INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-26 13:45 UTC

TonRAT Targets Hotel Front Desks: Blockchain C2 and Authentication Laundering Define a Campaign Security Teams Cannot Ignore

THREAT CAMPAIGN | HIGH | CVSS 7.5

SCC Item ID	SCC-CAM-2026-0578
Type	Threat Campaign
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Hospitality sector organizations (European and Asian); delivery infrastructure abused: Calendly, Google redirect URLs, Cloudflare Turnstile; legitimate runtime abused: Node.js v24.13.0; C2 resolution via TON blockchain API
Published	2026-06-26T05:27:12
Discovery Source	Rss

Executive Summary

An active phishing campaign, active since April 2026, is targeting hotel front-desk staff across Europe and Asia with a Node.js-based remote access trojan called TonRAT. Attackers abuse trusted platforms, Calendly booking links and Google redirect URLs, to pass email authentication checks, delivering malware that communicates via the TON blockchain, making standard domain blocklists ineffective. Campaign activity was reported in June 2026; organizations in the hospitality sector face credential theft, persistent access, and the operational disruption that follows a compromised front-desk workstation.

Technical Analysis

TonRAT is a Node.js-based remote access trojan delivered via spear-phishing emails (T1566.001, T1566.002) that abuse Calendly booking links and Google open redirect URLs to pass SPF/DKIM/DMARC validation. Initial execution is triggered by LNK files (T1204.002) that invoke a bundled Node.js runtime (T1059.007) to load the implant. The implant uses TON (The Open Network) blockchain API queries for C2 resolution (T1568), bypassing DNS-layer controls and static domain blocklists. Cloudflare Turnstile CAPTCHA challenges are present in the delivery chain, potentially complicating automated sandbox analysis (T1027). Persistence is established via registry run keys (T1547.001). Additional behaviors include system reconnaissance (T1082, T1016, T1033), obfuscated code delivery (T1027, T1027.003), and use of rundll32 or equivalent LOLBins for execution (T1218.011). C2 traffic rides standard HTTP/S application-layer protocols (T1071.001). Relevant

CWEs: CWE-347 (improper verification of cryptographic signature via email spoofing), CWE-693 (email security control bypass and sandbox evasion). No CVE is assigned. Campaign is assessed as ongoing; no vendor patch exists because the abuse vector is platform misuse, not a software vulnerability.

Action Checklist

1. Step 1: Containment. Block outbound HTTPS connections to the TON blockchain API endpoint (toncenter.com and any TON API resolver domains) at the perimeter firewall and proxy layer for hospitality-sector workstations. Isolate any front-desk endpoints that have executed LNK files received via email in the past 90 days. Disable Node.js if it is not an authorized runtime in your environment (CIS 2.1, Maintain Inventory of Authorized Software).
2. Step 2: Detection. Query EDR telemetry for node.exe or node processes spawned from user-writable directories or %TEMP% paths. Search email gateway logs for messages containing Calendly URLs or Google redirect URLs (google.com/url?q=) delivered to front-desk role accounts since April 2026. Review proxy/firewall logs for outbound connections to TON API endpoints. Check Windows registry run keys (HKCU\Software\Microsoft\Windows\CurrentVersion\Run) for unfamiliar Node.js-related entries (NIST SP 800-53 AU-6, Audit Record Review, Analysis, and Reporting; AU-2, Event Logging). Use MITRE ATT&CK T1568, T1547.001, and T1059.007 as hunt pivots.
3. Step 3: Eradication. Remove LNK files and Node.js runtime bundles from affected endpoints. Delete identified persistence registry run keys. Revoke and rotate credentials for any account accessed from a potentially compromised front-desk workstation (NIST SP 800-53 IA-4, Identifier Management). Block Calendly-originated and Google-redirect URLs at the email gateway. If Calendly is a business requirement, implement URL rewriting with sandboxed detonation for all Calendly links rather than blocking outright. Apply application allowlisting to prevent unauthorized Node.js execution (NIST SP 800-53 AC-3, Access Enforcement).
4. Step 4: Recovery. Validate remediation by re-running EDR sweeps for node.exe spawning from unauthorized paths 24 and 72 hours post-cleanup. Confirm registry persistence keys remain absent. Monitor outbound proxy logs for renewed TON API connections as a reinfection indicator. Re-image workstations where implant residence period cannot be bounded. Validate that email authentication controls (SPF, DKIM, DMARC) are enforced in reject mode, not monitor-only, for inbound mail (NIST SP 800-53 AC-17, Remote Access; AU-12, Audit Record Generation).
5. Step 5: Post-Incident. Conduct phishing simulation exercises targeting front-desk and reservations staff with Calendly-style lure formats. Implement role-based email filtering rules that flag or quarantine booking-platform links sent to front-desk accounts from external senders. Evaluate whether DNS-layer controls are supplemented with TLS inspection capable of detecting blockchain API queries (CIS 7.1, Establish and Maintain a Vulnerability Management Process). Review and enforce least privilege on front-desk workstations to limit blast radius of any future compromise (NIST SP 800-53 AC-6, Least Privilege; CIS 5.4, Restrict Administrator Privileges to Dedicated Administrator Accounts).

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate immediately to senior IR leadership and legal/privacy counsel if forensic evidence confirms TonRAT implant residence on any workstation with access to a Property Management System (PMS), payment card data, or guest PII, as this triggers breach notification obligations under GDPR (EU hospitality), PDPA (Asian jurisdictions), and PCI-DSS Requirement 12.10.4; escalate additionally if TON C2 beacon logs indicate active tasking or data exfiltration commands were received during the implant's residence window.
Recovery Notes	Re-image any front-desk workstation where the earliest confirmed LNK execution date cannot be established from Prefetch or Sysmon logs, as an unbounded implant residence window prevents reliable eradication confirmation for a RAT with blockchain-based C2. Post-reimage, maintain elevated proxy log monitoring for outbound connections to toncenter.com and TON API resolver domains for a minimum of 30 days to detect reinfection via any lure emails not yet identified in the email gateway review. Confirm DMARC is enforced in reject mode and that Calendly URLs are routed through sandboxed detonation before delivery to front-desk mailboxes, as the campaign's authentication laundering technique means SPF/DKIM pass status is not a reliable clean indicator.
Forensic Artifacts	node.exe Prefetch file at C:\Windows\Prefetch\NODE.EXE-.pf — confirms execution of the TonRAT Node.js runtime and timestamps first and last execution; parse with PECmd.exe to extract run count and directory of execution, which distinguishes legitimate IT-deployed Node.js from the TonRAT bundle dropped to %TEMP% | LNK file(s) in %APPDATA%\Microsoft\Windows\Recent and the original delivery path — the TonRAT LNK is the initial execution artifact; extract target path, working directory, and embedded command-line arguments using LECmd.exe (Eric Zimmerman) to reconstruct the full dropper execution chain | Windows Registry key HKCU\Software\Microsoft\Windows\CurrentVersion\Run entries referencing node.exe or .js files — TonRAT establishes persistence via Run key pointing to the dropped Node.js bundle; export and hash the key value for IOC sharing | Proxy or DNS resolver logs showing outbound queries and CONNECT requests to toncenter.com or tonapi.io — each beacon represents a C2 polling event over the TON blockchain API; the frequency and timing of these connections establishes the implant's active period and may reveal tasking commands if TLS inspection was in place | Email gateway message trace records for front-desk accounts showing inbound messages with Calendly (calendly.com) or Google redirect (google.com/url?q=) URLs that passed SPF and DKIM authentication — these records prove the authentication laundering technique succeeded and are essential for establishing the initial access timeline and identifying all potentially affected recipients

Per-Action IR Details

Step 1: Containment — Block outbound HTTPS connections to the TON blockchain API endpoint (toncenter.com and any TON API resolver domains) at the perimeter firewall and proxy layer for hospitality-sector workstations. Isolate any front-desk endpoints that have executed LNK files received via email in the past 90 days. Disable Node.js if it is not an authorized runtime in your environment (CIS 2.3 — Address Unauthorized Software).

