

PATCHCORD: New malware cluster targets Afghan telecom and South Asian critical infrastructure

 acronis.com/en/tru/posts/patchcord-new-malware-cluster-targets-afghan-telecom-and-south-asian-critical-infrastructure

Introduction

Acronis [Threat Research Unit](#) (TRU) has identified an ongoing campaign delivering a previously undocumented custom backdoor against Afghan telecom providers and South Asian critical infrastructure organizations. The backdoor, tracked as PATCHCORD, is a compiled C/C++ implant delivered through sector-specific lures, including fake VPN installers impersonating Afghan Telecom (AFTEL) and telecom management tools. Infrastructure pivoting uncovered SHEETCORD, a Go-based implant that builds on PATCHCORD's capabilities while abusing Google Sheets for C2 communication. The malware was actively distributed through a domain impersonating India's National Informatics Centre (NIC).

The campaign's infrastructure centers on a single C2 server with multiple associated domains, including domains impersonating Afghan telecom operators and a hijacked legitimate healthcare domain. An exposed staging server revealed the operator's broader toolkit, including SuperShell C2 framework, multiple RAT frameworks, credential harvesting tools, and exploit tooling for [CVE-2024-6387](#) (regreSSHion). This research details TRU's analysis of the PATCHCORD, SHEETCORD and HACKERAI C2 Agent malware families, their C2 mechanisms and the supporting infrastructure.

TRU assesses with moderate confidence that the campaign overlaps with the APT36 ([Transparent Tribe](#)) cluster based on targeting patterns, malware similarities, shared infrastructure, and operational tradecraft.

The campaign reflects an evolution of Transparent Tribe's recent operations. While the group has historically focused on government, military and diplomatic organizations in India and the broader South Asian region, our investigation identified a stronger operational focus on Afghan telecom providers alongside government, defense and energy organizations. Combined with three previously undocumented malware families and the use of Google Sheets and GitHub Gists for C2, the campaign demonstrates continued evolution in both the group's targeting priorities and operational tradecraft.

Telecom providers remain particularly attractive targets for espionage actors because they provide access to communications infrastructure, subscriber information and government

communications, enabling intelligence collection that extends well beyond a single organization.

Technical details

Initial analysis and delivery mechanism

The investigation began with the discovery of a suspicious ZIP archive, Telecom_TMS, on VirusTotal in June 2026. The archive contained a malicious Inno Setup installer, TMS_AfghanTelecom.exe, which delivered the PATCHCORD implant. The installer is a 32-bit Inno Setup (v6.7.0) executable compiled with Embarcadero Delphi 12.0 and protected with .NET Reactor.



Afghan Telecom-themed installer metadata

The Inno Setup installer contains version metadata designed to impersonate Afghan Telecom, with the CompanyName, FileDescription, and ProductName fields set to "Afghan Telecom," "TMS Afghan Telecom Setup," and "TMS Afghan Telecom," respectively (version 1.0.24).



Embedded TMS_Launcher overlay



Afghan Telecom TMS request portal

The installer contains an overlay at offset 0x00136200 holding the Inno Setup virtual file system, including a single embedded executable, TMS_Launcher.exe (147 KB), alongside the standard installer metadata and uninstaller module.

The installer further reinforces the Afghan Telecom impersonation through its application metadata. The AppPublisher field is set to "Afghan Telecom", while AppPublisherUrl points to `hxxps[:]//tms[.]afghantelecom[.]af`, the legitimate Afghan Telecom portal used for tracking service requests. This suggests the threat actor was familiar with the organization's infrastructure and deliberately crafted the installer to appear legitimate to Afghan Telecom employees. During installation, TMS_Launcher.exe is extracted and executed as the primary payload, which we track as PATCHCORD.

PATCHCORD implant

The extracted TMS_Launcher.exe is a 64-bit Windows executable compiled with the Microsoft Visual C/C++ (MSVC) toolchain and serves as the primary PATCHCORD implant.



PATCHCORD main execution flow

Following initialization, PATCHCORD hides its console window, establishes persistence through browser shortcut hijacking, fingerprints the host, registers with its C2 server and enters its main tasking loop.



Microsoft Edge path argument and browser-launch thread

PATCHCORD operates in two execution modes. When launched through a hijacked browser shortcut, it transparently starts the legitimate browser before continuing execution in the background, preserving the expected user experience while maintaining persistence. Otherwise, it proceeds directly to its primary functionality.



BeaconBrowserHijack Registry Run Key

The implant then checks for a registry value named BeaconBrowserHijack under HKCU\Software\Microsoft\Windows\CurrentVersion\Run. If the value already exists, it skips the shortcut hijacking phase, as the system has already been compromised. If not found, it writes its own executable path to this key, establishing persistence across reboots and triggering the browser shortcut hijacking routine that gets executed next.



Privilege check and browser-hijack dispatch

PATCHCORD establishes persistence by hijacking browser shortcuts. Before modifying shortcuts, the implant checks whether it is running with elevated privileges and then attempts to hijack shortcuts for Microsoft Edge, Google Chrome and Mozilla Firefox.



Browser identifier and executable-name mapping

The implant searches for shortcuts associated with Microsoft Edge, Google Chrome and Mozilla Firefox before attempting to hijack them.

The FindAndHijackBrowserShortcut function iterates through five shortcut locations on the victim system, including the taskbar, Quick Launch, Start Menu and both public and user desktop folders. For each location, it enumerates all .lnk files using the search pattern %s*.lnk and checks if the shortcut filename contains the target browser's executable name or displays name.

For each matching shortcut, the implant resolves the shortcut's actual target path using COM interfaces (IShellLinkW and IPersistFile) to verify it points to a genuine browser executable, preventing rehijacking of already compromised shortcuts. Once confirmed, the implant creates a backup of the original shortcut by appending .backup to its filename and then rewrites the shortcut through the HijackShortcutToPayload function, which sets the shortcut's target path to the implant's own executable, passes the original browser path as a quoted argument, changes the working directory to C:\Windows\System32, and preserves the original browser icon via SetIconLocation so the shortcut appears unchanged to the user.

As a result, every time the user clicks a browser shortcut, PATCHCORD executes first, silently launches the real browser via ShellExecuteW so the user notices nothing unusual, and proceeds with its C2 operations in the background. After establishing persistence, the implant continues with host fingerprinting and C2 registration.



Beacon initialization

The implant then enumerates the host operating system by querying the registry key HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion for the ProductName value, which returns the Windows edition string such as "Windows 10 Pro" or "Windows 11 Enterprise" on the victim's machine.



User-Agent String named Beacon/1.0.0

PATCHCORD initializes its C2 beacon by calling InternetOpenW with the user-agent string Beacon/1.0.0 and configures connection, send and receive timeouts to 30 seconds. The

resulting session handle is stored in the global variable `hInternet` and reused for subsequent C2 communications. After completing initialization, the implant enters its main C2 loop, periodically polling the server for tasking.



PATCHCORD JSON Schema Payload

The implant fingerprints the victim system by collecting the hostname, username, operating system version, process identifier, executable path and process name before constructing a JSON registration payload for the C2 server.



Client ID extraction and `/api.jsp` task URL



Hardcoded `appstoore[.]solutions` C2

The registration payload is sent as an HTTP POST request to the root path of the C2 server. In response, the server returns a `clientId=` parameter containing a 36-character UUID that uniquely identifies the session. The implant extracts this identifier and appends it to the `/api.jsp` endpoint, which it subsequently polls in a loop to receive tasking from the operator.



`/api.jsp` GET polling and response handling



Server response containing "It's Work!" [sic]



HTML comment wrapper

Once registered, the implant enters a polling loop, sending GET requests to the constructed `/api.jsp` tasking URL at regular intervals. If the server responds with the string "It's Work!", the implant treats it as a termination signal and exits the loop. Otherwise, the response is passed through a function we identified as `StripHtmlCommentWrapper`, which checks if the response body is wrapped in HTML comment markers (`<!--` and `-->`). If present, the function strips these markers and extracts the inner content before passing it to the command dispatcher.



PATCHCORD command handling

PATCHCORD supports five primary tasking operations, summarized in the table below.

The first feature of process enumeration is performed using the basic process enumeration technique, using `CreateToolHelp32Snapshot`, getting all the PID, names and path of the processes and converting them to a JSON format, after which the entire enumerated data gets encoded using the Base64 encoding schema, which then gets delivered to the C2 server.



In-Memory shellcode execution handler

Next, looking into second and one of the significant capabilities in PATCHCORD's command set is the in-memory shellcode execution handler. When triggered by the C2 server, the implant receives an encoded payload as part of the tasking response, decodes it using the same custom Base64 alphabet and decrypts it using a XOR-based routine with a key derived from the session context. The decrypted payload is then allocated into executable memory via `VirtualAlloc`, copied into the allocated region, and its memory protection is changed to `PAGE_EXECUTE_READ` via `VirtualProtect` before being executed as a new thread via `CreateThread`. The implant waits for the thread to complete before cleaning up the allocated memory. Notably, the shellcode itself is never written to disk, making this a fully in-memory execution chain that leaves minimal forensic artifacts on the victim's machine.

Infrastructure analysis also identified artifacts associated with the Metasploit and SuperShell C2 frameworks on the same server. This suggests the operator may leverage these frameworks to generate shellcode payloads delivered through PATCHCORD's in-memory execution capability.



Execution of Arbitrary Commands

Next, the third capability of this implant allows the operator to execute arbitrary commands on the victim's system. The function constructs a command string using `cmd.exe /c` followed by the received command and spawns it as a hidden process using `CreateProcessA` API with the `CREATE_NO_WINDOW` flag. The process output is then captured through an anonymous pipe, with `stdout` and `stderr` both redirected to the pipe's write handle.

The implant waits up to 60 seconds for the process to complete, and if the command exceeds this timeout, it forcefully terminates the process and appends a timeout notification to the output. The captured output is then encoded using the custom Base64 alphabet and sent back to the C2 server as a POST request to the tasking URL.



Remote Control Handler

Next, the fourth capability of this implant is to provide the operator with interactive, remote control over the shortcut hijacking mechanism. The function accepts a command string from the C2 server and supports several operations: "all" to hijack shortcuts for all three browsers, "edge" or "chrome" to target a specific browser, "restore" to revert all hijacked shortcuts back to their original state and "status" to retrieve a detailed JSON report of currently hijacked shortcuts including the browser name, original path, modified path and hijack status.

The operator can also pass a comma-separated list of browser names to selectively target multiple browsers in a single command. Each operation returns a JSON response indicating success or failure of the dedicated command, giving the operator a full-fledged leverage and full post-compromise flexibility to selectively deploy, remove or audit the shortcut hijacking persistence on a per-browser basis, depending on the need.

Infrastructure analysis

Analysis of the campaign infrastructure revealed that PATCHCORD communicates with the hardcoded C2 domain appstoore[.]solutions over TCP port 8080.



FOFA Result of appstoore[.]solutions

At the time of analysis, the campaign infrastructure remained active, with the C2 server continuing to respond to requests. Infrastructure pivoting identified the server at 46.30.188.13 (AS199959, Gwy IT Pty Ltd, Netherlands).

In addition, we found other campaigns operating on the same infrastructure, targeting additional sectors across South Asia, with different lure themes and delivery mechanisms, which we will discuss in the following section.

Additional campaigns

During the investigation, infrastructure pivoting uncovered additional campaigns and infrastructure linked to the same threat actor, revealing a broader operational footprint beyond the initial PATCHCORD deployment.



Additional campaigns identified through C2 pivoting

Pivoting on the C2 IP 46.30.188.13 through [Validin](#), we identified multiple domain resolutions tied to this server, all still actively resolving as of July 15, 2026. The domains reveal a clear pattern of impersonation targeting Afghan Telecom and Indian government and entities, with the threat actor steadily expanding its infrastructure over a six-month period.

The earliest domain which we saw in the cluster, **appstoore[.]solutions**, has been active since January 27, 2026, and serves as the primary C2 domain hardcoded in the PATCHCORD implant. By May 2026, the operator had registered **afghantelecom[.]site** which was first seen May 20, directly impersonating Afghan Telecom, along with **caprispine[.]health** which was first seen May 13, a fake domain impersonating [Capri Spine](#), a legitimate physiotherapy clinic chain based in Delhi, India, which we believe had been used as a lure or phishing front targeting Indian victims.

Through June 2026, the cluster expanded further with **afghanistanupdates[.]site** which was first seen June 22, 2026, designed to impersonate an Afghan government updates portal, along with **servicesindia[.]services** (first seen June 22) and **zala-aer[.]info** which was first seen June 16, 2026.

Live Campaign targeting Indian Ministry of Defense



nic-support[.]site payload delivery



ZIP archive contents

While the current research was undergoing, the most recently added domain, **nic-support[.]site** which have been seen by Acronis TRU Team first seen on **July 13, 2026**, appears to impersonate India's National Informatics Centre (NIC), the agency responsible for managing IT infrastructure across the Indian government, suggesting an expansion of targeting into Indian government IT networks. had been actively serving a file known as **MDEB_Update_Setup[.]exe**.



MDEB_Update_Setup.exe File Properties



MDEB_Update_Setup.exe file metadata

Analysis of the binary identified another Inno Setup installer resembling the PATCHCORD delivery mechanism. The installer impersonates a Ministry of Defense Employee Breach

Update and, when extracted, deploys a second implant, which we've tracked as SHEETCORD.

SHEETCORD implant

The Go-based malware combines functionality previously observed in the [SHEETCREEP](#) RAT with several capabilities introduced in PATCHCORD, suggesting an evolution of the operator's tooling.



SHEETCORD executing shell command function

The implant implements a remote command execution capability `main.executeShellCommand` similar to PATCHCORD. However, instead of invoking `cmd.exe /c`, it executes commands through `powershell -Command` with script block wrapping, suggesting an evolution of the operator's tooling.



SHEETCORD hostname collection routine

SHEETCORD also includes a `main.getHostInfo` function that serves nearly the same purpose as PATCHCORD's registration payload builder, collecting the victim's hostname for initial check in. Notably, the Go variant collects significantly less system information compared to PATCHCORD's detailed fingerprinting, which included hostname, username, process details and OS version.



Startup folder persistence

SHEETCORD also introduces an additional persistence mechanism not present in PATCHCORD. The `main.installStartupEntry` function drops a VBScript file named `SystemHelper.vbs` into the Windows Startup folder at `%APPDATA%\Microsoft\Windows\Start Menu\Programs\Startup\`, which silently launches the implant on every user logon using `WScript.Shell` with a hidden window. It then adds a registry Run key under `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` with the value name `SystemHelper`, pointing to the implant's executable path. Compared to PATCHCORD, which used the value name `BeaconBrowserHijack` and the `RegSetValueExW` API, the Go variant shells out to `reg.exe` to achieve the same result.



Taskbar shortcut scan and six-browser target list



temp_update.vbs shortcut rewrite generation

The implant also reimplements PATCHCORD's browser shortcut hijacking mechanism. Similar to its predecessor, it searches the user's Desktop and %APPDATA%\Microsoft\Internet Explorer\Quick Launch\User Pinned\TaskBar for browser shortcuts. Unlike PATCHCORD, which targets three browsers, the Go-based implant expands support to six: Chrome, Firefox, Edge, Brave, Opera, and Vivaldi.

Rather than modifying shortcuts through COM interfaces, the implant generates a temporary VBScript (temp_update.vbs) that rewrites each shortcut to launch the malware while preserving the original browser path as an argument. The script is then executed via wscript to apply the changes.



Google Sheets service-account initialization



Google Sheets A1-range task retrieval



GCP Service-Account Credentials

Analysis of the Google Sheets-based C2 mechanism revealed significant code and functionality overlaps with the previously documented SHEETCREEP RAT. The SHEETCORD implant uses the Google Sheets API v4 with a hardcoded GCP service account for authentication and creates per-victim tabs in the operator's spreadsheet for bidirectional communication, all of which are consistent with the documented SHEETCREEP implant.

Campaign targeting Indian energy sector

Infrastructure pivoting identified an earlier campaign, observed in **March 2026**, targeting India's energy sector with a different PATCHCORD variant. Aside from the updated implant, the delivery mechanism remained largely unchanged.



Lure used in the campaign



Installer metadata

The delivery mechanism closely mirrors the previously observed campaign, with the primary difference being the lure, which impersonates the NHPC Fuel Conservation Client. The installer also uses a legitimate-looking icon to reinforce its authenticity.



Anti-analysis techniques

The only notable difference we found with this variant of PATCHCORD is the inclusion of anti-analysis techniques which were not present in the earlier sample. The implant checks for VirtualBox and VMware device handles (VBoxGuest, VBoxTrayIPC, vmci, HGFS), verifies the system has more than one processor and at least 2GB of RAM and checks for attached debuggers through both `IsDebuggerPresent` and the PEB `BeingDebugged` flag. It also scans active TCP connections for ports commonly associated with analysis tools such as Burp Suite and Fiddler and enumerates running processes against a hardcoded list of security tools including `wireshark.exe`, `x64dbg.exe`, `ida.exe`, `processhacker.exe`, `sysmon.exe`, and `burpsuite.exe`. It further monitors cursor movement and user input to detect automated sandbox environments. If any check is triggered, the implant enters a randomized sleep loop of 30 to 90 seconds, designed to exhaust sandbox execution timeouts without raising process-termination alerts.

Infrastructural rotations

Throughout the investigation, we observed that the threat actor maintained a single C2 server at 46.30.188.13 (AS 199959, Gwy IT Pty Ltd) while steadily rotating the domains pointing to it over a seven-month period. Validin passive DNS history captured the full domain lifecycle, which we have summarized below.

The earliest activity traces back to December 2025 with `appstoore[.]duckdns[.]org`, a dynamic DNS domain, followed by the registration of `appstoore[.]solutions` in late January 2026, which was subsequently hardcoded as the primary C2 domain in the PATCHCORD implant. By May 2026, the operator began expanding the infrastructure with domains impersonating specific organizations: `caprispine[.]health` (first seen May 13), `afghantelecom[.]site` (first seen May 20) and `nicservice[.]org` (first seen May 21). Notably, `nicservice[.]org` was the only domain in the cluster to go offline, with its last observed resolution on June 16, 2026, suggesting the operator retired it in favor of newer infrastructure.

Through June 2026, additional domains were introduced including `zala-aer[.]info` (first seen June 16), `afghanistanupdates[.]site` (first seen June 22) and `servicesindia[.]services` (first seen June 22). The most recent addition, `nic-support[.]site`, appeared on July 13, 2026, and is currently being used to serve the SHEETCORD implant. As of July 16, 2026, while the

research had been drafted, all domains except nicservice[.]org remain actively resolving to the same IP, indicating the operator has not yet begun decommissioning older infrastructure.

The server's TLS certificates, issued through Let's Encrypt, were rotated three times during this period, each time bundling different domain sets as Subject Alternative Names. The data we found from Validin scan data also confirmed that all active domains consistently redirected visitors to a /download path, indicating the same nginx instance was being used to serve payloads across all domain fronts simultaneously.

Exposed staging infrastructure

An exposed staging server revealed the threat actor's broader toolkit, including proof-of-concept exploits, publicly available C2 frameworks, AI-assisted malware projects and campaign-related files.



Exposed staging server directory listing



Campaign lures, exploits and tooling related to APT36 (1)



Campaign lures, exploits and tooling related to APT36 (2)

The listing exposed campaign-specific files including TMS_AfghanTelecom.exe, Bonus_Salaam_telecom.zip, Ministry of Communication & Information Technology.msi, and an accompanying MCIT.pdf decoy document, directly tying this server to the PATCHCORD campaign targeting Afghan telecom operators. We also identified what appear to be staged spear-phishing archives such as Bonus_Salaam_telecom.zip, system_update.zip, chrome.zip, and inf.zip, none of which had been submitted to VirusTotal at the time of our analysis, suggesting these were prepared for upcoming or ongoing campaigns that had not yet been widely distributed.

In addition to the campaign files, the directory also revealed the operator's broader offensive toolkit. We identified antnium/, an open-source Go-based C2 framework for initial access publicly available on GitHub, which is notable given that both PATCHCORD's C2 backend and the SHEETCORD implant are also Go-based, suggesting the operator has a strong preference for Go-based tooling. The directory also contained GateSentinel-C2-Rat-Hvnc/, the GateSentinel C2 framework independently documented by researchers at [Bitdefender](#) as part of APT36's toolkit in their March 2026 report on Transparent Tribe's operations, along

with a Nim-Backdoor/ directory consistent with APT36's documented shift toward niche programming languages and AI-vibecoded malware.

Metasploit was present through .msf4/ and msfinstall, consistent with the Maltrail signature flagged on this IP and the open port 4444 observed during our initial scans. We also found HackBrowserData/, an open-source browser credential harvesting tool, exploit tooling for CVE-2024-6387 (regreSSHion) and CVE-2021-4034 (PwnKit), and credential brute-forcing artifacts including hydra.restore and rockyou_2025_05.txt.

A directory named chm_da_content/ suggests the use of CHM files as a delivery mechanism, a technique commonly observed in South Asian APT campaigns. Multiple Python-based C2 listeners (c2_server.py, https_c2.py, https_listener.py, http_listen.py) indicate the operator maintains custom Python infrastructure alongside the Go and Metasploit frameworks. Perhaps most concerning, files such as Routers_backup.zip, Sessions_Backup.zip, and kCallingHistoryCallStateV1RecordType_record.json, the last of which follows an iOS call history database schema, suggest the operator may have exfiltrated data from compromised network devices and mobile targets in addition to Windows systems.



CVE-2024-6387 exploit

Also, the presence of exploit tooling for CVE-2024-6387 (regreSSHion) alongside credential brute-forcing artifacts such as hydra.restore and rockyou_2025_05.txt suggests the operator is actively scanning for and exploiting vulnerable OpenSSH instances, likely targeting internet-facing telecom and government infrastructure in the region to gain initial access at the network level before deploying endpoint implants such as PATCHCORD.

SuperShell pivoting



SuperShell C2 servers found



SuperShell login panel

The staging server contained SuperShell v2.0.0, an open-source Chinese-language C2 framework commonly used by Chinese-speaking threat actors. The login panel, accessible on port 443 of the C2 server, displays a fully Chinese interface with fields for username (用户名) and password (密码). SuperShell is an open-source webshell management and C2 platform that provides operators with remote command execution, file management, and reverse shell capabilities through a web-based interface.



Matching port number 8080 across campaigns



Shodan results of SuperShell servers

FOFA pivoting identified a second server at 108.187.42.63 (Antbox Networks Limited, AS138995, Huizhou, Guangdong, China) exposing the same SuperShell C2 response on TCP port 8080. This infrastructure was independently documented by Kaspersky as a ValleyRAT C2 server associated with the SilverFox [ABCDoor](#) campaign. While this represents an interesting infrastructure overlap, we found no evidence of operational coordination or attribution linkage between the two groups.

AI-assisted malware: HACKERAI C2



Pivotal Intelligence leading to discovery of a AI-generated implant

Historical SSL certificate data associated with the campaign infrastructure identified additional domains previously hosted on the same IP address, including defence[.]cgda[.]site, a domain impersonating India's Controller General of Defence Accounts (CGDA), first observed on September 8, 2025. As the earliest domain in the infrastructure cluster, predating the PATCHCORD campaign by several months, it was used to distribute a previously undocumented malware framework that we track as **HACKERAI C2 Agent**.

Historical infrastructure also revealed vpn01[.]afghantelecom[.]site, first observed on April 15, 2026, impersonating an Afghan Telecom VPN portal and matching the AFTEL_VPN_Setup.exe lure observed in the PATCHCORD campaign.

The implant contains code comments and implementation patterns consistent with AI-assisted code generation. Together, these observations suggest that the malware was developed, at least in part, using LLM-assisted coding tools.

HACKERAI C2 Agent shares multiple capabilities with PATCHCORD and SHEETCORD, including system fingerprinting, remote command execution, and browser shortcut hijacking for persistence. **For C2 communications, however, it replaces PATCHCORD's custom HTTP server and SHEETCORD's Google Sheets with GitHub Gists, using dedicated upload and download functions for both tasking and data exfiltration.** This represents a third distinct C2 mechanism observed across the malware cluster.



Junk-operation based features inside the AI-generated implant

The implant also includes basic anti-analysis features such as environment validation checks against standard Windows variables and a performJunkOperations function that loads meaningless placeholder strings like "foo", "bar", and "baz" in a loop, calling randomSleep between each iteration to introduce unpredictable execution delays without relying on traditional sleep APIs. Combined with the recursive Fibonacci calculator observed earlier, these techniques are designed to frustrate sandbox analysis and automated behavioral detection, although they remain relatively unsophisticated compared to the full anti-analysis suite found in the PATCHCORD variant.



Junk-operation based features inside the AI-Generated implant – II

The implant exhibits several characteristics consistent with AI-assisted development. These include AI-style code comments, clear-text debugging messages such as [LOG] Agent Started, leftover test code, a redundant double XOR routine using the same 0xAB key, and a hardcoded GitHub Personal Access Token (PAT). While none of these characteristics are individually conclusive, together they suggest the developer relied, at least in part, on AI-assisted coding tools during implementation.

Attribution and confidence

Although the infrastructure contains a diverse set of tools, TRU assesses with moderate confidence that the activity overlaps with the broader APT36 (Transparent Tribe) cluster or a closely related Pakistan-linked threat actor based on targeting, malware similarities, infrastructure and operational tradecraft. The first and most direct indicator is the campaign's sustained focus on Afghan telecom and government infrastructure. The operator-maintained lures targeting Afghan Telecom (AFTEL), Salaam Telecom through Bonus_Salaam_telecom.zip, and Afghanistan's Ministry of Communications and Information Technology through the Ministry of Communication & Information Technology.msi installer and accompanying MCIT.pdf decoy document. Purpose-registered domains such as afghantelecom[.]site and afghanistanupdates[.]site further reinforce this targeting.

Afghanistan has been previously documented as a target of APT36 operations, and the presence of exploit tooling for CVE-2024-6387 (regreSSHion), a critical OpenSSH vulnerability on the same server suggests the operator may also be pursuing access to Afghan telecom network infrastructure at the service level, not just through endpoint compromise.

The second indicator is the presence of HackBrowserData/ on the operator's server. HackBrowserData is an open-source browser credential harvesting tool that has been previously observed in APT36 operations targeting [Indian Air Force officials](#), where a modified version of the tool was compiled into a Go-based stealer and delivered through defense-themed ISO lures impersonating the Sukhoi-30 MKI procurement. The presence of this same tool on the PATCHCORD operator's server, combined with the shared targeting of Indian critical infrastructure and government entities, suggests a common operational toolkit.

The third indicator is the presence of GateSentinel-C2-Rat-Hvnc/ in the operator's open directory. GateSentinel has been independently documented as part of APT36's evolving toolkit and has been observed alongside other frameworks such as Cobalt Strike and Havoc in recent Transparent Tribe campaigns. The co-location of GateSentinel with PATCHCORD campaign files on the same server provides a direct link to APT36's known operational preferences.

The fourth and perhaps most compelling indicator is the use of Google Sheets as a C2 channel in the SHEETCORD implant distributed through nic-support[.]site. This technique closely resembles the previously documented SHEETCREEP campaign, which has been attributed with medium confidence to APT36 or a related Pakistan-linked subgroup. