INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-08 08:12 UTC

CVE-2026-48095: 7-Zip is a file archiver with a high compression ratio. Versions 26.00 and prior contain a heap buff

CVE VULNERABILITY | HIGH | CVSS 8.8

SCC Item ID	SCC-CVE-2026-0272
Type	CVE Vulnerability
CVE ID	CVE-2026-48095
Severity	HIGH
CVSS Base Score	8.8
EPSS Score	0.0004 (14th percentile)
Affected Products	7-Zip versions 26.00 and prior; fixed in 7-Zip 26.01
Published	2026-06-05T15:16:53.520
Discovery Source	Nvd

Executive Summary

A heap buffer overflow in 7-Zip versions 26.00 and earlier allows an attacker to achieve arbitrary code execution by convincing a user to open or extract a crafted NTFS archive. The vulnerability is triggered by default 7-Zip behavior, no special configuration is required, making any organization with 7-Zip deployed on user workstations or servers directly exposed. CISA added this to its Known Exploited Vulnerabilities catalog on 2026-05-20, confirming active exploitation in the wild.

Technical Analysis

CVE-2026-48095 is a heap buffer overflow in 7-Zip's NTFS compressed stream handler, affecting all versions through 26.00. The root cause is an integer overflow (CWE-190) in CInStream::GetCuSize(): the expression (UInt32)1 = 28 and CompressionUnit == 4, driving the exponent to 32 and collapsing _inBuf allocation to 1 byte on x86/x64. ReadStream_FALSE then writes up to 256 MB of attacker-controlled data into that 1-byte buffer in 64 KB iterations, an out-of-bounds write (CWE-787). The CInStream vtable pointer sits 304 bytes after _inBuf; overflow overwrites it, and the next virtual dispatch achieves a vtable hijack enabling arbitrary code execution (MITRE T1203, T1059). On 32-bit builds, the code path is unconditionally reachable. On 64-bit, exploitation requires a successful parallel 8 GB _outBuf allocation; failure degrades to denial of service. The NTFS handler is enabled by default in 7z.dll and activates via signature-based fallback matching 'NTFS ' at offset 3, triggering on crafted images regardless of file extension. CVSS base score: 8.8 (High). Fixed in 7-Zip 26.01. CISA KEV

alert dated 2026-05-20.

Action Checklist

1. Step 1: Containment. Immediately identify all endpoints and servers with 7-Zip 26.00 or earlier installed. Use endpoint management tooling (SCCM, Intune, or equivalent) to query installed software. Restrict or disable 7-Zip extraction operations via application control policy until patching is confirmed. Treat systems that have opened untrusted archives recently as potentially compromised.
2. Step 2: Detection. Query EDR and AV logs for execution chains spawned from 7z.exe or 7zFM.exe, particularly child processes indicating post-exploitation (cmd.exe, powershell.exe, mshta.exe). Hunt for files with NTFS image signatures ('NTFS ' at offset 3) delivered via email, web download, or file share in the past 30 days (MITRE T1204.002). Review process creation events for abnormal crashes or access violation events originating from 7z.dll. No specific public IOCs, hashes, IPs, or domains are confirmed in available source data at this time.
3. Step 3: Eradication. Deploy 7-Zip 26.01 from the official vendor source (7-zip.org) to all affected endpoints and servers. Validate installation via software inventory reconciliation. Remove or replace any unmanaged copies of 7z.dll in application-bundled deployments.
4. Step 4: Recovery. After patching, verify 7-Zip 26.01 is the active version on all previously exposed assets. Re-scan systems that processed untrusted NTFS archives prior to patching with EDR behavioral analysis. Monitor for anomalous process execution from archive handler processes for a minimum of 14 days post-remediation. Validate that no persistence mechanisms (scheduled tasks, registry run keys, startup entries) were installed on systems flagged in Step 2.
5. Step 5: Post-Incident. Conduct a review of software patch management controls to assess why 7-Zip 26.00 remained deployed post-disclosure. Update software inventory and vulnerability management process to include third-party archive utilities. Evaluate whether application allowlisting can restrict 7-Zip execution to approved use cases. Document findings and update incident response procedures.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to senior IR leadership, legal counsel, and executive stakeholders immediately if Step 2 detection confirms a process execution chain from 7z.exe or 7zFM.exe to cmd.exe, powershell.exe, or mshta.exe on any host, or if persistence artifacts are found on any system that handled sensitive, PII, PHI, or regulated data — as active exploitation of CVE-2026-48095 on such a system may trigger breach notification obligations under HIPAA, GDPR, or applicable state privacy law.
Recovery Notes	After deploying 7-Zip 26.01 and clearing flagged hosts, maintain an elevated monitoring posture specifically for process creation events originating from archive handler processes (7z.exe, 7zFM.exe, and any application bundling 7z.dll) for a minimum of 14 days, as post-exploitation implants installed prior to patching will survive the 7-Zip update and may resume beaconing or lateral movement after the immediate response subsides. Re-run the full osquery persistence sweep (scheduled tasks, registry run keys, new services) on all previously flagged hosts at Day 7 and Day 14 post-patch to detect delayed-activation persistence mechanisms. Confirm that no additional unmanaged copies of 7z.dll were identified in subsequent software scans before formally closing the incident under IR-5.

