INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-07-02 07:14 UTC

CVE-2026-53309: Off-by-One Error in Linux Kernel ocfs2/dlm dlm_match_regions() Affecting Azure Linux 3.0

CVE VULNERABILITY | **CRITICAL** | CVSS 9.8

SCC Item ID	SCC-CVE-2026-0381
Type	CVE Vulnerability
CVE ID	CVE-2026-53309
Severity	CRITICAL
CVSS Base Score	9.8
EPSS Score	0.0040 (32th percentile)
Affected Products	Microsoft azl3 kernel 6.6.139.1-1 on Azure Linux 3.0
Published	2026-07-01T14:48:24
Discovery Source	Msrc Patch Tuesday

Executive Summary

Microsoft disclosed CVE-2026-53309, a critical off-by-one error (CVSS 9.8) in the Linux kernel's OCFS2 distributed lock manager, affecting the Azure Linux 3.0 kernel package (azl3 kernel 6.6.139.1-1) as part of the June 2026 Patch Tuesday cycle. The flaw resides in boundary comparison logic within the dlm_match_regions() function and, according to source material, could potentially enable out-of-bounds memory access leading to privilege escalation or remote code execution depending on how region data is supplied. Organizations running Azure Linux 3.0 workloads should treat this as a priority patching event given the critical severity rating and the kernel-level nature of the flaw.

Technical Analysis

CVE-2026-53309 is a CWE-193 (Off-by-One Error) in the dlm_match_regions() function within the Linux kernel's OCFS2 DLM subsystem, specifically affecting Microsoft's azl3 kernel package version 6.6.139.1-1 on Azure Linux 3.0. The flaw is located in boundary comparison logic during region matching operations; an off-by-one condition in this context can produce out-of-bounds memory access. Per the source description, potential consequences include privilege escalation or remote code execution, though the precise attack vector and full exploitability details require confirmation from NVD and MSRC advisories; consult those sources directly for the most current technical analysis. CVSS base score is 9.8 (Critical); EPSS score is 0.00404 (32.3rd percentile),

indicating low observed exploitation probability at time of reporting. The vulnerability is not listed in CISA KEV as of the configuration date. MITRE ATT&CK technique T1068 (Exploitation for Privilege Escalation) is the mapped technique. No threat actor attribution is established. Authoritative sources: MSRC Update Guide (https://msrc.microsoft.com/update-guide/vulnerability/CVE-2026-53309), NVD (https://nvd.nist.gov/vuln/detail/CVE-2026-53309), CVE Record (https://www.cve.org/CVERecord?id=CVE-2026-53309).

Action Checklist

1. Step 1: Containment, Identify all Azure Linux 3.0 hosts running azl3 kernel 6.6.139.1-1 using your asset inventory. Isolate or restrict network access to any OCFS2/DLM-enabled cluster nodes until the patch is confirmed applied. Reference MSRC Update Guide at https://msrc.microsoft.com/update-guide/vulnerability/CVE-2026-53309 for Microsoft's specific guidance. (Supports NIST AC-4, Information Flow Enforcement; CIS 1.1, Establish and Maintain Detailed Enterprise Asset Inventory)
2. Step 2: Detection, Query your SIEM for Azure Linux 3.0 hosts with kernel version 6.6.139.1-1 in your asset data. Review system logs (dmesg, kernel ring buffer, /var/log/kern.log) for OCFS2/DLM subsystem errors or unexpected out-of-bounds memory events. Monitor for anomalous privilege escalation attempts on affected hosts using process audit logs, referencing MITRE ATT&CK T1068 behavioral patterns. (Supports NIST AU-6, Audit Record Review, Analysis, and Reporting; CIS 8.2, Collect Audit Logs)
3. Step 3: Eradication, Apply the updated azl3 kernel package provided by Microsoft through the June 2026 Patch Tuesday cycle. Consult the MSRC consolidated advisory at https://api.msrc.microsoft.com/cvrf/v3.0/cvrf/2026-Jun for the specific package version that resolves CVE-2026-53309. Follow your standard kernel update and reboot procedures for Azure Linux 3.0 nodes. (Supports NIST SI-2, Flaw Remediation; CIS 7.3, Perform Automated Operating System Patch Management)
4. Step 4: Recovery, After patching, verify the running kernel version on all previously affected hosts to confirm the vulnerable package is no longer active. Re-enable any isolated OCFS2/DLM cluster nodes after patch verification. Monitor kernel logs and DLM subsystem activity for at least 72 hours post-patch for anomalous behavior. (Supports NIST AU-6, Audit Record Review, Analysis, and Reporting; D3-SFA, System File Analysis)
5. Step 5: Post-Incident, Review patch management processes for Azure Linux 3.0 to confirm coverage of kernel-level advisories from MSRC Patch Tuesday cycles. Assess whether automated OS patch management (CIS 7.3) is consistently applied to all Azure Linux hosts. Document any hosts that were not captured in initial inventory queries as a gap finding. (Supports CIS 7.1, Establish and Maintain a Vulnerability Management Process; CIS 7.2, Establish and Maintain a Remediation Process)

