29th December, 2025

MongoBleed vulnerability - extracting sensitive data from the MongoDB server memory without authentication.

Radware's Cyber Threat Intelligence (CTI) team is monitoring the active exploitation of CVE-2025-14847, nicknamed "MongoBleed." This critical vulnerability affects the zlib decompression logic in MongoDB, allowing unauthenticated attackers to extract sensitive data from server memory remotely [1].

Key Attack Insights:

- **Unauthenticated Data Exfiltration**: The vulnerability resides in the wire protocol compression layer, specifically within the zlib library integration. It allows attackers to trigger a memory leak by sending malformed, compressed network packets without requiring valid credentials or user interaction. [2]
- **Widespread Impact and Active Exploitation:** The flaw affects multiple major versions of MongoDB Server. Research indicates that 42% of cloud environments contain at least one vulnerable instance, and proof-of-concept (PoC) exploits are currently being used in the wild [2]
- **Distinct Behavioral Signatures**: Exploitation attempts leave a specific forensic footprint in server logs. Attacks are characterized by high-velocity connection bursts that lack standard client metadata events, distinguishing them from legitimate application traffic. [3]

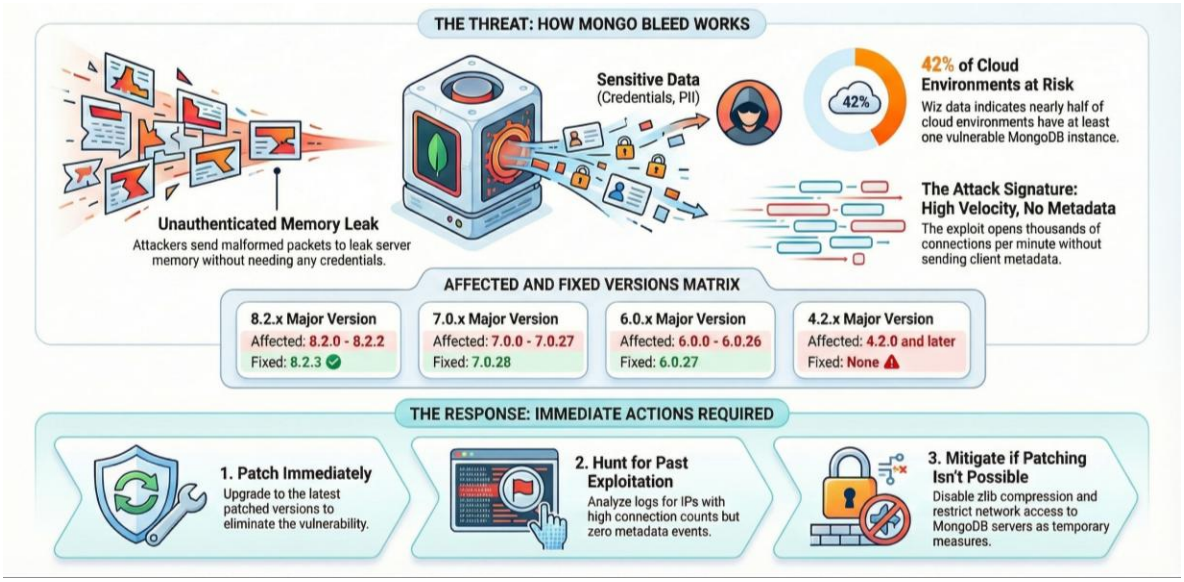

Figure 1: CVE-2025-14844 – a high-severity disclosure vulnerability in MongoDB's zlib decompression logic (source: Gemini)

Background

Disclosed in late December 2025, CVE-2025-14847 has been assigned a CVSS v4.0 severity score of 8.7 (High). The vulnerability, compared by researchers to the infamous "Heartbleed" bug, impacts the availability and confidentiality of data residing in MongoDB instances.[3]

Because the flaw exists in the pre-authentication phase of the connection handling, any internet-exposed MongoDB server with zlib compression enabled is immediately at risk. Attackers can exploit this to scrape credentials, session tokens, and Personally Identifiable Information (PII) directly from the heap memory.[1]

The Flaw: Zlib Decompression Mishandling

The root cause of MongoBleed is a logic error in MongoDB's handling of decompressed data lengths. Specifically, the vulnerability exists within the message_compressor_zlib.cpp file. [2]

