extensive customization options create an incredibly complex system.

Consolidating everything under one umbrella runs against the principle of least privilege: one exploit can reach many systems at once.

Each component, from Oracle Forms to Managed Servers, requires its own security measures, and common threats such as OS command injection, SQL injection, and session hijacking exploit the complex interactions between modules. Security and convenience will always be in tension; the goal of this guide is to keep the convenience while closing the gaps.

Common Vulnerability Classes in EBS

SQL Injection

SQL injection lets attackers inject malicious queries through input fields, leading to unauthorized data access, modification, or full database compromise. CVE-2019-2638 showed the worst case in EBS: attackers could bypass access controls in the Consolidation Hierarchy Viewer and redirect electronic fund transfers into their own accounts (severity 9.9/10). Mitigate with prepared statements/parameterized queries, ORM frameworks, and strict input validation and sanitization.

Insecure Deserialization

Java applications serialize data to move it between components, and EBS components exchange serialized data constantly. Because byte streams are practically unreadable in transit, attackers can hide malicious payloads that execute when deserialized, enabling arbitrary code execution, authentication bypass, or privilege escalation.

CVE-2022-21500 exploited exactly this in Oracle's Manage Proxies component, exposing users' PII with no username or password required. Mitigate by validating input before deserialization, using safe libraries, restricting deserialization to allow-listed classes, and disabling unsafe features such as object-creation hooks.

Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF)

XSS lets attackers inject scripts into pages viewed by other users, stealing session cookies or redirecting users, while CSRF tricks an authenticated user's browser into performing unintended actions such as changing account details or initiating transactions. The 2020 "BigDebIT" vulnerabilities (CVE-2020-2586 and CVE-2020-2587) in the General Ledger module allowed both, letting attackers alter financial reports or take over the EBS environment entirely.

Even a quiet alteration of closed-period financial records can trigger costly compliance failures. Mitigate with input validation and output encoding, a Content Security Policy, anti-CSRF tokens on all state-changing requests, SameSite cookies, and re-authentication for high-risk operations.

Insufficient Authentication and Authorization

Weak authentication gives attackers an easy window to gain access and escalate privileges. The essentials:

- **Strong password policy:** Enforce complex passwords, regular rotation, and no reuse; change every default password.
- **Multi-factor authentication:** Require a second factor (authenticator app, hardware token, or passkeys) for all users.
- **Session management:** Use session timeouts, automatic logout for inactive sessions, and session-activity monitoring to prevent hijacking.
- **Role-based access control:** Configure RBAC for least privilege; review roles regularly to prevent privilege creep.
- **Segregation of duties:** Split critical tasks (e.g., creating vs. approving purchase orders) to limit fraud.
- **Continuous auditing:** Audit access logs and permissions continuously, with automated alerts for anomalies.

Information Exposure (CWE-200)

Sensitive information leaks through detailed error messages (stack traces revealing file paths and query details), misconfigured permissions, and unencrypted transmission. Note that CVE-2025-61884 is fundamentally an information-exposure flaw: unauthenticated access to sensitive configuration data.

Use generic user-facing error messages with detailed errors logged only for authorized personnel, enforce strict access controls, and encrypt data at rest (AES-256) and in transit (TLS 1.2+), with keys stored in HSMs or a KMS.

Cleartext Transmission (CWE-319)

Data sent unencrypted can be intercepted via packet sniffing or man-in-the-middle attacks. Enforce HTTPS/TLS between clients and EBS servers, use SSH/SFTP for administration and file transfer, disable SSL 2.0/3.0 and TLS 1.0/1.1, and restrict cipher suites to strong options (AES, SHA-256). In EBS, set the ICX: Forms Launcher and Application Framework Agent profile options to HTTPS URLs.

Insecure File Upload and Handling (CWE-434)

Unvalidated uploads let attackers introduce malicious files. Whitelist approved file types, cap file sizes, and verify both MIME type and magic number (extensions are trivially spoofed; an .exe renamed .jpg passes extension checks). Sanitize HTML content with OWASP AntiSamy under a maintained policy file.

Configure EBS profile options: disable Document of Record attachments if unused, set a reasonable ICX: Session Timeout, and specify a secure text-attachment viewer. Encrypt sensitive attachments, restrict access via RBAC, enable auditing of upload/download activity, and store files in Oracle Secure Files rather than publicly accessible directories.

Hardening and Operating Your EBS Environment

Securing EBS is not just about responding to threats; it is about proactively minimizing your attack surface. The recent CVEs make one practice non-negotiable:

Patch Management: Oracle's New Monthly Cadence

In 2026 Oracle moved to a monthly security patching rhythm. The familiar quarterly, cumulative Critical Patch Updates (CPUs), released the third Tuesday of January, April, July, and October, are now complemented by monthly Critical Security Patch Updates (CSPUs): smaller, targeted collections of high-priority security fixes released on the third Tuesday of the remaining eight months (February, March, May, June, August, September, November, and December).

The first CSPU shipped on May 28, 2026, and it carried the fix for CVE-2026-46817, which attackers were exploiting within six weeks. Oracle publishes a pre-release announcement the Thursday before each CPU and CSPU.

- **Treat CSPUs with CPU-level urgency:** CSPUs are designed to be smaller and easier to apply with minimal disruption. Do not defer them to your next quarterly cycle. CVE-2026-46817 proves monthly fixes get weaponized on monthly timelines.
- **Plan for twelve patch events a year:** Replace quarterly patch windows with a standing monthly window aligned to Oracle's third-Tuesday schedule, using each Thursday pre-release announcement to scope the work in advance.
- **Watch for Security Alerts between releases:** Oracle issues out-of-cycle Security Alerts (such as those for CVE-2025-61882 and CVE-2025-61884) for flaws too critical to wait even a month. Apply these immediately.

- **Stay informed:** Subscribe to Oracle Security Alert notifications, and monitor CISA's Known Exploited Vulnerabilities catalog for EBS entries. The Cl0p exploit chain included bugs already fixed in the July 2025 CPU; organizations current on patches had a smaller blast radius.
- **Reduce exposure:** Inventory internet-facing EBS endpoints (e.g., /OA_HTML/, /configurator/) and remove any that are not strictly required.

Network Security

- Implement HTTP security headers: Strict-Transport-Security (HSTS), Content-Security-Policy, X-Content-Type-Options, X-Frame-Options.
- Deploy TLS with certificates from trusted CAs; audit cryptographic settings regularly and disable weak protocols and ciphers.

Monitoring and Incident Response

- Enable comprehensive logging (application, database, server) and centralize it with a SIEM or tools such as Splunk or ELK; set retention to meet regulatory requirements.
- Define alert criteria around key indicators of compromise (repeated failed logins, off-hours access to sensitive data, unusual configuration changes), and tune thresholds to avoid alert fatigue.
- Maintain a written incident response plan with roles, communication protocols, and escalation paths; run post-incident reviews and update the plan. Companies affected by the 2025 Cl0p campaign that had rehearsed plans contained the damage far faster.

Why Traditional Security Tools Fall Short for EBS

The manual practices above are essential, but they are time-consuming, complex, and prone to human error. The tools most organizations reach for to automate them each carry heavy drawbacks in an EBS context.

Web Application Firewalls (WAFs)

WAFs build a solid perimeter and stop plenty of commodity attacks, but they rely on tuned rules for known threats. They struggle with the unique configurations of EBS applications, generate high false-positive rates that block legitimate traffic, and offer no help against zero-days: a WAF rule for CVE-2025-61882 could not exist during the eight weeks Cl0p was exploiting it. Once an attacker is past the wall, the WAF is blind, and placing a WAF in front of every application is cost-prohibitive.

SAST and DAST Scanners

Static analysis scans code that never executes, producing high false-positive rates and vulnerability fatigue; it cannot see runtime behavior. Dynamic analysis crawls the application from outside as a black box, missing deep logic flaws and providing no code-level context for fixes. Using them effectively means buying and operating both, plus extensive testing time, and both still lack the contextual awareness to distinguish real, exploitable issues from noise. When scanners cry wolf too often, real alerts lose urgency; when they miss, attackers get free reign.

The Waratek Platform: Runtime Security for Oracle EBS

Waratek takes a fundamentally different approach: security enforced from inside the Java runtime itself, where the application's actual behavior (not signatures or source-code guesses) determines what is malicious. For EBS customers the platform now delivers two complementary capabilities: RASP for protecting production, and the newly launched Waratek IAST for finding vulnerabilities before production. Together they form a continuous "Detect-to-Protect" loop from development through production.

Waratek RASP: Protection in Production

Waratek's Runtime Application Self-Protection embeds security policies directly into your EBS applications' runtime environments.

Deployment is as simple as installing Waratek agents across EBS components (Oracle Forms, Managed Servers) with no code changes and no downtime. Key properties:

- **Zero false positives:** Because Waratek observes actual code execution rather than matching patterns, it distinguishes real attacks from legitimate traffic with 100% accuracy: no detective work for your team and no broken EBS functionality.
- **Real-time protection, known and unknown:** Instantly blocks SQL injection, XSS, CSRF, insecure deserialization, command injection, and related classes at runtime, including exploitation of unknown (zero-day) vulnerabilities, closing the patch-lag window that Cl0p exploited.
- **Safe deserialization:** For deserialization attacks, the platform deserializes bytecode in an isolated environment first, studies the effects, and sanitizes or blocks malicious payloads before they reach your application.
- **SQL injection defense:** SQL queries are analyzed and modified at runtime to neutralize injection, regardless of database vendor.
- **Flexible, granular rules:** Rules run in detect or protect mode with CEF logging, custom severities and messages, and the ability to ignore reviewed-safe payloads, precise enough for even the most customized EBS modules.
- **No performance penalty:** Waratek runs directly in EBS with no perceivable performance hit, and scales simply: spin up a new application, deploy an agent alongside it.
- **Audit-ready reporting:** Every security event is logged for reports to executives and compliance auditors, supporting SOC 2, GDPR, and CCPA obligations.

