Hive Pro

Threat Level

Red

Hiveforce Labs

THREAT ADVISORY

VULNERABILITY REPORT

Java4Shell: OpenJDK Base64 Heap Overflow Risks Remote Code Execution

Date of Publication	Admiralty Code	TA Number
November 18, 2026	A1	TA2026343

Summary

First Seen: November 17, 2026
Affected Products: OpenJDK, Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition, Oracle JavaFX, Oracle OpenJDK, Amazon Corretto, Eclipse Temurin, Microsoft Build of OpenJDK, Azul Zulu, BellSoft Liberica, SAP SapMachine, Red Hat OpenJDK
Impact: Java4Shell (CVE-2026-114938) is a high-severity (CVSS 8.1) heap overflow in the HotSpot JVM's AVX-512 Base64 decoder, affecting OpenJDK 17.0.6 and later and its downstream builds on Intel Xeon and AMD Zen 4+ servers. Unauthenticated attackers can reach it through routine Base64 handling in logins, tokens, and file uploads to leak secrets such as private keys or execute code, and researchers report pre-auth compromise of Keycloak, Jira, Confluence, Artifactory, Jenkins, and PingFederate. Fixes were released on November 17, 2026. No in-the-wild exploitation has been reported, but public exploit code is already circulating on GitHub.

CVE

CVE	NAME	AFFECTED PRODUCT	ZERO-DAY	CISA KEV	PATCH
CVE-2026-114938	(Java4Shell) OpenJDK HotSpot x86-64 Base64 Intrinsic Out-of-Bounds Write Vulnerability	Oracle/OpenJDK Java SE, OpenJDK, GraalVM for JDK and downstream OpenJDK distributions	✗	✗	✓

Vulnerability Details

#1

Java4Shell is a heap memory corruption flaw in HotSpot's x86-64 Base64 decoding stub, a hand-written AVX-512 routine that the JIT compiler swaps in for java.util.Base64 decoding once the method is frequently used. When the decoder reaches a final chunk made up only of "=" padding, its output-length calculation underflows to a negative value, and the resulting write mask allows a full 64-byte store instead of a few bytes.

#2

A simple input of 64 "A" characters followed by "=" (128 on JDK 17) is enough to trigger it, writing 62 bytes past the end of the output buffer. The decoder still rejects the input with an IllegalArgumentException, but only after neighboring heap objects have already been overwritten, so catching the exception offers no protection.

#3

The flaw is reachable remotely and often without authentication through any feature that decodes attacker-supplied Base64, such as Basic Auth headers, tokens, and file uploads, and a Java client connecting to a malicious TLS server is also exposed. Exploitation requires the decoder to be JIT-compiled first, typically by flooding the target with valid requests, and only affects x86-64 CPUs with AVX-512 VBMI support, leaving ARM64 systems unaffected.

#4

The overwrite is largely fixed-value, but researchers report that through heap grooming it can corrupt adjacent array lengths and be chained into Heartbleed-style memory disclosure and ultimately remote code execution. Exploitation is application-specific and sensitive to network conditions, and failed attempts can crash the JVM.

#5

The fix, JDK-8393212, adds a small guard that routes short padded tails to the safe scalar decoder. The discoverers are withholding their product-specific exploit chains, but a public GitHub repository already offers a trigger, staged exploit primitives, and attack modules for Keycloak, Jenkins, Spring Boot, and Elasticsearch, which so far confirm the overflow but not reliable code execution.