NIST Phase: Containment

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

Controls: CIS 2.3 (Establish and Maintain a Software Inventory — Address Unauthorized Software), NIST AC-4 (Information Flow Enforcement)

Compensating: On Windows hosts without EDR, run `Get-Process node` and `netstat -ano | findstr :443` to identify live node.exe processes with established outbound connections before isolation. Use Windows Firewall (`netsh advfirewall firewall add rule name='Block TON API' dir=out action=block remoteip=`) to block TON API egress per workstation. For LNK execution history without EDR, query Prefetch files using Eric Zimmerman's PECmd.exe:

`PECmd.exe -d C:\Windows\Prefetch --csv . --csvf prefetch_output.csv` and grep for node.exe entries.

Evidence: Before isolating any front-desk endpoint, capture: (1) full RAM image using WinPmem or Magnet RAM Capture to preserve in-memory TonRAT implant and any decrypted C2 beacon state; (2) live network connections via `Get-NetTCPConnection | Where-Object {$_.State -eq 'Established'}` and `netstat -ano` — specifically looking for ESTABLISHED connections to toncenter.com or TON API resolver IPs over port 443; (3) running process tree via `Get-WmiObject Win32_Process | Select-Object Name,ProcessId,ParentProcessId,CommandLine | Export-Csv procs.csv` to capture node.exe parent-child relationships showing the LNK-spawned execution chain; (4) list of recently accessed LNK files from `%APPDATA%\Microsoft\Windows\Recent` and ShellBags before any filesystem changes.

Step 2: Detection — Query EDR telemetry for node.exe or node processes spawned from user-writable directories or %TEMP% paths. Search email gateway logs for messages containing Calendly URLs or Google redirect URLs (google.com/url?q=) delivered to front-desk role accounts since April 2026. Review proxy/firewall logs for outbound connections to TON API endpoints. Check Windows registry run keys (HKCU\Software\Microsoft\Windows\CurrentVersion\Run) for unfamiliar Node.js-related entries (AU-6 — Audit Record Review, Analysis, and Reporting; AU-2 — Event Logging). Use MITRE ATT&CK T1568, T1547.001, and T1059.007 as hunt pivots.

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)

Compensating: Without EDR, deploy Sysmon with a config that logs Event ID 1 (Process Create) and Event ID 3 (Network Connection): filter on `Image` containing `node.exe` and `CurrentDirectory` matching `%TEMP%` or `%APPDATA%`. Query collected Sysmon logs with: `Get-WinEvent -LogName 'Microsoft-Windows-Sysmon/Operational' | Where-Object {$_.Message -match 'node.exe' -and $_.Message -match 'Temp'}`. For email log hunting without a SIEM, export O365/Exchange message trace to CSV and filter with PowerShell: `Import-Csv mail_trace.csv | Where-Object {$_.RecipientAddress -match 'frontdesk' -and ($_.Subject -match 'calendly' -or $_.MessageId -match 'google.com/url')}`. Use Autoruns.exe (Sysinternals) to enumerate `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` across endpoints for node.exe or .js file references.

Evidence: This detection step reads live state but does not alter it; however, before any follow-on containment action triggered by findings, preserve: (1) Sysmon Event ID 1 logs showing node.exe `CommandLine`, `ParentImage`, and `CurrentDirectory` — the TonRAT dropper characteristically spawns node.exe from `%TEMP%\\` with the LNK as the ultimate ancestor; (2) email gateway logs showing the originating Calendly or Google redirect URL, sender domain, and SPF/DKIM authentication results — authentication pass on a malicious Calendly URL is the specific detection gap this campaign exploits; (3) proxy/firewall logs for DNS queries or CONNECT requests to `toncenter.com` or `tonapi.io` from front-desk workstation IP ranges since April 2026; (4) registry export of `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` via `reg export HKCU\Software\Microsoft\Windows\CurrentVersion\Run run_keys_backup.reg` before any remediation.

