INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-05-10 06:16 UTC

JDownloader Supply Chain Compromise Deploys Modular Python RAT Across Windows and Linux

THREAT CAMPAIGN | **CRITICAL** | CVSS 9.5

SCC Item ID	SCC-CAM-2026-0297
Type	Threat Campaign
Severity	CRITICAL
CVSS Base Score	9.5
Affected Products	JDownloader Windows alternative installer and Linux shell installer, AppWork GmbH; C2 infrastructure: parkspringshotel[.]com, auraguest[.]lk, checkinnhotels[.]com
Published	2026-05-09T15:27:58
Discovery Source	Rss

Executive Summary

Between May 6-7, 2026, attackers compromised the official JDownloader website by exploiting an unauthenticated CMS vulnerability, silently replacing legitimate Windows and Linux installers with a Python-based remote access trojan (RAT). Any user who downloaded JDownloader during that window received malware that grants attackers full remote control of their machine. Organizations affected by this incident recommend complete OS reinstallation, making this a high-impact incident for any organization whose staff downloaded JDownloader during the exposure window.

Technical Analysis

Threat actors exploited an unauthenticated access control flaw (CWE-284) in JDownloader's CMS to modify download links without requiring server-level access. Legitimate Windows alternative and Linux shell installers were replaced with heavily obfuscated payloads delivering a Python-based remote access trojan (RAT). The RAT provides full remote code execution (RCE) across both platforms and communicates with C2 infrastructure at parkspringshotel[.]com, auraguest[.]lk, and checkinnhotels[.]com. No CVE has been assigned. CWE mapping: CWE-284 (unauthenticated CMS access), CWE-494 (download without integrity check). MITRE ATT&CK coverage includes T1195.002 (Compromise Software Supply Chain), T1059.006 (Python execution), T1071.001 (C2 over HTTP/S), T1027 (obfuscation), T1036.005 (masquerading), T1543.002 (systemd service persistence on Linux), T1546.004 (Unix shell profile persistence), T1548.001 (setuid/setgid abuse), T1608.001 (staged payload), and T1132 (data encoding). No patch exists; the attack vector was the distribution channel itself, not the application binary. Full OS reinstallation is required per guidance from incident response analysts. Installer integrity verification was absent, which allowed the substitution to go undetected. No CVSS vector

available; severity is editorial based on supply chain scope and required OS reinstallation. Threat actor attribution is unknown.

Action Checklist

1. Containment: Immediately block outbound connections to parkspringshotel[.]com, auraguest[.]lk, and checkinnhotels[.]com at the perimeter firewall and DNS resolver. Isolate any host that installed JDownloader between May 6-7, 2026 from the network pending investigation.
2. Detection: Query EDR and endpoint logs for Python interpreter execution spawned from a JDownloader installer process, new systemd services or cron entries created around May 6-7, 2026, and outbound DNS/HTTP connections to the three identified C2 domains. Search SIEM for T1059.006 (Python script execution) events on hosts where JDownloader was recently installed. Check file integrity on Linux systems for modified shell profile files (~/.bashrc, ~/.profile, /etc/profile.d/) consistent with T1546.004.
3. Eradication: For confirmed compromised hosts, perform full OS reinstallation as recommended by incident response best practices. Do not attempt to remove only the RAT - the modular design and interconnected persistence mechanisms mean partial removal will leave attacker footholds in place. Re-download JDownloader only from the official source after confirming the site has been remediated and verify installer hash against the vendor's published checksums.
4. Recovery: After reinstallation, verify no persistence mechanisms remain by reviewing scheduled tasks, startup entries, systemd services, and shell profiles on rebuilt hosts. Monitor rebuilt systems for 30 days for anomalous outbound connections. Rotate credentials stored on or accessible from any affected host, including saved browser credentials and SSH keys.
5. Post-Incident: This attack exploited the absence of installer integrity verification (CWE-494). Implement software download policies requiring hash verification before execution of any installer. Evaluate whether your software allowlisting policy covers third-party download tools. Review CMS and web infrastructure for similar unauthenticated modification vulnerabilities across your managed or vendor-hosted properties.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to executive leadership, legal counsel, and potentially relevant data protection authorities immediately if forensic analysis confirms the Python RAT achieved persistent access to hosts storing PII, PHI, financial data, or credentials with access to regulated systems, or if more than 10 internal hosts are confirmed compromised, triggering breach notification assessment under applicable regulations (GDPR 72-hour window, HIPAA 60-day window, state breach notification laws).
Recovery Notes	Reinstall only from clean OS media — do not restore from any backup image created after May 6, 2026, as backups may contain the trojanized JDownloader installer or active RAT persistence. After rebuilding, monitor all egress traffic from recovered hosts for a minimum of 30 days using firewall logs or Sysmon Event ID 3 (Network Connection), specifically alerting on any Python process initiating outbound connections or any DNS resolution of the three identified C2 domains. Treat any SSH key, browser-saved credential, or API token accessible from a confirmed compromised host as fully compromised and rotate before restoring the host to production use.

Forensic Artifacts	Trojanized JDownloader installer binary: preserve SHA256 hash and binary image from any host that received the May 6–7 download; compare hash against AppWork GmbH's pre-compromise installer hash to confirm tampering (T1195.002 — Supply Chain Compromise artifact). | Dropped Python RAT modules: search for .py and .pyc files in %TEMP%, %APPDATA%\Local, %APPDATA%\Roaming (Windows) and /tmp, /var/tmp, ~/.config, ~/.local/share, and the JDownloader install directory (Linux) — these files contain the modular RAT components and C2 communication logic. | Persistence artifacts specific to this RAT: malicious systemd .service files in /etc/systemd/system/ or ~/.config/systemd/user/ with creation timestamps in the May 6–7 window (Linux); modified ~/.bashrc, ~/.profile, or /etc/profile.d/*.sh files containing Python launcher stubs (T1546.004); Windows Registry Run keys or Scheduled Tasks referencing Python executables from non-standard paths. | Network forensics: PCAP captures or firewall/proxy logs showing HTTP or HTTPS C2 beaconing to parkspringshotel[.]com, auraguest[.]lk, or checkinnhotels[.]com — capture full URI paths and User-Agent strings from the RAT's C2 communication to characterize protocol and beacon interval. | Memory forensic image (WinPmem/LiME): live memory from an actively infected host will contain the Python RAT process in-memory with decrypted C2 configuration, active network socket handles, and any in-memory command execution results — critical for recovering C2 configuration that may not be present on disk.

Per-Action IR Details

Containment — Immediately block outbound connections to parkspringshotel[.]com, auraguest[.]lk, and checkinnhotels[.]com at the perimeter firewall and DNS resolver. Isolate any host that installed JDownloader between May 6–7, 2026 from the network pending investigation.

NIST Phase: Containment

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

Controls: NIST IR-4 (Incident Handling), NIST SC-7 (Boundary Protection), NIST SI-3 (Malicious Code Protection), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices)

Compensating: On Windows hosts, run: netsh advfirewall firewall add rule name='Block JDL C2' dir=out action=block remoteip=. On Linux hosts: iptables -A OUTPUT -d -j DROP. Block at DNS resolver by adding NXDOMAIN responses for all three C2 FQDNs in /etc/hosts (Windows: C:\Windows\System32\drivers\etc\hosts) pointing to 0.0.0.0. Resolve current IPs using: for d in parkspringshotel.com auraguest.lk checkinnhotels.com; do dig +short $d; done — capture before blocking. Physically unplug network cable or disable Wi-Fi adapter on suspected hosts pending triage.

Evidence: Before isolating, capture a full memory image using WinPmem (Windows) or LiME kernel module (Linux) to preserve the live RAT process tree and any in-memory Python bytecode or C2 connection state. Run netstat -anop (Linux) or netstat -ano (Windows) and record all ESTABLISHED/TIME_WAIT connections to the three C2 domains. On Linux, run ss -tulnp and lsof -i to capture open sockets tied to the Python RAT process. Capture DNS cache: ipconfig /displaydns (Windows) or journalctl -u systemd-resolved (Linux) to confirm C2 resolution history. Document exact JDownloader installation timestamp from file system metadata: stat ~/.local/share/applications/JDownloader* (Linux) or dir /T:C 'C:\Users*\AppData\Local\JDownloader*' (Windows).

Detection — Query EDR and endpoint logs for Python interpreter execution spawned from a JDownloader installer process, new systemd services or cron entries created around May 6–7, 2026, and outbound DNS/HTTP connections to the three identified C2 domains. Search SIEM for T1059.006 (Python script execution) events on hosts where JDownloader was recently installed. Check file integrity on Linux systems for modified shell profile files (~/.bashrc, ~/.profile, /etc/profile.d/) consistent with T1546.004.

NIST Phase: Detection Analysis

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

Controls: NIST SI-4 (System Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-2 (Event Logging), NIST IR-5 (Incident Monitoring), CIS 8.2 (Collect Audit Logs), CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: Windows (no EDR): Deploy Sysmon with SwiftOnSecurity config; query the Microsoft-Windows-Sysmon/Operational log for Event ID 1 (Process Create) where ParentImage contains 'JDownloader' and Image ends in 'python.exe' or 'pythonw.exe'. PowerShell: Get-WinEvent -LogName 'Microsoft-Windows-Sysmon/Operational' | Where-Object {$_.Message -match 'python' -and $_.Message -match 'JDownloader'}. Linux (no EDR): Run find /etc/systemd/system /etc/cron* /var/spool/cron -newer /tmp/ref_date -ls (create ref_date with touch -t 202605060000 /tmp/ref_date) to surface new persistence entries. Check modified shell profiles: find /home -name '.bashrc' -o -name '.profile' -newer /tmp/ref_date | xargs grep -l 'python\|http\|curl\|wget'. Use osquery: SELECT * FROM processes WHERE name LIKE '%python%' AND parent IN (SELECT pid FROM processes WHERE name LIKE '%JDownloader%'). Apply Sigma rule for T1059.006 against syslog or auditd logs. Use Wireshark or tcpdump -i any -w capture.pcap 'host parkspringshotel.com or host auraguest.lk or host checkinnhotels.com' to capture live C2 traffic on suspected hosts.

Evidence: Query Windows Security Event Log for Event ID 4688 (Process Creation) filtering on python.exe or pythonw.exe where the Creator Process Name includes the JDownloader install path (typically C:\Users\\AppData\Local\JDownloader2\). On Linux, review auditd logs (/var/log/audit/audit.log) for execve syscalls with argv containing python initiated from JDownloader install directories (/opt/JDownloader or ~/JDownloader2/). Capture crontab -l for all users and diff against known-good baselines; list /etc/systemd/system/*.service files with creation timestamps in the May 6–7 window. Collect ~/.bashrc, ~/.profile, and all files under /etc/profile.d/ and hash them (sha256sum) for comparison against pre-compromise state or fresh install reference. Review web proxy or DNS logs for queries to parkspringshotel[.]com, auraguest[.]lk, and checkinnhotels[.]com originating from any internal host.

Eradication — For confirmed compromised hosts, perform full OS reinstallation as recommended by the vendor. Do not attempt to remove only the RAT — the modular design and depth of compromise make partial remediation unreliable. Re-download JDownloader only from the official source after confirming the site has been remediated and verify installer hash against the vendor's published checksums.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST IR-4 (Incident Handling), NIST SI-2 (Flaw Remediation), NIST SI-7 (Software, Firmware, and Information Integrity), NIST CM-6 (Configuration Settings), CIS 2.2 (Ensure Authorized Software is Currently Supported), CIS 2.3 (Address Unauthorized Software)

Compensating: Before wiping, use WinPmem or LiME to take a final forensic image for evidence preservation (NIST 800-61r3 §3.3 evidence retention). Boot from a trusted USB OS image (e.g., System Rescue CD or Windows PE) to perform the reinstall, bypassing any bootkit persistence the modular RAT may have installed. Post-reinstall, verify the JDownloader installer hash before execution: Windows: certutil -hashfile JDownloader2Setup_x64.exe SHA256 and compare against AppWork GmbH's published checksum on their official site. Linux: sha256sum JD2_Setup_x64.sh and compare. If AppWork has not published new checksums, delay reinstallation of JDownloader until confirmed. Use a clean OS image from vendor media — do not restore from a backup taken after May 6, 2026.

Evidence: Before wiping, preserve: full disk image using dc3dd or FTK Imager for forensic retention; a copy of all Python scripts dropped by the installer (search for .py files in %TEMP%, %APPDATA%, /tmp, /var/tmp, ~/.config, and the JDownloader install directory); any compiled Python artifacts (.pyc) that reveal RAT module names and C2 communication logic; the original trojanized installer binary (preserve hash and binary for malware analysis and law enforcement if needed); all persistence artifacts (malicious systemd .service files, cron entries, modified shell profiles) before overwriting the disk. Document the full file tree of the JDownloader install directory with timestamps using: find /opt/JDownloader2 -type f -printf '%T+ %p\n' | sort (Linux) or dir /S /T:C 'C:\Users\\AppData\Local\JDownloader2' (Windows).

Recovery — After reinstallation, verify no persistence mechanisms remain by reviewing scheduled tasks, startup entries, systemd services, and shell profiles on rebuilt hosts. Monitor rebuilt systems for 30 days for anomalous outbound connections. Rotate credentials stored on or accessible from any affected host, including saved browser credentials and SSH keys.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST IR-4 (Incident Handling), NIST AC-2 (Account Management), NIST IA-5 (Authenticator Management), NIST SI-4 (System Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 5.3 (Disable Dormant Accounts), CIS 6.2 (Establish an Access Revoking Process)

Compensating: Post-reinstall persistence verification: Windows: schtasks /query /fo LIST /v | findstr /i 'python\|JDownloader' and Get-ChildItem HKCU:\Software\Microsoft\Windows\CurrentVersion\Run,HKLM:\Software\Microsoft\Windows\CurrentVersion\Run | Select-Object -ExpandProperty Property. Linux: systemctl list-units --type=service --state=enabled | grep -v 'standard-package-name' and crontab -l for each user account. For credential rotation, enumerate all SSH keys on compromised hosts: find / -name 'id_rsa' -o -name 'id_ed25519' 2>/dev/null and revoke any keys whose public counterpart appears in remote authorized_keys files. Export and review Chrome/Firefox saved passwords from the compromised profile directory before wiping — if credentials were stored, treat all associated accounts as compromised. Use Sysmon Event ID 3 (Network Connection) on rebuilt hosts for 30 days, alerting on any outbound connection to the three C2 domains or any Python process making outbound HTTP/HTTPS connections.

Evidence: Before declaring recovery complete, document: output of schtasks /query (Windows) and systemctl list-units (Linux) from the rebuilt host as a clean baseline; SHA256 hashes of newly installed JDownloader binaries from the official post-remediation source; list of all credentials rotated (account names, systems — not passwords) as an audit trail per NIST AU-11 (Audit Record Retention) requirements; network flow logs or firewall logs from the 30-day monitoring window showing zero egress to the three C2 domains. Capture browser credential store locations on rebuilt hosts to confirm no credential data was migrated from the compromised profile: %LOCALAPPDATA%\Google\Chrome\User Data\Default\Login Data (Windows) or ~/.config/google-chrome/Default/Login Data (Linux).

Post-Incident — This attack exploited the absence of installer integrity verification (CWE-494). Implement software download policies requiring hash verification before execution of any installer. Evaluate whether your software allowlisting policy covers third-party download tools. Review CMS and web infrastructure for similar unauthenticated modification vulnerabilities across your managed or vendor-hosted properties.

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 SI-2 (Flaw Remediation), NIST SI-7 (Software, Firmware, and Information Integrity), NIST RA-3 (Risk Assessment), NIST CM-6 (Configuration Settings), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: Implement installer hash verification via policy: create a one-page SOP requiring analysts to run certutil -hashfile SHA256 (Windows) or sha256sum (Linux) and compare against the vendor's official checksum page before any installer is executed — enforce via change management ticket requirement. For software allowlisting without budget: configure Windows Software Restriction Policies (SRP) or AppLocker free tier to block execution of unsigned Python interpreters from user-writable directories (%TEMP%, %APPDATA%, Downloads). On Linux, implement a simple pre-execution wrapper script that calls sha256sum and aborts if the hash is not in a local trusted-hashes file. Audit all managed CMS platforms (WordPress, Joomla, Drupal, etc.) for unauthenticated file modification vulnerabilities using WPScan (free) or equivalent; schedule quarterly reviews. Add JDownloader and similar third-party download utilities to your software inventory per CIS 2.1 and formally evaluate whether they are authorized.

Evidence: Compile and preserve the full incident timeline documenting: first confirmed trojanized download timestamp (May 6, 2026), C2 domain registration and hosting metadata (WHOIS/passive DNS for parkspringshotel[.]com, auraguest[.]lk, checkinnhotels[.]com), the CMS vulnerability class exploited by attackers to replace the AppWork GmbH installers (document the specific CMS and vulnerability type once disclosed by AppWork), MITRE ATT&CK technique coverage gaps identified (T1059.006, T1546.004, T1195.002 — Supply Chain Compromise), and a list of all internal hosts that downloaded JDownloader during the May 6–7 window sourced from proxy/DNS logs. This evidence package supports lessons-learned review, potential law enforcement referral, and regulatory breach notification

assessment if PII was accessible on compromised hosts.

Detection Guidance

Primary IOCs: outbound connections to parkspringshotel[.]com, auraguest[.]lk, or checkinnhotels[.]com. Query DNS logs and proxy logs for any resolution or HTTP/S request to these domains. In EDR telemetry, look for Python processes (python.exe on Windows, python3 on Linux) spawned by or shortly after a JDownloader installer execution between May 6-7, 2026. On Linux, inspect /etc/systemd/system/ and user-level systemd unit directories for services created in that window (T1543.002). Check shell initialization files for appended entries (T1546.004). On Windows, review scheduled tasks and registry run keys for Python-based persistence entries. Behavioral indicators include: obfuscated Python script execution, encoded outbound payloads (T1132), and privilege escalation attempts via setuid binaries (T1548.001). YARA or file scanning should target heavily obfuscated Python scripts in temp or installer staging directories. Note: IOC confidence is based on reporting from secondary sources; if additional C2 infrastructure is identified, update firewall and DNS blocking rules accordingly.

Indicators of Compromise

Type	Value	Context	Confidence
DOMAIN	`parkspringshotel[.]com`	Identified C2 infrastructure for the Python RAT delivered via compromised JDownloader installers	HIGH
DOMAIN	`auraguest[.]lk`	Identified C2 infrastructure for the Python RAT delivered via compromised JDownloader installers	HIGH
DOMAIN	`checkinnhotels[.]com`	Identified C2 infrastructure for the Python RAT delivered via compromised JDownloader installers	HIGH

Framework Mappings

MITRE-ATTACK

- **T1546.004** — Unix Shell Configuration Modification
- **T1071.001** — Web Protocols
- **T1548.001** — Setuid and Setgid
- **T1036.005** — Match Legitimate Resource Name or Location
- **T1059.006** — Python
- **T1543.002** — Systemd Service
- **T1608.001** — Upload Malware
- **T1132** — Data Encoding
- **T1195.002** — Compromise Software Supply Chain

- **T1027** — Obfuscated Files or Information

NIST-800-53R5

- **CM-7** — Least Functionality
- **SA-9** — External System Services
- **SR-3** — Supply Chain Controls and Processes
- **SI-7** — Software, Firmware, and Information Integrity
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **AC-3** — Access Enforcement
- **CM-3** — Configuration Change Control
- **SR-2** — Supply Chain Risk Management Plan

OWASP-TOP10-2021

- **A01:2021** — Broken Access Control
- **A08:2021** — Software and Data Integrity Failures

CIS-V8

- **6.1** — Establish an Access Granting Process
- **6.2** — Establish an Access Revoking Process
- **2.5** — Allowlist Authorized Software
- **2.6** — Allowlist Authorized Libraries
- **15.1** — Establish and Maintain an Inventory of Service Providers

SOC2-TSC

- **CC6.1** — The entity implements logical access security software, infrastructure, and architectures over protected information assets
- **CC9.2** — Manages risks associated with vendors and business partners

HIPAA-SECURITY

- **164.312(a)(1)** — Access Control

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities
- **A.5.21** — Managing information security in the ICT supply chain

NIST-CSF-2

- **GV.SC-01** — Cybersecurity supply chain risk management program

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1546.004	Unix Shell Configuration Modification	Privilege-Escalation

Technique ID	Technique Name	Tactic
T1071.001	Web Protocols	Command-And-Control
T1548.001	Setuid and Setgid	Privilege-Escalation
T1036.005	Match Legitimate Resource Name or Location	Defense-Evasion
T1059.006	Python	Execution
T1543.002	Systemd Service	Persistence
T1608.001	Upload Malware	Resource-Development
T1132	Data Encoding	Command-And-Control
T1195.002	Compromise Software Supply Chain	Initial-Access
T1027	Obfuscated Files or Information	Defense-Evasion

Sources

Source	URL	Tier
Security News	https://www.bleepingcomputer.com/news/security/jdownloader-site-hac...	T3
Is the website hacked? : r/jdownloader - Reddit	https://www.reddit.com/r/jdownloader/comments/1t6goqe/is_the_websit...	T3
■■ The official JDownloader website was breached, attackers ...	https://x.com/IntCyberDigest/status/2052861247696019465	T3
Trust Hijacked: Official JDownloader Website Breached to Distribute ...	https://securityonline.info/jdownloader-website-breach-malware-inst...	T3
JDownloader Website Hacked — Malicious Installers Served to ...	https://www.cyberkendra.com/2026/05/jdownloader-website-hacked-mali...	T3

Generated 2026-05-10 06:16 UTC by TJS Security Command Center