Vulnerability

CVE ID	AFFECTED PRODUCTS	AFFECTED CPE	CWE ID
CVE-2026-114938	OpenJDK 17 (17.0.6 - Before 17.0.21.1), OpenJDK 21 (Before 21.0.13.1), OpenJDK 25 (Before 25.0.5.1), OpenJDK 27 (Before 27.0.1.1); Oracle Java SE (8u511b33, 8u511b33-perf, 11.0.33.0.3, 17.0.21.0.3, 21.0.13.0.3, 25.0.5.0.3, 27.0.1.0.3); Oracle GraalVM for JDK (17.0.21.0.3, 21.0.13.0.3); Oracle GraalVM Enterprise Edition (21.3.20.0.3); Oracle JavaFX (21.0.13, 25.0.5, 27.0.1); Oracle OpenJDK; Amazon Corretto; Eclipse Temurin; Microsoft Build of OpenJDK; Azul Zulu; BellSoft Liberica; SAP SapMachine; Red Hat OpenJDK (x86-64 with AVX-512 VBMI only)	cpe:2.3:a:oracle:openjdk:*:*:*:*:*:*:*:*:* cpe:2.3:a:oracle:java_se:*:*:*:*:*:*:*:*:* cpe:2.3:a:oracle:java_se:*:*:*:*:enterprise_performance:*:*:* cpe:2.3:a:oracle:graalvm_for_jdk:*:*:*:*:*:*:*:*:* cpe:2.3:a:oracle:graalvm:*:*:*:*:enterprise:*:*:* cpe:2.3:a:oracle:javafx:*:*:*:*:*:*:*:* cpe:2.3:a:amazon:corretto:*:*:*:*:*:*:*:* cpe:2.3:a:eclipse:temurin:*:*:*:*:*:*:*:* cpe:2.3:a:microsoft:build_of_openjdk:*:*:*:*:*:*:*:* cpe:2.3:a:azul:zulu:*:*:*:*:*:*:*:* cpe:2.3:a:bellsoft:liberica_jdk:*:*:*:*:*:*:*:* cpe:2.3:a:sap:sapmachine:*:*:*:*:*:*:*:* cpe:2.3:a:redhat:openjdk:*:*:*:*:*:*:*:*	CWE-122

Recommendations

Identify Exposed Hosts by CPU Capability: Exposure requires an x86-64 processor that supports avx512f, avx512bw, avx512vbmi, and bmi2. On Linux, run grep -m1 '^flags' /proc/cpuinfo | tr ' ' '\n' | grep -E '^(avx512f|avx512bw|avx512vbmi|bmi2)$' and treat hosts that return all four flags as in scope. Cloud instance types can change underlying CPU generations, so use this check to prioritize rather than to defer patching.

Restrict Network Exposure of Java Services: Limit access to affected services to trusted networks, place administrative and authentication endpoints behind VPN or zero-trust access, and restrict outbound TLS from Java applications to approved destinations, since connecting to a malicious server can also compromise a Java client.

Monitor for Trigger Patterns and JVM Instability: Review web, WAF, and application logs for Base64-bearing fields containing long runs of "A" followed by "=" (64 characters for JDK 21 and later, 128 for JDK 17) or malformed padding in Authorization: Basic headers, and alert on high-volume Basic Auth bursts from a single source, which mark the JIT warm-up phase. On hosts, investigate bursts of IllegalArgumentException from the Base64 decoder and hs_err_pid crash logs showing SIGSEGV in G1 GC or StubRoutines frames. Treat matches as leads rather than conclusive detections.

Apply the November 2026 Java Security Updates Immediately: Install the fixes released on November 17, 2026 through Oracle's November 2026 Critical Security Patch Update (Java SE patch document CPU435) and the OpenJDK Vulnerability Group advisory, moving to 17.0.21.1, 21.0.13.1, 25.0.5.1, 27.0.1.1 or later (8u513-perf for the Java SE 8 Performance Pack), and prioritize internet-facing systems and hosts running identity, CI/CD, artifact, and collaboration platforms. Restart every JVM process after upgrading, since running JVMs keep the vulnerable code in memory.

Disable the Vulnerable Intrinsic Where Patching Is Delayed: As a temporary workaround reported by Calif and independent researchers, and not published by Oracle or OpenJDK, start affected JVMs with -XX:-UseBASE64Intrinsics to disable the hardware-accelerated Base64 path, or apply it fleet-wide through JAVA_TOOL_OPTIONS or product-specific variables such as ES_JAVA_OPTS, JAVA_OPTS_APPEND for Keycloak, and JAVA_OPTS for Jenkins, then restart the affected processes. Alternatively, -XX:UseAVX=2 disables AVX-512 entirely, but apply it only on x86-64 hosts because OpenJDK on ARM does not recognize the flag and will fail to start. Remove the flag once the patched JDK is deployed.

Update Downstream JDK Distributions and Bundled Runtimes: Many applications ship with an embedded JRE or run on vendor builds such as Amazon Corretto, Eclipse Temurin, Microsoft Build of OpenJDK, Azul Zulu, BellSoft Liberica, SAP SapMachine, or Red Hat OpenJDK. Track each distribution's corresponding release, rebuild container images that pin a JDK base layer, and confirm with product vendors when updated bundles are available. Products such as Elasticsearch ship a bundled JDK that cannot be patched independently, so apply the interim workaround through their own JVM option variables until the vendor releases an update.

Rotate Secrets on High-Risk Systems: Because the flaw enables heap memory disclosure, rotate private keys, signing keys, API tokens, and service credentials held in memory by exposed internet-facing Java applications after patching, particularly on identity providers and CI/CD servers where leaked material grants broad downstream access.