Step 3: Eradication — Remove LNK files and Node.js runtime bundles from affected endpoints. Delete identified persistence registry run keys. Revoke and rotate credentials for any account accessed from a potentially compromised front-desk workstation (D3-CRO — Credential Rotation). Block Calendly-originated and Google-redirect URLs at the email gateway if your organization has no legitimate dependency on them; if Calendly is used, implement URL rewriting with sandboxed detonation for all Calendly links. Apply application allowlisting to prevent unauthorized Node.js execution (AC-3 — Access Enforcement).

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

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

Compensating: Without an enterprise allowlisting product, use Windows Software Restriction Policies (SRP) or AppLocker (available on Windows 10/11 Pro and above at no cost) to create a deny rule for `node.exe` executed from any path outside an IT-managed directory. Export the rule via `Get-AppLockerPolicy -Effective | Export-Clixml applocker_policy.xml` for documentation. For credential rotation without a PAM tool, script bulk AD password resets for

all accounts with interactive logon history on affected workstations: `Get-ADUser -Filter * -SearchBase 'OU=FrontDesk,...' | Set-ADAccountPassword -Reset -NewPassword (ConvertTo-SecureString -AsPlainText 'TempPass!01' -Force)` followed by forced change at next logon. For LNK and Node.js bundle removal, use: `Get-ChildItem -Path C:\Users -Recurse -Include *.lnk,node.exe -ErrorAction SilentlyContinue | Remove-Item -Force`.

Evidence: Before revoking credentials or deleting persistence artifacts, capture: (1) memory image if not already acquired — credential material for PMS (Property Management System) accounts or booking platform credentials may be present in node.exe heap memory and are unrecoverable post-termination; (2) full copy of identified LNK files (hash them with `Get-FileHash -Algorithm SHA256`) and Node.js bundle directories for malware analysis before deletion — these are the primary TonRAT delivery artifacts; (3) registry export of all Run/RunOnce keys under both HKCU and HKLM before deletion: `reg export HKLM\Software\Microsoft\Windows\CurrentVersion\Run hklm_run.reg`; (4) export of Windows Security Event Log Event ID 4624 (Successful Logon) and 4648 (Logon with explicit credentials) for compromised accounts covering the entire 90-day exposure window to establish the credential abuse timeline.

Step 4: Recovery — Validate remediation by re-running EDR sweeps for node.exe spawning from unauthorized paths 24 and 72 hours post-cleanup. Confirm registry persistence keys remain absent. Monitor outbound proxy logs for renewed TON API connections as a reinfection indicator. Re-image workstations where implant residence period cannot be bounded. Validate that email authentication controls (SPF, DKIM, DMARC) are enforced in reject mode — not monitor-only — for inbound mail (AC-17 — Remote Access; AU-12 — Audit Record Generation).

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST AU-12 (Audit Record Generation), NIST AC-17 (Remote Access)

Compensating: Without EDR for re-sweep, schedule a recurring Sysmon + PowerShell validation task via Windows Task Scheduler at 24h and 72h intervals: `Get-WinEvent -LogName 'Microsoft-Windows-Sysmon/Operational' -MaxEvents 5000 | Where-Object {$_.Message -match 'node.exe'}`. For DMARC enforcement validation without a paid email security platform, use the free MXToolbox DMARC lookup (mxtoolbox.com/dmarc.aspx) against your sending domains and verify the policy tag reads `p=reject`, not `p=none`. For TON API reinfection monitoring without a SIEM, configure a Windows Firewall audit rule and parse Security Event Log Event ID 5157 (blocking outbound connection) daily via: `Get-WinEvent -LogName Security | Where-Object {$_.Id -eq 5157 -and $_.Message -match 'toncenter'}`.

Evidence: Before re-imaging any workstation, ensure the following are preserved for post-incident review: (1) forensic disk image (using FTK Imager Lite or `dd`) of the affected endpoint — this is the only means to establish implant residence period, lateral movement scope, and data staged for exfiltration from the front-desk workstation; (2) full export of Windows Security, System, and Application event logs in EVTX format covering the entire April 2026 to present window; (3) network flow records (NetFlow/IPFIX or proxy logs) for the workstation IP showing the complete history of TON API connections — each C2 beacon interval may correspond to tasking events relevant to the scope determination; (4) a final Autoruns snapshot post-eradication to confirm persistence key absence as a remediation validation artifact.