When a client sends a compressed message, the server attempts to decompress it. In vulnerable versions, the code incorrectly returned the size of the allocated buffer (output.length()) rather than the actual length of the decompressed data. By sending a specially crafted, undersized packet, an attacker can force the server to respond with a buffer that includes the decompressed payload, along with adjacent, uninitialized heap memory. This "over-read" is returned to the attacker, thereby leaking any sensitive data that happens to reside in that memory space.[4]

Affected Versions The vulnerability spans years of MongoDB releases. The following versions are confirmed vulnerable: [1]

• **8.2.x:** 8.2.0 - 8.2.2

• **8.0.x:** 8.0.0 - 8.0.16

• **7.0.x:** 7.0.0 - 7.0.27

• **6.0.x:** 6.0.0 - 6.0.26

• **5.0.x:** 5.0.0 - 5.0.31

• **4.4.x:** 4.4.0 - 4.4.29

• **Legacy Versions:** All versions of 4.2, 4.0, and 3.6 are vulnerable and, as of this writing, have no patch available.

Use the Heading styles from the Word toolbar to set heading 1, 2, 3. Use ‘Normal’ to format copy pasted text in the document style.

Reasons for Concern

Radware’s CTI team identifies several factors that elevate the risk level of this vulnerability:

1. **Zero-Authentication Barrier:** Unlike many exploits that require prior access or valid credentials, this vulnerability allows any remote attacker to read server memory anonymously. This "pre-authentication" nature makes it trivial for threat actors to scrape API keys and passwords necessary for lateral movement.

2. **Legacy Debt and Unpatchable Systems:** While patches exist for newer versions, widely deployed End-of-Life (EOL) versions—specifically 3.6, 4.0, and 4.2—remain permanently vulnerable with no fix available. Organizations running these legacy systems must rely solely on mitigation strategies like disabling compression or network isolation.

3. **Detection Blind Spots:** The exploit is primarily detectable via MongoDB server logs, which many organizations do not forward to a central SIEM. Furthermore, identifying the attack requires complex logic—monitoring for "silent" connections that lack metadata and exhibit extreme velocity—which is challenging to implement in standard signature-based detection engines.

What to search for in the logs?

While patching is the primary defense, log analysis is critical for identifying past or ongoing exploitation. Logs can identify active exploitation through two specific anomalies:

1. **Absence of Client Metadata:** Legitimate MongoDB drivers (Node.js, PyMongo, etc.) automatically send a "client metadata" handshake immediately after connecting (Event ID 51800). The current MongoBleed exploit connects (Event ID 22943) and disconnects (Event ID 22944) without ever sending this metadata. A metadata rate of 0% across multiple connections is a high-fidelity indicator of attack.

2. **High-Velocity Bursts:** Legitimate traffic typically generates persistent connections or lower connection rates (e.g., single digits per minute). To effectively scrape memory, attackers must rapidly establish tens of thousands of connections. Forensic analysis of attack traffic has shown velocities exceeding 111,000 connections per minute, compared to legitimate production traffic of roughly 0.2 to 3.2 connections per minute.

Recommandations

Radware recommends organizations take immediate steps to secure their data infrastructure:

Patch Immediately: Upgrade to the fixed versions released by MongoDB: 8.2.3, 8.0.17, 7.0.28, 6.0.27, 5.0.32, or 4.4.30.

• **Disable Compression (Workaround):** If immediate patching is not feasible, or if running end-of-life versions (4.2 and older), disable zlib compression in the configuration. This can be done by omitting zlib from networkMessageCompressors or net.compression.compressors.

• **Network Segmentation:** Ensure MongoDB instances are not exposed to the public internet. Use firewalls and VPNs to restrict access solely to trusted application servers.

• **Forensic Review:** Utilize tools like Velociraptor or SIEM queries to hunt for the specific log patterns described above (High connection count + Zero metadata events) to determine if data was exfiltrated prior to patching.

