INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-20 06:53 UTC

CVE-2026-35258: Vulnerability in the WebLogic Server product of Oracle Fusion Middleware (component: Console). Supp...

CVE VULNERABILITY | HIGH | CVSS 8.7

SCC Item ID	SCC-CVE-2026-0338
Type	CVE Vulnerability
CVE ID	CVE-2026-35258
Severity	HIGH
CVSS Base Score	8.7
EPSS Score	0.0033 (24th percentile)
Affected Products	Oracle WebLogic Server 14.1.2.0.0 and 15.1.1.0.0 (Fusion Middleware, Console component)
Published	2026-06-17T10:40:17.287
Discovery Source	Nvd

Executive Summary

A high-severity vulnerability (CVE-2026-35258, CVSS 8.7) has been identified in the Console component of Oracle WebLogic Server versions 14.1.2.0.0 and 15.1.1.0.0. A low-privileged attacker with network access can exploit an open redirect flaw to steal administrator credentials, which could then be used to gain unauthorized read and write access to all data accessible by WebLogic Server. Organizations running these specific versions in internet-facing or hybrid environments should prioritize patching and access controls immediately.

Technical Analysis

CVE-2026-35258 affects Oracle WebLogic Server 14.1.2.0.0 and 15.1.1.0.0 (Fusion Middleware, Console component). The vulnerability is classified as CWE-601 (URL Redirection to Untrusted Site / Open Redirect). CVSS 3.1 Base Score: 8.7; Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:H/A:N. Attack complexity is low; the attacker requires only low privileges and network access via HTTPS, but exploitation requires user interaction from a person other than the attacker. Successful exploitation results in a scope change, impacts extend beyond the Console component itself. Confidentiality and Integrity are both rated High; Availability is not impacted. The open redirect flaw can be weaponized for credential harvesting, session token theft, or chained phishing attacks targeting WebLogic Console administrators. MITRE ATT&CK techniques associated with this

vulnerability include T1204.001 (User Execution: Malicious Link), T1566.002 (Phishing: Spearphishing Link), and T1078 (Valid Accounts). EPSS score is 0.00326 (24th percentile) as of 2026-03-04, indicating low observed exploitation in the wild at this time. EPSS scores are dynamic; check the latest EPSS data before making prioritization decisions. The vulnerability is not currently listed in the CISA Known Exploited Vulnerabilities catalog. Oracle has not yet published an independent CVSS score; the base score of 8.7 is from NVD. Source authority: NVD (T1).

Action Checklist

1. Step 1: Containment, Identify all instances of Oracle WebLogic Server 14.1.2.0.0 and 15.1.1.0.0 in your environment. Restrict access to the WebLogic Administration Console to trusted internal IP ranges only; block direct internet exposure of the Console port (default 7001/7002) at the firewall or load balancer. Apply WAF rules to inspect and block HTTPS requests containing open redirect patterns (e.g., parameters with external URLs) targeting WebLogic Console endpoints. Reference: NIST AC-17 (Remote Access), CIS 4.4 (Implement and Manage a Firewall on Servers).
2. Step 2: Detection, Review HTTPS access logs on WebLogic servers for Console endpoint requests containing redirect parameters pointing to external or unexpected domains. Correlate against AU-2 event logging for unusual low-privileged account activity on the Console. Search for T1566.002 indicators: inbound spearphishing emails referencing WebLogic Console URLs. Monitor for T1078 indicators: account logins from unexpected sources or at unusual hours. Enable and review WebLogic Server audit logs for Console authentication events. Reference: NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 8.2 (Collect Audit Logs).
3. Step 3: Eradication, Apply the Oracle patch for CVE-2026-35258 as issued in the relevant Oracle Critical Patch Update (CPU) for Oracle Fusion Middleware. Upgrade or patch WebLogic Server 14.1.2.0.0 and 15.1.1.0.0 to the vendor-specified remediated version. Consult the Oracle CPU advisory for exact patch IDs and version targets. Verify patch integrity before deployment. Reference: NIST CM-3 (Change Management), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management).
4. Step 4: Recovery, After patching, validate WebLogic Console access controls are enforced and no unauthorized accounts were created during the exposure window (NIST AC-2, Account Management). Rotate credentials for all accounts with Console access as a precaution, prioritizing administrative accounts. Review audit logs for the period prior to patching for signs of exploitation (redirected sessions, credential use from unexpected IPs). Re-enable Console access for legitimate users only after confirming patch success and access control review. Reference: NIST AC-2, NIST IA-4 (Identifier Management), NIST AU-6 (Audit Record Review, Analysis, and Reporting).
5. Step 5: Post-Incident, Document whether the Console was internet-accessible and whether MFA was enforced for Console logins. Implement MFA on all administrative access paths to WebLogic Console (CIS 6.5, Require MFA for Administrative Access; NIST IA-2, Authentication). Review user account inventory for dormant or over-privileged accounts (CIS 5.3, Disable Dormant Accounts; CIS 5.4, Restrict Administrator Privileges to Dedicated Administrator Accounts). Evaluate whether outbound URL filtering and WAF redirect detection are in place to reduce open redirect attack surface going forward (NIST AC-4, Information Flow Enforcement; CIS 4.4, Implement and Manage a Firewall on Servers).