Forensic Artifacts	Windows Application Event Log — Event ID 1000 (Application Error) with Faulting Module Name '7z.dll' and Exception Code 0xC0000005 (Access Violation): indicates heap buffer overflow trigger in the NTFS archive parser, potentially representing failed or probe exploitation attempts before a successful execution. | Prefetch files at %SystemRoot%\Prefetch\7Z*.pf and 7ZFM*.pf: record last execution timestamps and the paths of archive files opened by 7-Zip, allowing reconstruction of which crafted NTFS images were processed and when — critical for scoping the compromise window. | Windows Error Reporting (WER) crash dumps at %LOCALAPPDATA%\CrashDumps\ and %ProgramData%\Microsoft\Windows\WER\ReportArchive\: may contain heap memory state from 7z.dll at the moment of overflow, providing evidence of vtable corruption and potentially shellcode or ROP chain artifacts if exploitation was attempted but failed. | Sysmon Event ID 1 (Process Create) records where ParentImage matches 7z.exe or 7zFM.exe and child Image is cmd.exe, powershell.exe, mshta.exe, or rundll32.exe: the definitive indicator of successful arbitrary code execution via CVE-2026-48095, as legitimate 7-Zip operation does not spawn shell interpreters. | File system artifacts in email attachment staging paths (%LOCALAPPDATA%\Microsoft\Windows\INetCache, Outlook Secure Temp folder at %APPDATA%\Local\Microsoft\Windows\Temporary Internet Files\Content.Outlook\) and browser download directories for files with the NTFS volume boot record signature (bytes 3–10: 0x4E 0x54 0x46 0x53 0x20 0x20 0x20 0x20) regardless of file extension: these are the crafted NTFS archive delivery artifacts specific to this exploit's attack vector.

Per-Action IR Details

Step 1: Containment — Immediately identify all endpoints and servers with 7-Zip 26.00 or earlier installed. Use endpoint management tooling (SCCM, Intune, or equivalent) to query installed software against CIS 1.1 asset inventory. Restrict or disable 7-Zip extraction operations via application control policy until patching is confirmed. Treat systems that have opened untrusted archives recently as potentially compromised.

NIST Phase: Containment

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

Controls: NIST IR-4 (Incident Handling), NIST CM-7 (Least Functionality), NIST CM-8 (System Component Inventory), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 2.1 (Establish and Maintain a Software Inventory)

Compensating: Without SCCM/Intune, run the following PowerShell one-liner across all Windows hosts via PSRemoting to enumerate exposed systems: `Invoke-Command -ComputerName (Get-Content hosts.txt) -ScriptBlock { Get-ItemProperty HKLM:\Software\Microsoft\Windows\CurrentVersion\Uninstall\* | Where-Object { $_.DisplayName -like '*7-Zip*' } | Select-Object PSComputerName, DisplayName, DisplayVersion }`. On Linux/macOS, use `dpkg -l | grep -i 7zip` or `find / -name '7z' -o -name '7zFM' 2>/dev/null`. To block execution without EDR, create a Windows AppLocker rule or a Software Restriction Policy denying execution of 7z.exe, 7zFM.exe, and 7zG.exe by path or hash derived from the 26.00 binary.

