INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-10 07:27 UTC

CVE-2026-11456: A vulnerability was identified in Chanjet CRM 1.0. This affects an unknown part of the file /tools/j...

CVE VULNERABILITY | HIGH | CVSS 7.3

SCC Item ID	SCC-CVE-2026-0287
Type	CVE Vulnerability
CVE ID	CVE-2026-11456
Severity	HIGH
CVSS Base Score	7.3
EPSS Score	0.0003 (9th percentile)
Affected Products	Chanjet CRM 1.0
Published	2026-06-07T09:16:21.673
Discovery Source	Nvd

Executive Summary

A SQL injection vulnerability in Chanjet CRM 1.0 allows unauthenticated remote attackers to manipulate the application's database through a publicly exposed PHP endpoint. Organizations running this CRM version with internet-facing deployments risk unauthorized data access, exfiltration of customer records, and potential full database compromise. No vendor patch is available following unresponsive disclosure, leaving affected organizations reliant on network-level controls until a fix is issued.

Technical Analysis

CVE-2026-11456 is a SQL injection vulnerability (CWE-89, CWE-74) in Chanjet CRM 1.0, located in /tools/jxf_dump_systable.php. The 'gblOrgID' parameter in the HTTP GET request handler lacks proper input sanitization, enabling an attacker to inject arbitrary SQL via crafted GET requests. The attack is remotely exploitable, requires no authentication, and a public exploit is available. CVSS base score: 7.3 (High). EPSS score: 0.0003 (9th percentile, low observed exploitation in the wild as of scoring date). MITRE ATT&CK technique: T1190 (Exploit Public-Facing Application). The vendor did not respond to responsible disclosure; no official patch exists. Affected version: Chanjet CRM 1.0 only. Source quality score: 0.632, primarily T3 sources supplementing the NVD T1 entry; consider technical details provisional until the NVD record includes vendor-confirmed details or public exploit analysis matures.

Action Checklist

1. Step 1: Containment, Immediately restrict external access to /tools/jxf_dump_systable.php on all Chanjet CRM 1.0 instances. Apply WAF rules blocking GET requests containing SQL metacharacters (single quotes, UNION, SELECT, --, ;) targeting this endpoint. If internet-facing, consider taking the endpoint offline until remediation is complete. Reference: NIST AC-4 (Information Flow Enforcement), enforce approved authorizations controlling information flow between connected systems.
2. Step 2: Detection, Query web server and application logs for GET requests to /tools/jxf_dump_systable.php containing anomalous 'gblOrgID' values, particularly strings with SQL keywords (UNION, SELECT, FROM, WHERE, --), encoded characters (%27, %3D), or unusually long parameter values. Review database query logs for unexpected UNION-based or error-based SQL patterns originating from the application layer. Reference: NIST AU-6 (Audit Record Review, Analysis, and Reporting); CIS 8.2 (Collect Audit Logs).
3. Step 3: Eradication, No vendor patch is available. Apply input validation and parameterized query enforcement at the application or WAF layer for the gblOrgID parameter. If the application permits, disable or remove /tools/jxf_dump_systable.php entirely if it is not required for production operations. Consider migrating off Chanjet CRM 1.0 if the vendor remains unresponsive. Reference: NIST SI-10 (Information Input Validation); for unpatched third-party applications, escalate to procurement and vendor management per NIST IA-4 (Third-Party Assessment).
4. Step 4: Recovery, After WAF or network controls are in place, validate that requests to the affected endpoint return appropriate blocks or 403 responses. Review database access logs for signs of prior exploitation, including unexpected data exports, schema queries, or unusual row counts in CRM tables. Rotate database credentials used by the Chanjet CRM application as a precaution. Reference: NIST AC-2 (Account Management) for credential scope review; D3-CRO (Credential Rotation).
5. Step 5: Post-Incident, Conduct a review of all PHP endpoints in the Chanjet CRM installation for similar unsanitized parameter handling. Formalize a vendor responsiveness requirement in procurement and third-party risk assessments going forward. Review WAF coverage for all CRM and customer-data-handling applications. Reference: NIST AC-6 (Least Privilege), ensure database accounts used by CRM have only required permissions; CIS 7.1 (Establish and Maintain a Vulnerability Management Process).

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to legal, privacy officer, and executive leadership if database query logs or binary logs confirm successful UNION-based data extraction from Chanjet CRM customer tables, as exfiltrated PII or business records likely triggers breach notification obligations under applicable data protection regulations (GDPR, CCPA, or state equivalents); additionally escalate if the team lacks WAF capability to block the endpoint and the instance is confirmed internet-facing with no compensating network control in place.