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to CISO and legal/privacy counsel if WebLogic Console logs show successful redirect-based session hijacking (HTTP 302 responses to external domains from authenticated Console sessions) or if any account password hashes or session tokens are confirmed stolen, as credential theft via this open redirect may trigger breach notification obligations under applicable data protection regulations (e.g., GDPR 72-hour notification, US state breach laws).
Recovery Notes	After patching to the Oracle CPU-remediated version and completing credential rotation, monitor WebLogic Console access logs and `auditlog.xml` for a minimum of 30 days for authentication attempts using the pre-rotation credentials or logins from IPs identified during the exposure window analysis, as threat actors who harvested credentials via the open redirect may attempt delayed use. Validate that firewall rules restricting Console ports 7001/7002 remain in place post-recovery by scheduling a weekly `netstat` or firewall rule audit, since WebLogic upgrades and domain reconfigurations can inadvertently re-expose the Console port. Confirm that the patched WebLogic version no longer processes open redirect parameters by performing a controlled test: issue a crafted GET request to `/console/login/LoginForm.jsp?redirectUrl=https://external-test-domain.example` from an internal host and verify the server does not return an HTTP 302 to the external destination.
Forensic Artifacts	WebLogic Server access log (`$DOMAIN_HOME/servers//logs/-access.log`): Contains HTTP request records for Console endpoint URIs — search for GET/POST requests to `/console/` paths with query parameters containing `http://` or `https://` pointing to external domains, which are the direct fingerprint of CVE-2026-35258 open redirect exploitation attempts. \| WebLogic Security Audit Log (`$DOMAIN_HOME/servers//logs/auditlog.xml`): Contains `` authentication events with username, source IP, and timestamp for all Console logins — identify low-privileged accounts authenticating to the Console from external IPs or at anomalous hours, consistent with harvested credential reuse following open redirect phishing. | JVM heap dump of the WebLogic server process (captured pre-patch via `jmap -dump:format=b,file=/tmp/wl_heap.hprof `): May contain in-memory HTTP session objects, including session tokens, redirect parameter values, and authenticated user context from active or recently active Console sessions — analyzable with Eclipse Memory Analyzer (MAT, free) to recover attacker-controlled redirect destinations injected into session state. | Network packet capture on Console ports 7001/7002 (captured with `tcpdump -i -w /tmp/wl_console.pcap 'tcp port 7001 or tcp port 7002'`): Preserves the full HTTP exchange of any open redirect exploitation in-progress, including the crafted redirect parameter value, the victim's session cookie transmitted to the attacker-controlled domain, and the source IP of the attacker — critical volatile evidence lost once containment blocks traffic. | WebLogic domain `config.xml` (`$DOMAIN_HOME/config/config.xml`) and `DefaultAuthenticatorInit.ldift` (WebLogic embedded LDAP user store): Documents the account roster, role assignments, and Console access group memberships at the time of potential exploitation — compare against a known-good baseline to detect unauthorized account creation or privilege escalation by an attacker who gained write access via the CVSS 8.7 read/write impact scope of CVE-2026-35258.

