INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-04-24 13:43 UTC

Warning: Two critical unauthenticated code execution vulnerabilities in Rclone, Patch Immediately!

CVE VULNERABILITY | **CRITICAL** | CVSS 9.2

SCC Item ID	SCC-CVE-2026-0075
Type	CVE Vulnerability
CVE ID	CVE-2026-41176
Severity	CRITICAL
CVSS Base Score	9.2
EPSS Score	0.0279 (86th percentile)
Affected Products	Rclone versions below 1.73.5
Published	2 hours ago
Discovery Source	Serper

Executive Summary

Two critical unauthenticated remote code execution vulnerabilities have been disclosed in Rclone, an open-source cloud storage synchronization tool commonly used in enterprise environments for data movement across cloud platforms. Attackers can exploit an exposed Rclone RC (remote control) interface without credentials to execute arbitrary file system operations and potentially run malicious code on affected systems. Any organization running Rclone below version 1.73.5 with the RC interface exposed should treat this as an emergency patching event.

Technical Analysis

One or more critical vulnerabilities affect Rclone versions below 1.73.5. CVE-2026-41176 (CVSS 9.2) enables unauthenticated remote code execution. A companion advisory (GHSA-jfwf-28xr-xw6q) documents unauthenticated access to the Rclone RC interface via the operations/fsinfo endpoint, allowing attackers to bypass authentication entirely and interact with the RC API without credentials. CWE-306 (Missing Authentication for Critical Function) and CWE-94 (Code Injection) are the underlying weaknesses. CERT-Bund has published related guidance on Rclone flaws. A public proof-of-concept exploit has been released for at least one vulnerability, increasing exploitation probability across unskilled threat actors. EPSS score is 0.028 (86th percentile), indicating elevated exploitation probability relative to the broader CVE population. MITRE ATT&CK

techniques: T1059 (Command and Scripting Interpreter), T1190 (Exploit Public-Facing Application). T1083 (File and Directory Discovery) may be used post-exploitation for data enumeration. No threat actor attribution is confirmed at this time. The RC interface is the primary attack surface when started without authentication configured (missing --rc-user and --rc-pass flags), and it listens on a default port reachable by any network-adjacent or internet-facing attacker. Patch target: Rclone 1.73.5 or later. NVD entry: https://nvd.nist.gov/vuln/detail/CVE-2026-41176