Waratek IAST Finds What Scanners Miss

Launched at JavaOne 2026, Waratek IAST (Interactive Application Security Testing) works from within the running application during functional testing.

It identifies real, exploitable vulnerabilities as code actually executes, the visibility SAST and DAST fundamentally lack.

- **Perfect OWASP Benchmark score:** 100% true positive rate, 0% false positive rate, and the maximum OWASP Benchmark score of 100, achieved with zero manual tuning or custom rulesets.
- **No source code required:** IAST analyzes compiled bytecode, so it works on EBS customizations and extensions without source access, and is indifferent to whether a human or an AI wrote the code.
- **Zero scan time, zero friction:** Runs inside existing CI/CD pipelines with continuous feedback during functional testing: no separate scan windows, no context switching for developers.
- **Instant, actionable findings:** Every finding arrives with full stack traceability and remediation guidance, so teams fix the vulnerable code path, not a symptom. Waratek reports this eliminates 30–40% of manual testing hours.
- **A safety net for AI-generated code:** As teams use AI to accelerate development (including custom EBS extensions), IAST validates AI-suggested libraries and patterns in your actual runtime, catching the subtle flaws and insecure defaults AI-generated code often introduces.

Paired with RASP, IAST closes the loop: vulnerabilities discovered during testing inform runtime protection policies, and attacks blocked in production inform what to test next.

How Waratek Maps to the Recent CVEs

Threat pattern (recent example)	Waratek coverage
Zero-day RCE via crafted requests (CVE-2025-61882)	Runtime behavioral protection blocks exploit primitives (command execution, malicious deserialization, illegitimate process forking) even when the CVE is unknown and unpatched

Unauthenticated SSRF / data exposure (CVE-2025-61884)	Context-aware policies restrict what application code can reach and return; anomalous outbound requests and data access are detected and blocked at runtime
Abuse of exposed endpoints in payment flows (CVE-2026-46817)	Granular rules on specific URIs (e.g., protect mode on payment endpoints) block unsafe payloads while CEF logs feed your SIEM
Injection and deserialization classes (CVE-2019-2638, CVE-2022-21500)	Core RASP protections neutralize SQLi and deserialization payloads automatically, with zero false positives

Oracle EBS Security Checklist

Area	Action	Done
Patching	Current on latest quarterly CPU, all monthly Critical Security Patch Updates, and all EBS Security Alerts (incl. CVE-2025-61882, CVE-2025-61884, CVE-2026-46817 fixes)	☐
Patching	Monthly patch window scheduled for Oracle's third-Tuesday CPU/CSPU cadence	☐
Patching	Subscribed to Oracle Security Alerts and monitoring CISA KEV	☐
Exposure	Internet-facing EBS endpoints inventoried and minimized	☐
Authentication	Default passwords changed; strong password policy enforced; MFA enabled	☐
Authorization	RBAC configured for least privilege; SoD policies in place;	☐

	permissions audited regularly	
Sessions	ICX: Session Timeout configured; inactive sessions auto-terminated	☐
Network	TLS 1.2+ everywhere; SSL and TLS 1.0/1.1 disabled; strong ciphers only; security headers deployed	☐
Data	Encryption at rest (AES-256) and in transit; keys in HSM/KMS	☐
Files	Upload whitelist, size limits, MIME/magic-number checks, AntiSamy sanitization, secure attachment profile options	☐
Monitoring	Centralized logging with SIEM; IoC-based alerting tuned; logs reviewed routinely	☐
Response	Incident response plan documented, communicated, and tested; post-incident review process in place	☐
Runtime protection	RASP agents deployed on EBS components for zero-day coverage	☐
Testing	IAST integrated in CI/CD for custom EBS code and extensions	☐

Get Started with Waratek

Securing Oracle EBS manually is challenging and resource-intensive, and the 2025–2026 exploitation wave shows the cost of gaps. Waratek automates your best practices, protects the patch-lag window, and now tests your code with the same runtime precision it uses to protect it.

Deployment is straightforward, integration requires no code changes, and protection begins immediately.

- Request a demo or speak with a security expert: sales@waratek.com

- Learn more: waratek.com (IAST, RASP, and IAST+RASP solution pages)

“An excellent security solution, very easy to implement, and instant protection for vulnerabilities. Waratek’s fantastic team supported us through the entire process.” (Verified User in Financial Services, G2)