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate immediately to senior IR leadership and, if applicable, cloud/infrastructure ownership if any Azure Linux 3.0 OCFS2/DLM cluster node shows kernel BUG or oops messages in `/var/log/kern.log` referencing dlm_match_regions(), any unexplained UID 0 privilege escalation in audit logs on an affected host, or if the environment processes regulated data (PII, PHI, PCI-DSS scope) that may trigger mandatory breach notification obligations under HIPAA, GDPR, or applicable state law.
Recovery Notes	After applying the patched azl3 kernel from the June 2026 Patch Tuesday cycle, verify that `uname -r` on every previously affected node no longer reports 6.6.139.1-1 and that the OCFS2/DLM cluster has returned to a healthy quorum state confirmed via `o2cb status` and `dlm_tool ls`. Monitor `/var/log/kern.log` and kernel ring buffer output continuously for a minimum of 72 hours post-patch for any residual OCFS2 or DLM subsystem errors that may indicate prior heap corruption from the OOB condition persisting across the patch boundary. Any node that experienced a confirmed kernel oops or BUG during the exposure window should be considered for full reimage rather than patch-in-place recovery, as slab corruption from an off-by-one OOB write may leave the system in an indeterminate state.
Forensic Artifacts	Kernel ring buffer output (`dmesg` and `/var/log/kern.log`): an exploited or triggered off-by-one OOB in dlm_match_regions() would likely surface as a kernel BUG, WARN, or general protection fault with a stack trace naming the dlm_match_regions() or adjacent OCFS2/DLM functions — this is the primary indicator of exploitation or crash-triggering. | Linux audit log (`/var/log/audit/audit.log`) SYSCALL records: look for `type=SYSCALL` entries showing an unexpected UID/EUID transition to 0 on OCFS2/DLM cluster nodes, which would indicate successful local privilege escalation via the OOB memory write primitive. | Volatile memory image (LiME dump): a RAM capture from an affected azl3 6.6.139.1-1 host preserves the kernel heap state at time of exploitation, enabling post-mortem reconstruction of the OOB write in the dlm_match_regions() boundary comparison logic and identification of any overwritten adjacent kernel structures. | `/proc/slabinfo` and `/proc/meminfo` snapshots: an off-by-one OOB in DLM region boundary logic may corrupt adjacent slab allocations; anomalous slab cache counts or unexpected memory pressure on a node with low workload are a forensic indicator of heap manipulation. | TDNF transaction log (`/var/log/tdnf.log`) and `rpm -qa --last | grep kernel` output: these establish the authoritative pre- and post-patch kernel version timeline on Azure Linux 3.0 nodes, confirming when the vulnerable azl3 6.6.139.1-1 package was present and when the remediated package replaced it.

Per-Action IR Details

Step 1: Containment — Identify all Azure Linux 3.0 hosts running azl3 kernel 6.6.139.1-1 using your asset inventory. Isolate or restrict network access to any OCFS2/DLM-enabled cluster nodes until the patch is confirmed applied. Reference MSRC Update Guide at https://msrc.microsoft.com/update-guide/vulnerability/CVE-2026-53309 for Microsoft's specific guidance. (Supports NIST AC-4 — Information Flow Enforcement; CIS 1.1 — Establish and Maintain Detailed Enterprise Asset Inventory)

NIST Phase: Containment

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

Controls: NIST AC-4 (Information Flow Enforcement), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory)

Compensating: Run `uname -r` remotely across all Azure Linux 3.0 hosts via a parallel SSH loop (e.g., `pssh` or `for h in $(cat hosts.txt); do ssh $h uname -r; done`) to enumerate nodes running 6.6.139.1-1. For OCFS2/DLM-specific exposure, run `lsmod | grep ocfs2` and `dlm_tool ls` on each candidate node to confirm DLM is active before prioritizing isolation. Use host-based firewall rules (`iptables -I INPUT -p tcp --dport 7777 -j DROP` for the default DLM port) as a lightweight isolation measure if full network segmentation is unavailable.