Action Checklist

1. Step 1: Containment (0-4 hours). Immediately identify all systems running Rclone below version 1.73.5 using your asset inventory or endpoint management tooling. Disable the Rclone RC interface entirely if operationally unnecessary (remove --rc and --rc-addr flags from startup scripts and service definitions). If the RC interface is operationally required, immediately block inbound access to port 5572 (or any custom RC port configured in your environment) at the host firewall or network perimeter until patching is complete.
2. Step 2: Detection (0-8 hours). Query endpoint logs, process execution logs, and network flow data for Rclone RC interface activity. Look for inbound HTTP connections to port 5572 (or any custom RC port) originating from unexpected source IPs. Search command execution logs for rclone processes spawned with --rc, --rc-addr, or --rc-no-auth flags. Review server access logs for POST requests to /operations/fsinfo or other RC API endpoints from unauthorized sources. Check for anomalous file system operations or data transfers initiated around the time of any suspicious RC activity.
3. Step 3: Eradication (4-24 hours). Upgrade all Rclone installations to version 1.73.5 or later using your standard software deployment process. Update package manager definitions (apt, yum, brew, Chocolatey) and pull the latest release from the official Rclone repository (https://rclone.org/downloads/). For containerized deployments, rebuild images with the patched base version. After upgrade, enforce RC authentication on any remaining RC-enabled instances using --rc-user and --rc-pass flags, or bind the RC interface exclusively to localhost (--rc-addr 127.0.0.1:5572) where remote access is not required.
4. Step 4: Recovery (24-72 hours). Verify the installed Rclone version on all endpoints with 'rclone version' and confirm output shows 1.73.5 or later. Re-enable any RC interfaces that were disabled for containment only after authentication controls are confirmed in place. Monitor port 5572 and RC-related process activity for 72 hours post-patch for any signs of residual exploitation or persistence. Review Rclone logs for operations executed during the exposure window to assess whether unauthorized file system access occurred.
5. Step 5: Post-Incident (3-7 days). Conduct an inventory audit to establish a current baseline of all Rclone deployments, versions, and RC interface configurations across the environment. Add Rclone to your vulnerability management scanning scope if not already present. Implement a policy requiring RC interface authentication and localhost binding by default in all Rclone service configurations. Review network segmentation to ensure developer and data pipeline tooling with RC-style management interfaces is not directly reachable from untrusted networks. Document exposure window and affected systems for internal reporting.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate to CISO and legal/privacy counsel immediately if rclone log review or cloud provider audit logs reveal RC API calls to /operations/copyfile, /sync/sync, or /config/create from any unauthorized source IP during the exposure window, or if rclone.conf contains cloud remotes storing PII, PHI, or data subject to PCI-DSS, HIPAA, or GDPR breach notification obligations.
Recovery Notes	After patching to Rclone 1.73.5 and re-enabling any RC interfaces with enforced authentication, conduct a cross-reference of rclone process logs against configured cloud remote access logs (AWS CloudTrail, Azure Monitor, GCP Cloud Audit Logs) for the full exposure window to rule out data exfiltration via authenticated cloud remotes that were accessible through the unauthenticated RC interface. Monitor port 5572 and rclone process execution continuously for 72 hours post-patch using host-level tooling (Sysmon Event ID 3 for network connections, or tcpdump on Linux) to detect any persistence mechanisms — such as attacker-added cron jobs or systemd units — that re-invoke rclone with RC flags. Any detection of rclone RC traffic post-patch should be treated as evidence of active persistence and trigger a full re-scoping of the incident.
Forensic Artifacts	Rclone RC HTTP server logs: If --log-file and --log-level=INFO or DEBUG were active, these logs record every inbound HTTP request to the RC interface including method, endpoint path (e.g., /operations/fsinfo, /core/command, /config/create), source IP, and timestamp — the primary artifact for confirming whether CVE-2026-41176 was exploited and which RC operations were invoked. | Rclone configuration file (rclone.conf) at ~/.config/rclone/rclone.conf (Linux/macOS) or %APPDATA%\rclone\rclone.conf (Windows): Modification timestamps and new remote entries added during the exposure window indicate an attacker used /config/create or /config/update RC endpoints to add attacker-controlled cloud storage destinations for data exfiltration. | Process execution logs with parent-child relationship for rclone: Sysmon Event ID 1 (ProcessCreate) or auditd execve syscall records showing processes spawned as children of the rclone process — a child shell (bash, cmd.exe, powershell.exe) or wget/curl process parented to rclone is definitive evidence of successful RCE via the /core/command RC endpoint, mapping to MITRE ATT&CK T1059. | Network flow records and pcap data for TCP port 5572: Full packet capture or NetFlow/IPFIX records showing inbound HTTP POST request volume, source IPs, payload sizes, and session durations to port 5572 during the exposure window — large outbound data volumes following RC API calls indicate active file exfiltration via /operations/copyfile or /sync/sync endpoints. | Cloud provider audit logs (AWS CloudTrail, Azure Monitor Activity Log, GCP Cloud Audit Logs) for service accounts or access keys stored in rclone.conf: Attacker-initiated sync operations through the unauthenticated RC interface would authenticate to cloud providers using credentials already stored in rclone.conf, making cloud-side audit logs the authoritative evidence source for data exfiltration scope and destination.

Per-Action IR Details

Step 1: Containment — Immediately identify all systems running Rclone below version 1.73.5 using your asset inventory or endpoint management tooling. If the Rclone RC interface (default port 5572) is enabled and exposed to any network segment beyond localhost, block inbound access to that port at the host firewall or network perimeter now, before patching is complete. Disable the RC interface entirely if it is not operationally required (remove --rc or --rc-addr flags from startup scripts and service definitions).

NIST Phase: Containment

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

Controls: NIST IR-4 (Incident Handling), NIST CM-7 (Least Functionality), NIST SC-7 (Boundary Protection), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory)

Compensating: For teams without enterprise asset management: run `find / -name 'rclone' -type f 2>/dev/null` on Linux hosts or `Get-Command rclone | Select-Object Source` in PowerShell on Windows to locate rclone binaries, then pipe to `xargs -I{} {} version` to extract version strings. Block port 5572 immediately with `iptables -I INPUT -p tcp --dport 5572 -j DROP` (Linux) or `netsh advfirewall firewall add rule name='Block Rclone RC' protocol=TCP dir=in localport=5572 action=block` (Windows). Use osquery with `SELECT * FROM processes WHERE cmdline LIKE '%--rc%'` to enumerate any running rclone instances with RC enabled across hosts where osquery is deployed.