Per-Action IR Details

Step 1: Containment — Identify all instances of Oracle WebLogic Server 14.1.2.0.0 and 15.1.1.0.0 in your environment. Restrict access to the WebLogic Administration Console to trusted internal IP ranges only; block direct internet exposure of the Console port (default 7001/7002) at the firewall or load balancer. Apply WAF rules to inspect and block HTTPS requests containing open redirect patterns (e.g., parameters with

external URLs) targeting WebLogic Console endpoints. Reference: NIST AC-17 (Remote Access), CIS 4.4 (Implement and Manage a Firewall on Servers).

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy

Controls: NIST AC-17 (Remote Access), NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers)

Compensating: Run `netstat -tlnp | grep -E '7001|7002'` on each Linux WebLogic host (or `netstat -ano | findstr '7001'` on Windows) to confirm Console port exposure. Apply host-based firewall rules immediately via `iptables -I INPUT -p tcp --dport 7001 -s -j ACCEPT && iptables -I INPUT -p tcp --dport 7001 -j DROP`. For WAF-less environments, deploy ModSecurity with a rule matching `REQUEST_URI` containing open redirect patterns such as `//`, `@`, or `http[s]?://` in Console path parameters (e.g., `/console/login/LoginForm.jsp?redirectUrl=`). Use `grep -E 'redirectUrl=https?://[^/]' /u01/oracle/middleware/user_projects/domains/*/servers/*/logs/access.log` to immediately identify prior exploitation attempts.

Evidence: Before blocking Console ports or applying firewall rules, capture active network connections to WebLogic Console ports: run `ss -tnp sport = :7001 or sport = :7002` (Linux) or `Get-NetTCPConnection -LocalPort 7001,7002 | Where-Object State -eq Established` (Windows) and record all established sessions with remote IPs. Also collect `netstat -ano` output and the current WebLogic server access log tail (`/u01/oracle/middleware/user_projects/domains//servers//logs/-access.log`) to preserve evidence of any open redirect requests in-flight before traffic is blocked.

Step 2: Detection — Review HTTPS access logs on WebLogic servers for Console endpoint requests containing redirect parameters pointing to external or unexpected domains. Correlate against AU-2 event logging for unusual low-privileged account activity on the Console. Search for T1566.002 indicators: inbound spearphishing emails referencing WebLogic Console URLs. Monitor for T1078 indicators: account logins from unexpected sources or at unusual hours. Enable and review WebLogic Server audit logs for Console authentication events. Reference: NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 8.2 (Collect Audit Logs).

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis

Controls: NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST AU-2 (Event Logging), NIST AU-12 (Audit Record Generation), CIS 8.2 (Collect Audit Logs)

Compensating: Parse WebLogic access logs for open redirect exploitation attempts with: `grep -iE 'GET /console.*[?&](redirectUrl|redirect|url|next|return)=https?://' /u01/oracle/middleware/user_projects/domains/*/servers/*/logs/*-access.log`. Enable WebLogic Audit Provider via WLST: `edit(); startEdit(); cmo.setAuditSeverity('DEBUG'); save(); activate()`. Query WebLogic security audit log at `$DOMAIN_HOME/servers//logs/auditlog.xml` for `` authentication events tied to the Console application context. Use Sigma rule `title: WebLogic Console Open Redirect Abuse` matching `cs-uri-stem contains '/console' AND cs-uri-query contains 'http'` against IIS/W3C-format access logs if parsed with chainsaw or grep.

Evidence: Capture the full WebLogic server access log for the Console component (`*-access.log`) covering the maximum retention window before any log rotation occurs — these contain the URI parameters showing redirect destinations. Collect WebLogic audit log (`auditlog.xml`) for authentication success/failure events on Console endpoints tied to low-privileged accounts. Preserve the WebLogic domain configuration file (`config.xml`) to document which accounts held Console roles at the time of potential exploitation. If the host is live, extract current authenticated session tokens from WebLogic memory using a full RAM acquisition (WinPmem on Windows, LiME kernel module on Linux) before any session termination or server restart.