Detection Rules

Sigma rules | CVE-2026-114938 Java4Shell exploitation indicators (1/2)

```yaml
title: Java4Shell - JVM Crash in GC (Post-Exploitation Indicator)
id: b2c3d4e5-f6a7-4b8c-9d0e-1f2a3b4c5d6e
status: experimental
description: Detects JVM crash logs indicative of heap corruption from Java4Shell
references:
    - https://github.com/ikow/java4shell-poc/blob/main/detection/java4shell.yaml
logsource:
    category: file_event
    product: linux
detection:
    selection:
        TargetFilename|endswith: ".log"
        TargetFilename|contains: "hs_err_pid"
    condition: selection
falsepositives:
    - JVM crashes caused by unrelated bugs, out-of-memory errors or native code
    - Crash logs created during testing or troubleshooting
level: medium
tags:
    - attack.impact
    - cve.2026.114938
```

```yaml
title: Java4Shell - Repeated Base64 Decode Exceptions
id: c3d4e5f6-a7b8-4c9d-0e1f-2a3b4c5d6e7f
name: java4shell_base64_decode_exception
status: experimental
description: >
    Detects repeated IllegalArgumentException from Base64 decoder,
    which occurs on every trigger attempt (exception thrown after corruption)
references:
    - https://github.com/ikow/java4shell-poc/blob/main/detection/java4shell.yaml
logsource:
    category: application
    product: java
detection:
    selection:
        Message|contains:
            - "IllegalArgumentException"
            - "Input byte array has wrong 4-byte ending unit"
    condition: selection
falsepositives:
    - Clients or integrations sending malformed or truncated Base64 values
    - Fuzzing or authorised vulnerability scanning
level: low
tags:
    - attack.execution
    - cve.2026.114938
---
title: Java4Shell - Repeated Base64 Decode Exceptions (Correlation)
id: 7c1f4a2e-3b5d-4e8f-9a6c-2d4b8e1f3a57
status: experimental
correlation:
    type: event_count
    rules:
        - java4shell_base64_decode_exception
    timespan: 5m
    condition:
        gt: 10
level: high
```

Detection Rules

Sigma rules | CVE-2026-114938 Java4Shell exploitation indicators (2/2)

```yaml
title: Java4Shell - SIGSEGV in G1 Garbage Collector
id: d4e5f6a7-b8c9-4d0e-1f2a-3b4c5d6e7f8a
status: experimental
description: >
    Detects JVM crash output whose problematic frame is in the G1
    garbage collector or StubRoutines, indicating heap corruption
references:
    - https://github.com/ikow/java4shell-poc/blob/main/detection/java4shell.yaml
logsource:
    product: linux
    service: syslog
detection:
    keywords:
        - "G1ParScanThreadState"
        - "StubRoutines"
    condition: keywords
falsepositives:
    - Unrelated JVM crashes (JIT bugs, native library faults, hardware errors)
    - Crash reports generated during load testing or debugging
level: critical
tags:
    - attack.impact
    - cve.2026.114938
```

Note: These Sigma rules are experimental. Test them in your environment before deploying, and review each rule's falsepositives notes when triaging a hit, since unrelated JVM crashes, malformed client input and application errors can also trigger them.

Potential MITRE ATT&CK TTPs

Tactic	Technique	Sub-technique
Reconnaissance	T1595: Active Scanning	T1595.002: Vulnerability Scanning
Resource Development	T1588: Obtain Capabilities	T1588.005: Exploits
Initial Access	T1190: Exploit Public-Facing Application	
Execution	T1203: Exploitation for Client Execution	
Credential Access	T1212: Exploitation for Credential Access	
Impact	T1499: Endpoint Denial of Service	T1499.004: Application or System Exploitation

Patch Link

https://www.oracle.com/security-alerts/cspunov2026.html

References

https://www.oracle.com/security-alerts/cspunov2026.html

https://openjdk.org/groups/vulnerability/advisories/2026-11-17

https://calif.io/java4shell/

https://www.cisa.gov/news-events/alerts/2026/11/17/critical-java-vulnerability-requires-immediate-patching

https://github.com/ikow/java4shell-poc

https://www.vulncheck.com/blog/java4shell

What Next?

At **Hive Pro**, it is our mission to detect the most likely threats to your organization and to help you prevent them from happening.

Book a Demo of **HivePro**.

REPORT GENERATED ON

November 18, 2026 • 07:00 AM

© 2026 All Rights are Reserved by Hive Pro

More at www.hivepro.com