Evidence: BEFORE blocking port 5572, capture active network connections to it with `ss -tnp sport = :5572` (Linux) or `netstat -ano | findstr :5572` (Windows) and record all source IPs and associated PIDs. Capture process trees with `ps auxf | grep rclone` or `Get-WmiObject Win32_Process | Where-Object {$_.Name -eq 'rclone.exe'} | Select-Object ProcessId, CommandLine, ParentProcessId` to document --rc flags and launch context. Preserve `/etc/systemd/system/` unit files, cron entries (`crontab -l`), and Windows Task Scheduler exports referencing rclone before modification. Snapshot active rclone configuration files at `~/.config/rclone/rclone.conf` or `%APPDATA%\rclone\rclone.conf` — these will reveal configured cloud remotes that may have been accessed during exploitation.

Step 2: Detection — Query endpoint logs, process execution logs, and network flow data for Rclone RC interface activity. Look for inbound HTTP connections to port 5572 (or any custom RC port configured in your environment) originating from unexpected source IPs. Search command execution logs for rclone processes spawned with --rc, --rc-addr, or --rc-no-auth flags. Review server access logs for POST requests to /operations/fsinfo or other RC API endpoints from unauthorized sources. Check for anomalous file system operations or data transfers initiated around the time of any suspicious RC activity.

NIST Phase: Detection Analysis

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

Controls: NIST IR-4 (Incident Handling), NIST IR-5 (Incident Monitoring), NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST SI-4 (System Monitoring), CIS 8.2 (Collect Audit Logs)

Compensating: Deploy Sysmon with a config including ProcessCreate (Event ID 1) filters on `rclone.exe` or `rclone` with CommandLine containing `--rc`, `--rc-addr`, or `--rc-no-auth`. Query collected Sysmon logs with: `Get-WinEvent -LogName 'Microsoft-Windows-Sysmon/Operational' | Where-Object {$_.Message -match 'rclone' -and $_.Message -match '--rc'}`. On Linux, search auditd logs with `ausearch -c rclone` or grep `/var/log/syslog` for rclone invocations. For network detection without a SIEM, use Wireshark or `tcpdump -i any -nn 'tcp port 5572' -w rclone_rc_capture.pcap` to capture RC traffic and inspect POST bodies for `/operations/fsinfo`, `/operations/copyfile`, `/sync/sync`, or `/core/command` endpoint calls — the last of which maps to MITRE ATT&CK T1059 (Command and Scripting Interpreter) if used to execute arbitrary commands. Apply the public Sigma rule for Rclone RC exploitation if available in your ruleset.

Evidence: Collect rclone's own log output (enabled with `--log-file=/path/to/rclone.log --log-level=INFO` or DEBUG) — this will contain HTTP request records including source IP, endpoint called, and operation parameters for every RC API call made during the exposure window. Capture web server or reverse proxy access logs if rclone RC was fronted by nginx/Apache, filtering for POST requests to `/operations/`, `/sync/`, `/core/`, and `/config/` paths. On Linux, extract auditd syscall records (`ausearch -sc execve --start today`) for process executions spawned as children of the rclone process (parent PID correlation), which would indicate successful RCE via CVE-2026-41176. Preserve netflow or firewall connection logs showing all source IPs that reached port 5572 during the exposure window for threat actor IP attribution.

Step 3: Eradication — Upgrade all Rclone installations to version 1.73.5 or later using your standard software deployment process. Update package manager definitions (apt, yum, brew, Chocolatey) and pull the latest release from the official Rclone repository (https://rclone.org/downloads/). For containerized deployments, rebuild images with the patched base version. After upgrade, enforce RC authentication on any remaining RC-enabled instances using --rc-user and --rc-pass flags, or bind the RC interface exclusively to localhost (--rc-addr 127.0.0.1:5572) where remote access is not required.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST SI-2 (Flaw Remediation), NIST CM-7 (Least Functionality), NIST IR-4 (Incident Handling), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management), CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: For Linux hosts without a centralized patch manager: `curl https://downloads.rclone.org/rclone-current-linux-amd64.zip -o rclone.zip && unzip rclone.zip && sudo cp rclone-*-linux-amd64/rclone /usr/local/bin/ && sudo chown root:root /usr/local/bin/rclone && sudo chmod 755 /usr/local/bin/rclone`. For Windows: `choco upgrade rclone` if Chocolatey is present, otherwise download v1.73.5 directly from rclone.org and replace the binary. For containerized deployments, update the Dockerfile `FROM` or explicit rclone install line to pin `rclone==1.73.5`, rebuild with `docker build --no-cache`, and redeploy. After patching, audit all rclone startup scripts (`grep -r -- '--rc-no-auth' /etc/systemd/ /etc/cron* ~/.config/`) and remove any `--rc-no-auth` flag — its presence indicates intentional unauthenticated RC exposure and must be treated as a misconfiguration finding independent of the CVE patch.

Evidence: Before executing the upgrade, collect a binary hash of the existing rclone binary (`sha256sum /usr/local/bin/rclone` or `Get-FileHash 'C:\rclone\rclone.exe'`) and preserve it for chain-of-custody documentation. If exploitation is suspected, perform a memory dump of any running rclone process (`procdump -ma ` on Windows or `gcore ` on Linux) before termination, as successful RCE via the RC interface may have injected or spawned child processes whose artifacts will be lost after eradication. Preserve the full rclone configuration file (`rclone.conf`) as evidence of configured cloud remotes — if an attacker used `/operations/copyfile` or `/sync/sync` RC endpoints, they may have exfiltrated data to an attacker-controlled remote added to this config.

Step 4: Recovery — Verify the installed Rclone version on all endpoints with rclone version and confirm output shows 1.73.5 or later. Re-enable any RC interfaces that were disabled for containment only after authentication controls are confirmed in place. Monitor port 5572 and RC-related process activity for 72 hours post-patch for any signs of residual exploitation or persistence. Review Rclone logs for operations executed during the exposure window to assess whether unauthorized file system access occurred.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST IR-4 (Incident Handling), NIST SI-7 (Software, Firmware, and Information Integrity), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-11 (Audit Record Retention), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: Script version verification across all hosts: `for host in $(cat hosts.txt); do ssh $host 'rclone version | head -1'; done` on Linux, or use PowerShell remoting `Invoke-Command -ComputerName (Get-Content hosts.txt) -ScriptBlock { rclone version }`. Validate RC authentication is enforced by attempting an unauthenticated curl request post-patch: `curl -s http://127.0.0.1:5572/rc/noop` — a 401 response confirms auth is active; a 200 response indicates `--rc-no-auth` is still present and must be corrected immediately. For the 72-hour monitoring window without a SIEM, schedule a cron job every 15 minutes: `*/15 * * * * ss -tnp sport = :5572 >> /var/log/rclone_rc_monitor.log 2>&1` to capture any new connections to the RC port.

Evidence: During the exposure window review, parse rclone log files for RC API calls to `/operations/copyfile`, `/operations/movefile`, `/sync/sync`, `/sync/copy`, `/core/command`, and `/config/create` — these endpoints represent the highest-impact operations an unauthenticated attacker could invoke via CVE-2026-41176. Cross-reference source IPs from rclone logs against cloud provider access logs (AWS CloudTrail, Azure Monitor, GCP Audit Logs) for the same timeframe to determine whether attacker-initiated rclone sync operations resulted in data exfiltration to external cloud storage. Check the rclone configuration file modification timestamp (`stat ~/.config/rclone/rclone.conf`) — a timestamp during the exposure window with new remote entries is strong evidence of attacker persistence via added cloud exfiltration destinations.

Step 5: Post-Incident — Conduct an inventory audit to establish a current baseline of all Rclone deployments, versions, and RC interface configurations across the environment. Add Rclone to your vulnerability management scanning scope if not already present. Implement a policy requiring RC interface authentication and localhost binding by default in all Rclone service configurations. Review network segmentation to ensure developer and data pipeline tooling with RC-style management interfaces is not directly reachable from

untrusted networks. Document exposure window and affected systems for internal reporting.

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 RA-3 (Risk Assessment), NIST CM-7 (Least Functionality), NIST SI-5 (Security Alerts, Advisories, and Directives), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 4.2 (Establish and Maintain a Secure Configuration Process for Network Infrastructure)

Compensating: Build a rclone-specific configuration baseline using osquery: `SELECT pid, cmdline, cwd, uid FROM processes WHERE cmdline LIKE '%rclone%'` scheduled as a recurring query pack. For the software inventory, use `find / -name 'rclone' -exec {} version \; 2>/dev/null | grep -E 'rclone v[0-9]'` across all hosts and feed output into a CSV for tracking. Codify the RC security policy as a shell script compliance check: `rclone config show | grep -E 'rc_no_auth|rc_addr'` — flag any instance where `rc_no_auth = true` or `rc_addr` is not bound to 127.0.0.1. Create a YARA rule targeting rclone binary versions below 1.73.5 based on version string patterns embedded in the binary for future asset discovery scans.

Evidence: Compile the final incident record including: (1) complete list of hosts with confirmed rclone RC exposure with exact version strings and exposure window start/end timestamps; (2) all RC API endpoint calls logged during the exposure window with source IPs; (3) rclone.conf snapshots from affected hosts showing configured cloud remotes at time of exposure; (4) cloud provider audit logs (CloudTrail, Azure Monitor, GCP Audit) covering the exposure window for any data movement operations initiated by rclone service accounts or credentials stored in rclone.conf; (5) network flow data showing volume and destinations of any outbound traffic from rclone processes during the exposure window as a data exfiltration indicator. If PII or regulated data was accessible via configured cloud remotes, flag for breach notification assessment under applicable regulatory frameworks.

Detection Guidance

Primary detection surface is the Rclone RC interface on its default port (5572) or any custom configured port. Query network flow logs for inbound TCP connections to port 5572 from sources outside localhost or approved management subnets. In SIEM, search for HTTP POST requests to paths containing /operations/fsinfo, /operations/, or /rc/ endpoints originating from unexpected IPs. Search process execution logs (Sysmon Event ID 1 on Windows; auditd execve on Linux) for rclone launched with --rc, --rc-addr, or --rc-no-auth arguments. Flag any rclone child processes (e.g., shell spawns) as high-priority indicators of successful RCE. On Linux systems, review /var/log/syslog or journalctl output for rclone service restarts or unexpected configuration changes. No confirmed IOCs (specific IPs, domains, or hashes) have been published at the time of this advisory. Monitor vendor threat feeds (e.g., Shodan, Censys, abuse.ch) and security research publications for observed exploitation activity and associated IP/domain indicators as the CVE matures in the wild.

Framework Mappings

MITRE-ATTACK

- **T1059** — Command and Scripting Interpreter
- **T1190** — Exploit Public-Facing Application
- **T1083** — File and Directory Discovery

NIST-800-53R5

- **CM-7** — Least Functionality

- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **CA-8** — Penetration Testing
- **RA-5** — Vulnerability Monitoring and Scanning
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **IA-2** — Identification and Authentication (Organizational Users)
- **SI-10** — Information Input Validation

OWASP-TOP10-2021

- **A07:2021** — Identification and Authentication Failures
- **A03:2021** — Injection

CIS-V8

- **6.3** — Require MFA for Externally-Exposed Applications
- **16.10** — Apply Secure Design Principles in Application Architectures
- **7.3** — Perform Automated Operating System Patch Management
- **7.4** — Perform Automated Application Patch Management

HIPAA-SECURITY

- **164.312(d)** — Person or Entity Authentication

SOC2-TSC

- **CC6.1** — Logical access security software, infrastructure, and architectures

ISO-27001-2022

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1059	Command and Scripting Interpreter	Execution
T1190	Exploit Public-Facing Application	Initial-Access
T1083	File and Directory Discovery	Discovery

Sources

Source	URL	Tier
	https://ccb.belgium.be/advisories/warning-two-critical-unauthentica...	T3
Rclone Critical Vulnerability Alert: Public PoC Released for ...	https://securityonline.info/rclone-rce-vulnerability-poc-disclosure...	T3
RClone: Unauthenticated operations/fsinfo allows attacker ... - GitHub	https://github.com/advisories/GHSA-jfwf-28xr-xw6q	T3
CERT-Bund Warns of Critical rclone Flaws (CVSS 9.8) | GovPing	https://changeflow.com/govping/data-privacy-cybersecurity/rclone-ve...	T3
Rclone RC Vulnerability Exposes Auth Bypass - Advisories	https://advisory.eventussecurity.com/advisory/rclone-rc-vulnerabili...	T3
NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-41176	T1

Generated 2026-04-24 13:43 UTC by TJS Security Command Center