Recovery Notes	After WAF or file-level blocking is confirmed via external curl validation, conduct a row-count and last-modified-timestamp audit across all Chanjet CRM customer-facing database tables against the most recent clean backup to detect any data deletion, exfiltration, or schema tampering that occurred prior to containment. Monitor the MySQL general query log and web server access logs continuously for a minimum of 72 hours post-containment for any attacker pivot attempts using credentials or session tokens that may have been harvested through the SQLi prior to blocking. If the vendor remains unresponsive beyond 30 days, treat Chanjet CRM 1.0 as an unsupported application per CIS 2.2 (Ensure Authorized Software is Currently Supported) and initiate formal migration planning.
Forensic Artifacts	Web server access log (/var/log/apache2/access.log or /var/log/nginx/access.log): contains the raw GET request history to /tools/jxf_dump_systable.php including full query strings with gblOrgID payloads, attacker source IPs, timestamps, and HTTP response codes — the primary artifact for reconstructing the attack timeline and confirming exploitation | MySQL/MariaDB general query log (/var/log/mysql/general.log, enabled via SET GLOBAL general_log=ON): captures all SQL statements passed from the Chanjet CRM PHP layer to the database, including any UNION SELECT, information_schema enumeration, or table-dump queries injected through the gblOrgID parameter | MySQL binary log (binlog, located per 'SHOW VARIABLES LIKE log_bin_basename'): records all data-modification and read events at the database engine level, enabling reconstruction of any rows read or exported via successful SQL injection payloads even if the general query log was not enabled at the time of the attack | PHP error log (/var/log/php_errors.log or as configured in php.ini): records SQL syntax errors returned by failed injection probes against gblOrgID, confirming error-based reconnaissance activity and providing a chronological probe sequence even before successful exploitation | Network packet capture (pcap via tcpdump -i [interface] -w chanjet_capture.pcap 'host [crm-server-ip]'): captures inbound HTTP GET requests containing raw SQL payloads to /tools/jxf_dump_systable.php and any anomalous outbound data volume from the CRM host that may indicate active exfiltration via HTTP response body or DNS tunneling during the exploitation window

Per-Action IR Details

Step 1: Containment — Immediately restrict external access to /tools/jxf_dump_systable.php on all Chanjet CRM 1.0 instances. Apply WAF rules blocking GET requests containing SQL metacharacters (single quotes, UNION, SELECT, --, ;) targeting this endpoint. If internet-facing, consider taking the endpoint offline until remediation is complete. Reference: NIST AC-4 (Information Flow Enforcement) — enforce approved authorizations controlling information flow between connected systems.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy: isolate affected systems or endpoints to prevent further exploitation while preserving forensic state

Controls: NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices)

Compensating: On Apache: add 'RedirectMatch 403 ^/tools/jxf_dump_systable\.php$' to .htaccess or VirtualHost config. On nginx: add 'location = /tools/jxf_dump_systable.php { return 403; }'. For WAF-less environments, deploy ModSecurity (free, open-source) with the OWASP CRS ruleset targeting SQL metacharacter patterns (%27, UNION, SELECT, --, %3D) on this specific URI. Verify with: curl -v 'https://[host]/tools/jxf_dump_systable.php?gblOrgID=1%27' and confirm 403 response.

Evidence: Before blocking, capture a full snapshot of the current web server access log (/var/log/apache2/access.log or /var/log/nginx/access.log) and the application's PHP error log to establish a pre-containment exploitation baseline. Document all source IPs that have reached /tools/jxf_dump_systable.php with non-numeric or encoded gblOrgID values. Preserve raw log files with sha256sum hashes before any rotation or truncation occurs.

Step 2: Detection — Query web server and application logs for GET requests to /tools/jxf_dump_systable.php containing anomalous 'gblOrgID' values, particularly strings with SQL keywords (UNION, SELECT, FROM, WHERE, --), encoded characters (%27, %3D), or unusually long parameter values. Review database query logs for unexpected UNION-based or error-based SQL patterns originating from the application layer. 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: correlate log evidence to determine scope, attack timeline, and whether exploitation resulted in data access

Controls: NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST AU-2 (Event Logging), NIST AU-3 (Content Of Audit Records), CIS 8.2 (Collect Audit Logs)

Compensating: Run the following grep against Apache/nginx access logs to identify exploitation attempts against this specific endpoint: grep 'jxf_dump_systable' /var/log/apache2/access.log | grep -iE "(UNION|SELECT|FROM|WHERE|--|%27|%3D|%20OR%20|0x[0-9a-f]+)" > sqli_hits.txt. For MySQL general query log (enable with 'SET GLOBAL general_log = ON; SET GLOBAL general_log_file = "/var/log/mysql/general.log";'), run: grep -i 'UNION\|information_schema\|SLEEP\|BENCHMARK' /var/log/mysql/general.log. If Sysmon is deployed on the CRM host, query Event ID 3 (Network Connection) for outbound connections from the PHP-FPM or httpd process to unexpected external IPs, which may indicate data exfiltration via DNS or HTTP.

Evidence: Collect and hash: (1) full web server access logs covering at least 30 days prior to detection, filtering on /tools/jxf_dump_systable.php requests; (2) MySQL or MariaDB slow query log and general query log for queries referencing UNION, information_schema.tables, or table dump patterns consistent with Chanjet CRM's schema; (3) PHP error log for SQL syntax errors generated by malformed gblOrgID inputs, which confirm error-based injection probing; (4) network flow data (netflow or pcap via Wireshark/tcpdump) showing outbound data volume from the CRM host during the suspected exploitation window.

Step 3: Eradication — No vendor patch is available. Apply input validation and parameterized query enforcement at the application or WAF layer for the gblOrgID parameter. If the application permits, disable or remove /tools/jxf_dump_systable.php entirely if it is not required for production operations. Consider migrating off Chanjet CRM 1.0 if the vendor remains unresponsive. Reference: no mapped control for vendor-unpatched third-party application remediation; NIST SI-10 (Information Input Validation) is the relevant family but is not in the provided knowledge base — citing only what is confirmed above.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication: remove the vulnerability or threat mechanism from the environment; where vendor patching is unavailable, document compensating controls and residual risk

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

Compensating: If PHP source is accessible, rename or chmod 000 the file: 'chmod 000 /var/www/html/tools/jxf_dump_systable.php' and verify the file is unreachable. If source modification is permitted, wrap the gblOrgID parameter in a strict integer cast at the entry point: '$gblOrgID = (int)$_GET["gblOrgID"];' — this eliminates string-based SQL injection payloads without requiring a full patch. Deploy ModSecurity rule targeting this endpoint: 'SecRule REQUEST_URI "@contains jxf_dump_systable" "id:100001,phase:1,deny,status:403,msg:CVE-2026-11456 block"'. Document all changes with before/after file hashes for the forensic record.

Evidence: Before any file modification, capture: (1) a full filesystem hash of /tools/jxf_dump_systable.php using sha256sum and store in the incident record; (2) a copy of the PHP source file itself to confirm the unsanitized gblOrgID parameter handling and document the root cause; (3) a database schema snapshot (mysqldump --no-data) to establish the pre-eradication state of all tables the CRM account can access, enabling later comparison if data loss is suspected.

Step 4: Recovery — After WAF or network controls are in place, validate that requests to the affected endpoint return appropriate blocks or 403 responses. Review database access logs for signs of prior exploitation,

including unexpected data exports, schema queries, or unusual row counts in CRM tables. Rotate database credentials used by the Chanjet CRM application as a precaution. Reference: NIST AC-2 (Account Management) for credential scope review; D3-CRO (Credential Rotation).

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery: restore systems to normal operation, verify integrity of data and controls, and confirm the threat vector is closed before resuming production use

Controls: NIST AC-2 (Account Management), NIST AC-3 (Access Enforcement), NIST AC-6 (Least Privilege)

Compensating: Validate WAF/block effectiveness: run 'curl -v "https://[host]/tools/jxf_dump_systable.php?gblOrgID=1%27+UNION+SELECT+1--'" from an external IP and confirm HTTP 403 with no database response body. Rotate the Chanjet CRM database user credentials in MySQL: 'ALTER USER "chanjet_app"@"localhost" IDENTIFIED BY "[new-strong-password]"; FLUSH PRIVILEGES;'. Audit the CRM database account's current privileges with 'SHOW GRANTS FOR "chanjet_app"@"localhost";' and revoke any permissions beyond SELECT/INSERT/UPDATE on CRM-specific tables — DROP, FILE, and GRANT privileges are particularly dangerous for a SQLi-exposed account. Monitor MySQL general query log for 72 hours post-recovery for any resumption of anomalous UNION or schema-enumeration queries.

Evidence: Before credential rotation, capture: (1) current MySQL privilege grant output for the CRM database user to document pre-remediation privilege scope; (2) database row counts for all customer-facing tables (contacts, accounts, leads) compared against the most recent known-good backup to detect data deletion or exfiltration; (3) MySQL binary log (binlog) entries covering the suspected exploitation window to reconstruct any SELECT, UNION, or data-dumping queries executed against the CRM database via the vulnerable endpoint.

Step 5: Post-Incident — Conduct a review of all PHP endpoints in the Chanjet CRM installation for similar unsanitized parameter handling. Formalize a vendor responsiveness requirement in procurement and third-party risk assessments going forward. Review WAF coverage for all CRM and customer-data-handling applications. Reference: NIST AC-6 (Least Privilege) — ensure database accounts used by CRM have only required permissions; CIS 7.1 (Establish and Maintain a Vulnerability Management Process).

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: conduct lessons learned, update detection and response capabilities, and share intelligence to prevent recurrence

Controls: NIST AC-6 (Least Privilege), NIST AU-6 (Audit Record Review, Analysis, And Reporting), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: Use grep or a free DAST tool such as Nikto ('nikto -h https://[host] -Tuning 9') to enumerate all PHP files under the Chanjet CRM /tools/ and related directories for GET/POST parameters passed directly to database queries without sanitization. Write a Sigma rule targeting this endpoint pattern for future log monitoring: detect GET requests to any /tools/*.php URI containing SQL metacharacters in any parameter. Add a procurement checklist item requiring documented vendor patch SLAs and a point-of-contact for security disclosure before approving any new third-party CRM or web application — specifically noting Chanjet's unresponsive disclosure history as the trigger.

Evidence: Preserve the complete incident record including: (1) all raw log files collected during detection and analysis with sha256 hashes; (2) the original PHP source of jxf_dump_systable.php documenting the unsanitized gblOrgID parameter as root-cause evidence; (3) the full timeline of GET requests to the vulnerable endpoint from web server logs, annotated with attacker IPs and payload classifications; (4) the pre- and post-remediation MySQL privilege grants for the CRM database account to demonstrate least-privilege enforcement was implemented as a result of this incident.

Detection Guidance

Monitor web server access logs for GET requests to /tools/jxf_dump_systable.php. Flag any gblOrgID parameter values containing SQL syntax: single quotes ('), double dashes (--), UNION, SELECT, FROM,

WHERE, semicolons, or URL-encoded equivalents (%27, %2D%2D, %55%4E%49%4F%4E). Alert on unusually long parameter strings (over 50 characters) in this parameter. At the database layer, enable slow query logging and flag queries originating from the CRM application that reference information_schema, perform UNION SELECT operations, or scan unexpected tables. Baseline normal query patterns from this application before applying anomaly thresholds. Reference: NIST AU-2 (Event Logging), AU-3 (Content of Audit Records), AU-6 (Audit Record Review, Analysis, and Reporting); D3-SFA (System File Analysis) for monitoring application-layer configuration and log files.

Framework Mappings

MITRE-ATTACK

- **T1190** — Exploit Public-Facing Application

NIST-800-53R5

- **CA-8** — Penetration Testing
- **RA-5** — Vulnerability Monitoring and Scanning
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **SI-7** — Software, Firmware, and Information Integrity
- **SI-10** — Information Input Validation

OWASP-TOP10-2021

- **A03:2021** — Injection

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures

ISO-27001-2022

- **A.8.28** — Secure coding
- **A.8.8** — Management of technical vulnerabilities
- **A.5.21** — Managing information security in the ICT supply chain

SOC2-TSC

- **CC9.2** — Manages risks associated with vendors and business partners

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1190	Exploit Public-Facing Application	Initial-Access

Sources

Source	URL	Tier
nvd	https://nvd.nist.gov/vuln/detail/CVE-2026-11456	T1
CVE-2026-11456 in Chanjet - Vulnerability Database	https://vuldb.com/cve/CVE-2026-11456	T3
CVE-2026-11456: SQL Injection in Chanjet CRM - Threat Radar	https://radar.offseq.com/threat/cve-2026-11456-sql-injection-in-cha...	T3
CVE-2026-45611: Rejected CVE Entry - Not a Vulnerability	https://www.sentinelone.com/vulnerability-database/cve-2026-45611/	T3
CVE-2026-11456 - INCIBE	https://www.incibe.es/index.php/en/incibe-cert/early-warning/vulner...	T3

Generated 2026-06-10 07:27 UTC by TJS Security Command Center