Step 3: Eradication — Apply the Oracle patch for CVE-2026-35258 as issued in the relevant Oracle Critical Patch Update (CPU) for Oracle Fusion Middleware. Upgrade or patch WebLogic Server 14.1.2.0.0 and 15.1.1.0.0 to the vendor-specified remediated version. Consult the Oracle CPU advisory for exact patch IDs and version targets. Verify patch integrity before deployment. Reference: NIST SI-2 (no mapped control from

provided knowledge base — see Oracle CPU advisory directly), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management).

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: Before patching, verify the Oracle CPU patch bundle integrity using the SHA-256 hash published in the Oracle CPU advisory: `sha256sum .zip` and compare against the Oracle-published checksum. Apply the patch using Oracle's OPatch utility: `$ORACLE_HOME/OPatch/opatch apply ` and confirm with `opatch lspatches | grep `. For environments without automated patch management, script pre/post validation: capture WebLogic version pre-patch via `java -cp $ORACLE_HOME/wlserver/server/lib/weblogic.jar weblogic.version` and re-run post-patch to confirm version increment to the remediated build.

Evidence: Before applying the Oracle CPU patch, which will alter WebLogic binaries and restart the server (destroying live state), perform a full memory acquisition of the running WebLogic JVM process: capture heap dump via `jmap -dump:format=b,file=/tmp/wl_heap_pre_patch.hprof ` to preserve any in-memory evidence of exploit payloads or injected session data. Archive the complete WebLogic server logs directory (`$DOMAIN_HOME/servers//logs/`) and the Console application deployment directory (`$DOMAIN_HOME/autodeploy/` and `$MW_HOME/wlserver/server/lib/console*.war` checksums) before patching overwrites them. Record the pre-patch `opatch lspatches` output as baseline evidence.

Step 4: Recovery — After patching, validate WebLogic Console access controls are enforced and no unauthorized accounts were created during the exposure window (NIST AC-2, Account Management). Rotate credentials for all accounts with Console access as a precaution, prioritizing administrative accounts. Review audit logs for the period prior to patching for signs of exploitation (redirected sessions, credential use from unexpected IPs). Re-enable Console access for legitimate users only after confirming patch success and access control review. Reference: NIST AC-2, D3-CRO (Credential Rotation), D3-LAM (Local Account Monitoring).

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST AC-2 (Account Management), NIST AC-3 (Access Enforcement), CIS 5.1 (Establish and Maintain an Inventory of Accounts), CIS 6.2 (Establish an Access Revoking Process)

Compensating: Enumerate all WebLogic Console user accounts via WLST: `connect('weblogic','','t3://localhost:7001'); users = cmo.getRealm().lookupAuthenticationProvider('DefaultAuthenticator').listUsers('*',0)` and compare against your pre-incident account inventory to identify any accounts created during the exposure window. Rotate WebLogic admin credentials by navigating to Security Realms → myrealm → Users and Groups in the Console (post-patch), or via WLST: `cmo.changeUserPassword('weblogic','','')`. After credential rotation, search archived access logs for the exposure window using: `grep -E '|' $DOMAIN_HOME/servers/*/logs/*-access.log` to confirm whether stolen credentials were used laterally.

Evidence: Before rotating credentials — which invalidates all active WebLogic sessions and removes the ability to correlate session tokens to attacker activity — export the current active session table from WebLogic Server Runtime MBean: `serverRuntime = getMBeanServerConnection(); serverRuntime.getAttribute(new ObjectName('com.bea:Name=,Type=ServerRuntime'), 'OpenSocketsCurrentCount')` and capture active HTTP sessions via WebLogic Diagnostic Framework (WLDF). Preserve the complete WebLogic security audit log (`auditlog.xml`) covering the full exposure window (from the earliest vulnerable version deployment date to patch application) before any credential changes alter the audit context. Document all account names, last-login timestamps, and source IPs from the WebLogic user store before rotation.

Step 5: Post-Incident — Document whether the Console was internet-accessible and whether MFA was enforced for Console logins. Implement MFA on all administrative access paths to WebLogic Console (CIS 6.5, Require MFA for Administrative Access; D3-MFA, Multi-factor Authentication). Review user account

inventory for dormant or over-privileged accounts (CIS 5.3, Disable Dormant Accounts; CIS 5.4, Restrict Administrator Privileges to Dedicated Administrator Accounts). Evaluate whether outbound URL filtering and WAF redirect detection are in place to reduce open redirect attack surface going forward (NIST AC-4, Information Flow Enforcement; D3-PBWSAM, Proxy-based Web Server Access Mediation).

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity

Controls: CIS 6.5 (Require MFA for Administrative Access), CIS 5.3 (Disable Dormant Accounts), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts), NIST AC-4 (Information Flow Enforcement), NIST AC-6 (Least Privilege)

Compensating: Document internet exposure status by reviewing firewall ruleset history and load balancer configuration for WebLogic Console ports 7001/7002 — use `iptables -L -n --line-numbers` (Linux) or Windows Firewall logs at `%SystemRoot%\System32\LogFiles\Firewall\pfirewall.log` to reconstruct the exposure window timeline. Integrate WebLogic with an external Identity Provider supporting TOTP MFA (e.g., Keycloak with OTP, which is free and open-source) via WebLogic's SAML 2.0 or OIDC federation support to enforce MFA on Console logins. Disable dormant WebLogic Console accounts identified in Step 4 via WLST and document the remediation in the post-incident report as a lessons-learned finding tied specifically to the low-privilege exploitation path enabled by CVE-2026-35258.

Evidence: For the lessons-learned record, preserve the complete firewall and load balancer configuration snapshots showing the Console port exposure state at the time of CVE disclosure — this establishes the blast radius for any required breach notification assessment. Retain all WebLogic audit logs, access logs, and account inventory exports collected during Steps 1-4 in an evidence archive for a minimum retention period consistent with your regulatory obligations, as the open redirect flaw in CVE-2026-35258 may have facilitated credential theft that requires notification under applicable data protection regulations.

Detection Guidance

Focus detection on WebLogic Server HTTPS access logs and the WebLogic Audit Service. Query for Console endpoint requests (e.g., /console/ paths) where query parameters or Location headers contain fully qualified external domain names not belonging to your organization, this is the primary open redirect indicator for CWE-601. Look for T1204.001 and T1566.002 patterns: inbound email or web traffic delivering links to your WebLogic Console URL followed by a redirect parameter pointing to an attacker-controlled domain. Monitor for T1078 activity: low-privileged Console account logins from IP addresses not in your approved admin access list (NIST AU-6, AU-3). Use NIST AU-8 (Time Stamps) to ensure log timestamps are reliable for correlation. CIS 8.2 (Collect Audit Logs) requires audit logging to be enabled across enterprise assets, confirm WebLogic audit logging is active before declaring no evidence of exploitation. Behavioral indicator: a legitimate user session that authenticates to the Console but immediately navigates externally, or an administrative action performed from an IP that has no prior login history. No public IOCs (IPs, hashes, domains) are available in the provided source data for this CVE at this time.

Framework Mappings

MITRE-ATTACK

- **T1204.001** — Malicious Link
- **T1566.002** — Spearphishing Link
- **T1078** — Valid Accounts

NIST-800-53R5

- **AT-2** — Literacy Training and Awareness
- **SC-7** — Boundary Protection
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-8** — Spam Protection
- **AC-2** — Account Management
- **AC-6** — Least Privilege
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1204.001	Malicious Link	Execution
T1566.002	Spearphishing Link	Initial-Access
T1078	Valid Accounts	Defense-Evasion

Sources

Source	URL	Tier
nvd	https://nvd.nist.gov/vuln/detail/CVE-2026-35258	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35259	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35262	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35263	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35265	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35267	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35268	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35269	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-35270	T1
CVE-2026-35258 in WebLogic Server	https://vuldb.com/cve/CVE-2026-35258	T3

DISCLAIMER

Generated 2026-06-20 06:53 UTC by TJS Security Command Center