Evidence: Before isolating any OCFS2/DLM cluster node, capture: (1) full memory image using LiME kernel module (`insmod lime.ko path=/tmp/mem.lime format=lime`) to preserve any in-memory exploitation artifacts in the dlm_match_regions() stack frame; (2) active DLM lock state via `dlm_tool ls` and `dlm_tool lockdebug `; (3) current network connections from the DLM cluster using `ss -tunap | grep 7777` to identify which nodes are actively communicating over the DLM port before network access is restricted. Loss of live DLM state will make post-incident reconstruction of exploitation attempts significantly harder.

Step 2: Detection — Query your SIEM for Azure Linux 3.0 hosts with kernel version 6.6.139.1-1 in your asset data. Review system logs (dmesg, kernel ring buffer, /var/log/kern.log) for OCFS2/DLM subsystem errors or unexpected out-of-bounds memory events. Monitor for anomalous privilege escalation attempts on affected hosts using process audit logs, referencing MITRE ATT&CK T1068 behavioral patterns. (Supports 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), CIS 8.2 (Collect Audit Logs)

Compensating: Without a SIEM, run `journalctl -k --since '7 days ago' | grep -iE 'ocfs2|dlm|oob|out.of.bounds|BUG:|kernel BUG|general protection fault'` on each candidate host to surface kernel-level anomalies from the OCFS2/DLM subsystem. For privilege escalation indicators, use `ausearch -m AVC,USER_ROLE_CHANGE,SYSCALL -ts today` (requires auditd) and filter for unexpected UID 0 transitions. Deploy a lightweight Sigma rule targeting `syslog` sources matching OCFS2 BUG or oops strings if a local log aggregator (e.g., rsyslog forwarding to a central host) is available.

Evidence: Specific artifacts to collect before taking any containment action that alters live state: (1) full `dmesg` output — an off-by-one OOB in dlm_match_regions() may produce a kernel WARN or BUG message with a stack trace naming the faulting function; (2) `/var/log/kern.log` and `/var/log/audit/audit.log` entries around the time of any suspicious process UID change (look for `type=SYSCALL` records with `uid=1000 euid=0` transitions); (3) `/proc/slabinfo` and `/proc/meminfo` snapshots to capture heap state indicative of slab corruption from the OOB write; (4) `cat /proc//maps` for any process that escalated privileges to identify memory regions that may have been corrupted by the exploit.

Step 3: Eradication — Apply the updated azl3 kernel package provided by Microsoft through the June 2026 Patch Tuesday cycle. Consult the MSRC consolidated advisory at https://api.msrc.microsoft.com/cvrf/v3.0/cvrf/2026-Jun for the specific package version that resolves CVE-2026-53309. Follow your standard kernel update and reboot procedures for Azure Linux 3.0 nodes. (Supports NIST SI-4 — no mapped control from knowledge base for patching; CIS 7.3 — Perform Automated Operating System 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)

Compensating: For nodes without automated patch management, use `tdnf update kernel` (Azure Linux 3.0 uses TDNF as its package manager) to pull the patched azl3 kernel from the Microsoft CBL-Mariner package repository. Verify the specific fixed package version against the MSRC June 2026 CVRF advisory before applying. After the kernel update, execute `reboot` and confirm the new kernel version with `uname -r` post-boot. For air-gapped nodes, download the signed RPM from the Microsoft package repository on a connected system and transfer via secure copy before installing with `tdnf install ./kernel-.rpm`.

Evidence: Before applying the kernel patch and rebooting (which destroys all volatile state), capture: (1) complete RAM image using LiME if not already collected during containment — a reboot irrecoverably destroys any in-memory exploitation evidence from the dlm_match_regions() OOB condition; (2) `cat /proc/version` and `rpm -qi kernel` output to establish the pre-patch version as a forensic baseline; (3) a full copy of `/var/log/kern.log`, `/var/log/audit/audit.log`, and the output of `dmesg` timestamped immediately before patching; (4) list of currently loaded kernel modules via `lsmod` — a successful privilege escalation exploit may have loaded a malicious LKM that will persist or re-emerge if

not identified before reimaging.

Step 4: Recovery — After patching, verify the running kernel version on all previously affected hosts to confirm the vulnerable package is no longer active. Re-enable any isolated OCFS2/DLM cluster nodes after patch verification. Monitor kernel logs and DLM subsystem activity for at least 72 hours post-patch for anomalous behavior. (Supports NIST AU-6 — Audit Record Review, Analysis, and Reporting; D3-SFA — System File Analysis)

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-3 (Content Of Audit Records)

Compensating: Verify the patched kernel is active with `uname -r` and cross-reference against the fixed version listed in the MSRC June 2026 advisory. To confirm the dlm_match_regions() boundary fix is in place without a SIEM, run a 72-hour continuous log watch using `journalctl -k -f | grep -iE 'ocfs2|dlm|BUG:|oops|out.of.bounds'` on each recovered OCFS2 cluster node and pipe output to a timestamped log file. Reintroduce nodes to the DLM cluster incrementally (one node at a time) and validate cluster health with `o2cb status` and `dlm_tool ls` before restoring the full cluster.