Step 5: Post-Incident — Conduct phishing simulation exercises targeting front-desk and reservations staff with Calendly-style lure formats. Implement role-based email filtering rules that flag or quarantine booking-platform links sent to front-desk accounts from external senders. Evaluate whether DNS-layer controls are supplemented with TLS inspection capable of detecting blockchain API queries (CIS 7.1 — Establish and Maintain a Vulnerability Management Process). Review and enforce least privilege on front-desk workstations to limit blast radius of any future compromise (AC-6 — Least Privilege; CIS 5.4 — Restrict Administrator Privileges to Dedicated Administrator Accounts).

NIST Phase: Post Incident

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

Controls: CIS 7.1 (Establish and Maintain a Vulnerability Management Process), NIST AC-6 (Least Privilege), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts)

Compensating: Run phishing simulations using GoPhish (free, open-source) configured with a Calendly-mimic landing page and a Google redirect URL as the embedded link — this exactly replicates the TonRAT delivery chain your staff faced. For role-based email filtering without an enterprise SEG, use Exchange Transport Rules (available in Exchange Online and on-prem): `New-TransportRule -Name 'Block External Calendly to FrontDesk' -RecipientAddressContainsWords 'frontdesk' -SenderAddressLocation Header -FromScope NotInOrganization -HeaderMatchesPatterns 'calendly.com' -SetAuditSeverity High -DeleteMessage $true`. For DNS-layer TLS inspection without a commercial proxy, deploy Pi-hole with a custom blocklist containing TON API resolver domains (`toncenter.com`, `tonapi.io`) on the front-desk network segment.

Evidence: No volatile evidence capture required at this phase — this is a process improvement step executed after host state has been preserved and remediated. Document for the lessons-learned record: (1) the full email header chain of the confirmed phishing lure including Calendly booking URL, Google redirect hop, and final payload delivery URL — this becomes the detection signature baseline for future simulations; (2) a record of which front-desk accounts received and interacted with the lure, mapped to their Active Directory privilege level, to quantify blast radius and justify least-privilege enforcement changes; (3) the timeline delta between Microsoft's June 2026 campaign confirmation and your organization's initial detection — this gap metric drives the detection engineering backlog for blockchain C2 and authentication-laundering lure patterns.

Detection Guidance

Primary detection pivots: (1) Process telemetry, hunt for node.exe or node processes spawned from %TEMP%, %APPDATA%, or user-writable directories; flag any node.exe child process spawned by explorer.exe, cmd.exe, or wscript.exe following LNK execution. (2) Registry, monitor HKCU\Software\Microsoft\Windows\CurrentVersion\Run and HKLM equivalents for new Node.js-related entries (T1547.001). (3) Network, query proxy and firewall logs for outbound HTTPS to toncenter.com or other TON API resolver endpoints; blockchain-based C2 will appear as periodic HTTPS GET/POST requests to these resolvers rather than traditional C2 domains. (4) Email gateway, search for inbound messages containing calendly.com URLs combined with Google redirect wrappers (google.com/url?q=) delivered to hospitality role accounts; flag LNK attachments or ZIP archives containing LNK files. (5) Turnstile CAPTCHA challenges in email links may slow automated analysis but are not definitive evasion indicators; prioritize process telemetry and registry monitoring. Behavioral indicators consistent with MITRE T1568 (C2 via blockchain resolver), T1059.007 (Node.js execution), T1566.001/T1566.002 (spearphishing), and T1547.001 (registry persistence). Reference NIST SP 800-53 AU-6 (Audit Record Review, Analysis, and Reporting) and CIS 8.2 (Collect Audit Logs) for logging baseline coverage requirements.