Evidence: Before isolating or modifying any flagged system, capture: (1) Windows Security Event Log (Event ID 4688 — Process Creation) for any process spawned by 7z.exe or 7zFM.exe in the past 30 days; (2) the file system path and metadata (creation time, last accessed, SHA-256 hash) of any .ntfs, .img, or .iso files received via email attachment or web download, as these are the delivery vectors for a crafted NTFS archive exploit; (3) prefetch files (%SystemRoot%\Prefetch\7Z*.pf and 7ZFM*.pf) to establish archive open history and timestamps before any remediation alters them.

Step 2: Detection — Query EDR and AV logs for execution chains spawned from 7z.exe or 7zFM.exe, particularly child processes indicating post-exploitation (cmd.exe, powershell.exe, mshta.exe). Hunt for files with NTFS image signatures ('NTFS ' at offset 3) delivered via email, web download, or file share in the past 30 days (MITRE T1204.002). Review process creation events for vtable hijack indicators: abnormal virtual dispatch crashes or access violation events originating from 7z.dll. Reference NIST AU-2 for event logging

scope and AU-6 for audit record review cadence. No specific public IOCs — hashes, IPs, domains — are confirmed in available source data at this time.

NIST Phase: Detection Analysis

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

Controls: NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST IR-5 (Incident Monitoring), CIS 8.2 (Collect Audit Logs)

Compensating: Without EDR, deploy Sysmon with a configuration that enables Event ID 1 (Process Create) with ParentImage and CommandLine fields, and Event ID 10 (ProcessAccess) to catch memory injection from 7z.dll into child processes. Use the following Sigma rule logic as a manual grep against exported Sysmon XML logs: filter for EventID=1 where ParentImage ends in '7z.exe' OR '7zFM.exe' AND Image matches cmd.exe, powershell.exe, or mshta.exe. To hunt for crafted NTFS archives on disk or in email stores, run: `Get-ChildItem -Recurse -Path C:\Users -Include *.img,*.ntfs,*.iso | ForEach-Object { $bytes = [System.IO.File]::ReadAllBytes($_.FullName); if ($bytes[3..10] -join '' -match 'NTFS') { $_.FullName } }`. For crash/AV exception evidence from vtable corruption, query Windows Application Event Log for Event ID 1000 (Application Error) where Faulting Module Name contains '7z.dll'.

Evidence: Capture before hunting: (1) Windows Application Event Log entries for Event ID 1000 (Application Error) referencing 7z.dll as the faulting module — a vtable hijack via heap overflow will manifest as an access violation (exception code 0xC0000005) in 7z.dll prior to successful exploitation; (2) browser download history and email client attachment staging directories (e.g., %LOCALAPPDATA%\Microsoft\Windows\INetCache, Outlook's %APPDATA%\Local\Microsoft\Windows\Temporary Internet Files\Content.Outlook) for .ntfs or disk image files received in the 30-day window; (3) Windows Error Reporting (WER) crash dumps located at %LOCALAPPDATA%\CrashDumps or %ProgramData%\Microsoft\Windows\WER\ReportArchive for any 7z.exe or 7zFM.exe crashes, as these may contain heap state from the overflow condition.

Step 3: Eradication — Deploy 7-Zip 26.01 from the official vendor source (7-zip.org) to all affected endpoints and servers. Validate installation via CIS 2.1 software inventory reconciliation. Remove or replace any unmanaged copies of 7z.dll in application-bundled deployments. Apply CM-6 configuration baseline updates to reflect the new approved version.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST CM-6 (Configuration Settings), NIST CM-2 (Baseline Configuration), NIST SI-2 (Flaw Remediation), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 2.2 (Ensure Authorized Software is Currently Supported), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management)

Compensating: Without centralized patch management, use a PowerShell script to silently install 7-Zip 26.01 MSI remotely: `Invoke-Command -ComputerName (Get-Content hosts.txt) -ScriptBlock { Start-Process msiexec.exe -ArgumentList '/i \\fileserver\patches\7z2601-x64.msi /qn /norestart' -Wait }`. To find bundled copies of 7z.dll outside the standard Program Files path (which are not updated by the installer and remain vulnerable), run: `Get-ChildItem -Recurse -Path C:\ -Filter 7z.dll -ErrorAction SilentlyContinue | Where-Object { $_.DirectoryName -notlike '*7-Zip*' } | Select-Object FullName, LastWriteTime`. Each identified bundled DLL must be replaced or the parent application updated. Verify the patched DLL version with: `(Get-Item 'C:\Program Files\7-Zip\7z.dll').VersionInfo.FileVersion` — expect 26.01.

Evidence: Before eradicating, preserve: (1) SHA-256 hashes of the vulnerable 7z.exe and 7z.dll binaries from each affected host for chain-of-custody documentation and post-incident comparison; (2) a full directory listing with timestamps of %ProgramFiles%\7-Zip\ and any non-standard paths where 7z.dll was found, to document the pre-patch state of bundled deployments; (3) on any system flagged as potentially compromised in Step 1, acquire a memory image or at minimum a full process list with loaded modules (using Sysinternals Process Explorer or `tasklist /m 7z.dll`) before patching, as eradication will overwrite the vulnerable DLL and disrupt post-exploitation memory forensics.

Step 4: Recovery — After patching, verify 7-Zip 26.01 is the active version on all previously exposed assets. Re-scan systems that processed untrusted NTFS archives prior to patching with EDR behavioral analysis. Monitor for anomalous process execution from archive handler processes for a minimum of 14 days

post-remediation per IR-5 incident monitoring requirements. Validate that no persistence mechanisms (scheduled tasks, registry run keys, startup entries) were installed on systems flagged in Step 2.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST IR-5 (Incident Monitoring), NIST CM-3 (Configuration Change Control), NIST AU-6 (Audit Record Review, Analysis, And Reporting), CIS 4.6 (Securely Manage Enterprise Assets and Software)

Compensating: Without EDR for behavioral re-scan, use osquery to query persistence mechanisms on flagged hosts: run `SELECT * FROM scheduled_tasks WHERE action LIKE '%cmd%' OR action LIKE '%powershell%' OR action LIKE '%mshta%'` and `SELECT * FROM registry WHERE path LIKE 'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run%'` to detect attacker-installed persistence. For the 14-day monitoring window without SIEM, configure a Sysmon Event ID 1 alert that fires when any process with a parent of 7z.exe or 7zFM.exe is created and forward those events to a central syslog server. Additionally, use Autoruns (Sysinternals) in offline scan mode against flagged systems to enumerate all persistence points and compare against a known-good baseline from an uncompromised peer workstation.

Evidence: Before clearing systems for return to production: (1) export the full contents of HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, and the Windows Task Scheduler store (%SystemRoot%\System32\Tasks\) from each flagged host to document any attacker-installed persistence that must be removed; (2) collect a listing of all new services created since the earliest date a crafted NTFS archive was known to be processed, using `Get-WinEvent -LogName System | Where-Object { $_.Id -eq 7045 }` (Event ID 7045 — New Service Installed), as post-exploitation service installation is a common persistence technique following arbitrary code execution; (3) verify integrity of 7-Zip 26.01 installation by comparing installed DLL hash against the SHA-256 published in the 7-zip.org release notes to confirm no trojanized binary was deployed during the recovery window.

Step 5: Post-Incident — Conduct a gap review against CIS 7.3 and CIS 7.4 automated patch management controls to assess why 7-Zip 26.00 remained deployed post-disclosure. Update software inventory (CIS 2.1) and vulnerability management process (CIS 7.1) to include third-party archive utilities. Evaluate whether application allowlisting (CM-7 Least Functionality) can restrict 7-Zip execution to approved use cases. Document findings per IR-4 and IR-8 incident response plan requirements.