Evidence: This step does not destroy volatile state (it is post-patch verification), so no pre-action volatile capture is required. However, document: (1) `uname -r` output from every recovered node as the authoritative post-patch kernel version record; (2) `rpm -qa | grep kernel` output to confirm removal or supersession of the vulnerable 6.6.139.1-1 package; (3) `dlm_tool ls` and `o2cb status` output confirming clean DLM lockspace state post-recovery — anomalous lock contention or node fencing events in this window may indicate residual instability from a prior OOB corruption event rather than a new exploitation attempt.

Step 5: Post-Incident — Review patch management processes for Azure Linux 3.0 to confirm coverage of kernel-level advisories from MSRC Patch Tuesday cycles. Assess whether automated OS patch management (CIS 7.3) is consistently applied to all Azure Linux hosts. Document any hosts that were not captured in initial inventory queries as a gap finding. (Supports CIS 7.1 — Establish and Maintain a Vulnerability Management Process; CIS 7.2 — Establish and Maintain a Remediation Process)

NIST Phase: Post Incident

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

Controls: CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: For teams without automated vulnerability management tooling, establish a recurring monthly cron job or calendar reminder to query all Azure Linux 3.0 hosts for kernel version using `pssh` or Ansible ad-hoc (`ansible all -m command -a 'uname -r'`) within 72 hours of each MSRC Patch Tuesday release. Cross-reference output against the MSRC CVRF feed for Azure Linux advisories. Document any host not present in the inventory query result as an ungoverned asset and assign remediation ownership.

Evidence: No volatile evidence capture is required at this phase. Artifacts to preserve for the post-incident record include: (1) the full list of hosts identified as running azl3 kernel 6.6.139.1-1, with timestamps of identification and remediation; (2) any hosts discovered during incident response that were absent from the initial asset inventory — these represent an inventory gap directly relevant to CVE-2026-53309 exposure; (3) patch deployment timestamps from TDNF transaction logs (`/var/log/tdnf.log`) on each remediated host to establish a defensible remediation timeline for audit or regulatory purposes.

Detection Guidance

Query your asset inventory and configuration management database for hosts running Azure Linux 3.0 with azl3 kernel version 6.6.139.1-1. On potentially affected hosts, run 'uname -r' output collection via your endpoint management tooling and compare against the vulnerable version. Review kernel logs (/var/log/kern.log, dmesg

output) for OCFS2 or DLM subsystem errors, particularly messages referencing dlm_match_regions or memory boundary violations. For privilege escalation detection aligned with T1068, monitor audit logs for unexpected setuid/setgid execution, capability changes, or processes achieving elevated privilege outside normal operational baselines. No IOCs specific to active exploitation of this CVE were present in the source material; detection at this stage is posture-based, not IOC-based. Consult the NVD entry (https://nvd.nist.gov/vuln/detail/CVE-2026-53309) as it is updated for any additional detection signatures once full exploitability details are confirmed. (Supports NIST AU-2, Event Logging; NIST AU-6, Audit Record Review, Analysis, and Reporting; CIS 8.2, Collect Audit Logs; D3-LAM, Local Account Monitoring)

Framework Mappings

MITRE-ATTACK

- **T1068** — Exploitation for Privilege Escalation

NIST-800-53R5

- **AC-6** — Least Privilege
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation

CIS-V8

- **7.3** — Perform Automated Operating System Patch Management
- **7.4** — Perform Automated Application Patch Management
- **5.4** — Restrict Administrator Privileges to Dedicated Administrator Accounts

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities
- **A.5.23** — Information security for use of cloud services

SOC2-TSC

- **CC6.3** — Authorizes, modifies, or removes access

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1068	Exploitation for Privilege Escalation	Privilege-Escalation

Sources

Source	URL	Tier
MSRC Update Guide	https://msrc.microsoft.com/update-guide/vulnerability/CVE-2026-53309	T1
(consolidated)	https://api.msrc.microsoft.com/cvrf/v3.0/cvrf/2026-Jun	T1

Source	URL	Tier
CVE-2026-53309 Detail - NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-53309	**T1**
CVE-2026-53309 - CVE Record	https://www.cve.org/CVERecord?id=CVE-2026-53309	**T1**
CVE-2026-53309 Common Vulnerabilities and Exposures | SUSE	https://www.suse.com/security/cve/CVE-2026-53309.html	**T3**

Generated 2026-07-02 07:14 UTC by TJS Security Command Center