Resources List

[1] Eric Capuano, "Hunting MongoBleed (CVE-2025-14847)," Eric's Substack, December 27, 2025. https://blog.ecapuano.com/p/hunting-mongobleed-cve-2025-14847

[2] Wiz Research Team, "MongoBleed (CVE-2025-14847) exploited in the wild," Wiz Blog, December 28, 2025. https://www.wiz.io/blog/mongobleed-cve-2025-14847-exploited-in-the-wild-mongodb

[3] National Vulnerability Database, "CVE-2025-14847 Detail," NIST, December 19, 2025. https://nvd.nist.gov/vuln/detail/CVE-2025-14847

[4] OX Research Team, "MongoDB Unauthenticated Attacker Sensitive Memory Leak," OX Security, December 24, 2025. https://www.ox.security/blog/attackers-could-exploit-zlib-to-exfiltrate-data-cve-2025-14847/

EFFECTIVE DDOS PROTECTION ESSENTIALS

Hybrid DDoS Protection – Use on-premises and **cloud DDoS protection** for real-time **DDoS attack prevention** that also addresses high-volume attacks and protects from pipe saturation

Behavioral-Based Detection – Quickly and accurately identify and block anomalies while allowing legitimate traffic through

Real-Time Signature Creation – Promptly protect against unknown threats and zero-day attacks.

Web DDOS Tsunami Protection – Automated immediate detection and mitigation of Web DDOS encrypted high RPS and morphing attacks

A Cybersecurity Emergency Response Plan – Turn to a dedicated emergency team of experts who have experience with Internet of Things security and handling IoT outbreaks

Intelligence on Active Threat Actors – High fidelity, correlated and analyzed data for preemptive protection against currently active known attackers

For further **network and application protection** measures, Radware urges companies to inspect and patch their network to defend against risks and threats.

EFFECTIVE WEB APPLICATION SECURITY ESSENTIALS

Full OWASP Top 10 coverage against defacements, injections, etc.

Low false positive rate using negative and positive security models for maximum accuracy

Auto-policy generation capabilities for the widest coverage with the lowest operational effort

Bot protection and device fingerprinting capabilities to overcome dynamic IP attacks and achieve improved bot detection and blocking

Securing APIs by filtering paths, understanding XML and JSON schemas for enforcement, and using activity tracking mechanisms to trace bots and guard internal resources

Flexible deployment options including on-premises, out-of-path, virtual or cloud-based

LEARN MORE AT RADWARE'S SECURITY RESEARCH CENTER

To know more about today's attack vector landscape, understand the business impact of cyberattacks, or learn more about emerging attack types and tools, visit Radware's **Security**

Research Center. Additionally, visit Radware's **Quarterly DDoS & Application Threat Analysis Center** for quarter-over-quarter analysis of DDoS and application attack activity based on data from Radware's cloud security services and threat intelligence.

THIS REPORT CONTAINS ONLY PUBLICLY AVAILABLE INFORMATION, WHICH IS PROVIDED FOR GENERAL INFORMATION PURPOSES ONLY. ALL INFORMATION IS PROVIDED "AS IS" WITHOUT ANY REPRESENTATION OR WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES THAT THIS REPORT IS ERROR-FREE OR ANY IMPLIED WARRANTIES REGARDING THE ACCURACY, VALIDITY, ADEQUACY, RELIABILITY, AVAILBILITY, COMPLETENESS, FITNESS FOR ANY PARTICULAR PURPOSE OR NON-INFRINGEMENT. USE OF THIS REPORT, IN WHOLE OR IN PART, IS AT USER'S SOLE RISK. RADWARE AND/OR ANYONE ON ITS BEHALF SPECIFICALLY DISCLAIMS ANY LIABILITY IN RELATION TO THIS REPORT, INCLUDING WITHOUT LIMITATION, FOR ANY DIRECT, SPECIAL, INDIREC, INCIDENTAL, CONSEQUENTIAL, OR EXAMPLARY DAMAGES, LOSSES AND EXPENSES ARISING FROM OR IN ANY WAY RELATED TO THIS REPORT, HOWEVER CAUSED, AND WHETHER BASED ON CONTRACT, TORT (INCLUDING NEGLIGENCE) OR OTHER THEORY OF LIABILITY, EVEN IF IT WAS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, LOSSES OR EXPENSES. **CHARTS USED OR REPRODUCED SHOULD BE CREDITED TO RADWARE**

©2025 Radware Ltd. All rights reserved. The Radware products and solutions mentioned in this document are protected by trademarks, patents and pending patent applications of Radware in the U.S. and other countries. For more details please see: https://www.radware.com/LegalNotice/. All other trademarks and names are property of their respective owners.