NIST Phase: Post Incident

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

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST CM-7 (Least Functionality), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management)

Compensating: For teams without a formal vulnerability management platform, implement a monthly osquery scheduled query to detect any installed 7-Zip version below the current approved baseline: `SELECT name, version FROM programs WHERE name LIKE '%7-Zip%' AND version < '26.01'`. Encode this query into a cron job or Windows Scheduled Task that outputs to a shared log directory reviewed by the security team. To address the root cause of why 7-Zip 26.00 persisted post-disclosure, map the delay between CVE-2026-48095 CISA KEV listing (2026-05-20) and patch deployment against the organization's documented remediation SLA under CIS 7.2, and present findings at the lessons-learned meeting. Draft an AppLocker rule restricting 7zFM.exe (the GUI that allows casual opening of untrusted archives) to a named group of approved power users, while permitting 7z.exe only for scripted, IT-managed extraction workflows.

Evidence: For the lessons-learned record: (1) compile the timeline from CVE-2026-48095 public disclosure through CISA KEV addition (2026-05-20) to organizational patch completion, sourcing dates from patch management logs, change tickets, and the software inventory snapshots taken during Step 3; (2) document all systems where 7z.dll was found in non-standard bundled application paths (discovered in Step 3), as these represent a structural gap in the organization's third-party library visibility that the vulnerability management process must remediate; (3) retain all forensic artifacts collected across Steps 1–4 per NIST AU-11 (Audit Record Retention) requirements and the organization's incident records retention policy, ensuring WER crash dumps, Sysmon logs, prefetch captures, and

memory images from flagged hosts are preserved in a write-protected evidence store.

Detection Guidance

Primary detection surface is process execution telemetry. Alert on child processes spawned by 7z.exe, 7zFM.exe, or any process loading 7z.dll where the child is a shell interpreter (cmd.exe, powershell.exe, wscript.exe, mshta.exe). Secondary signal: access violation or crash events (Windows Event ID 1000/1001) with faulting module 7z.dll, these may indicate failed 64-bit exploitation degrading to denial of service. For file-based detection, hunt for NTFS image files delivered outside normal administrative channels: files matching the 'NTFS ' signature at byte offset 3, regardless of file extension (.img, .iso, .7z, .zip, or no extension). Query email gateway and proxy logs for archive file downloads from untrusted external sources in the 30 days prior to patch. No confirmed public IOC hashes, IPs, or domains are available in source data for this CVE at this time; absence of IOCs does not reduce risk given confirmed KEV status.

Framework Mappings

MITRE-ATTACK

- **T1204.002** — Malicious File
- **T1059** — Command and Scripting Interpreter
- **T1203** — Exploitation for Client Execution

NIST-800-53R5

- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **SI-16** — Memory Protection
- **AT-2** — Literacy Training and Awareness

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures
- **7.3** — Perform Automated Operating System Patch Management
- **7.4** — Perform Automated Application Patch Management
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks
- **8.2** — Collect Audit Logs

ISO-27001-2022

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1204.002	Malicious File	Execution
T1059	Command and Scripting Interpreter	Execution
T1203	Exploitation for Client Execution	Execution

Sources

Source	URL	Tier
nvd	https://nvd.nist.gov/vuln/detail/CVE-2026-48095	T1
CVE-2026-48095: 7-Zip Heap Overflow Flaw - SOC Prime	https://socprime.com/blog/cve-2026-48095-7-zip-heap-overflow-flaw/	T3
CVE-2026-48095 | Security Details | Sonatype Guide - OSS Index	https://guide.sonatype.com/vulnerability/CVE-2026-48095	T3
CVE-2026-48095 - CVE Record	https://www.cve.org/CVERecord?id=CVE-2026-48095	T3
CISA Adds Seven Known Exploited Vulnerabilities to Catalog	https://www.cisa.gov/news-events/alerts/2026/05/20/cisa-adds-seven-...	T1

Generated 2026-06-08 08:12 UTC by TJS Security Command Center