Indicators of Compromise

Type	Value	Context	Confidence
DOMAIN	`toncenter.com`	TON blockchain API endpoint used by TonRAT for C2 resolution in place of traditional DNS-based infrastructure	MEDIUM
URL	`calendly.com/*`	Calendly booking links abused in phishing lures to pass SPF/DKIM/DMARC authentication checks targeting hospitality staff	MEDIUM

Type	Value	Context	Confidence
URL	`google.com/url?q=*`	Google open redirect URLs used as a wrapper to bypass email gateway URL reputation filtering	MEDIUM

Framework Mappings

MITRE-ATTACK

- **T1568** — Dynamic Resolution
- **T1218.011** — Rundll32
- **T1105** — Ingress Tool Transfer
- **T1071.001** — Web Protocols
- **T1564.003** — Hidden Window
- **T1036.005** — Match Legitimate Resource Name or Location
- **T1566.001** — Spearphishing Attachment
- **T1027** — Obfuscated Files or Information
- **T1566.002** — Spearphishing Link
- **T1033** — System Owner/User Discovery
- **T1059.007** — JavaScript
- **T1547** — Boot or Logon Autostart Execution
- **T1059.001** — PowerShell
- **T1016** — System Network Configuration Discovery
- **T1204.002** — Malicious File
- **T1027.003** — Steganography
- **T1082** — System Information Discovery
- **T1547.001** — Registry Run Keys / Startup Folder

NIST-800-53R5

- **CA-7** — Continuous Monitoring
- **SC-7** — Boundary Protection
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **AT-2** — Literacy Training and Awareness
- **SI-8** — Spam Protection
- **CM-7** — Least Functionality
- **SI-7** — Software, Firmware, and Information Integrity
- **SR-2** — Supply Chain Risk Management Plan

OWASP-TOP10-2021

- **A08:2021** — Software and Data Integrity Failures

CIS-V8

- **2.5** — Allowlist Authorized Software
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks
- **15.1** — Establish and Maintain an Inventory of Service Providers

HIPAA-SECURITY

- **164.308(a)(5)(i)** — Security Awareness and Training

ISO-27001-2022

- **A.5.34** — Privacy and protection of personal information
- **A.5.21** — Managing information security in the ICT supply chain
- **A.5.23** — Information security for use of cloud services

NIST-CSF-2

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

SOC2-TSC

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1568	Dynamic Resolution	Command-And-Control
T1218.011	Rundll32	Defense-Evasion
T1105	Ingress Tool Transfer	Command-And-Control
T1071.001	Web Protocols	Command-And-Control
T1564.003	Hidden Window	Defense-Evasion
T1036.005	Match Legitimate Resource Name or Location	Defense-Evasion
T1566.001	Spearphishing Attachment	Initial-Access
T1027	Obfuscated Files or Information	Defense-Evasion
T1566.002	Spearphishing Link	Initial-Access
T1033	System Owner/User Discovery	Discovery
T1059.007	JavaScript	Execution
T1547	Boot or Logon Autostart Execution	Persistence
T1059.001	PowerShell	Execution

Technique ID	Technique Name	Tactic
T1016	System Network Configuration Discovery	Discovery
T1204.002	Malicious File	Execution
T1027.003	Steganography	Defense-Evasion
T1082	System Information Discovery	Discovery
T1547.001	Registry Run Keys / Startup Folder	Persistence

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/06/microsoft-warns-of-photo-zip-phis...	T3
A Browser AI API? - End of Bug Bounties? - YouTube	https://www.youtube.com/watch?v=EWJbJgHFLcg	T3
Node.js 20.x < 20.20.0 / 22.x < 22.22.0 / 24.x < 24.13.0 / 24....	https://www.tenable.com/plugins/nessus/282656	T3
CDN Safety, Microsoft's Behavior, CDK Ransomware Attack - YouTube	https://www.youtube.com/watch?v=dcWo90vsvI8	T3
[PDF] Threats against people, apps, and infrastructure - Cloudflare	https://cf-assets.www.cloudflare.com/slt3lc6tev37/5vNmKWmtnMk1vubRK...	T3

Generated 2026-06-26 13:45 UTC by TJS Security Command Center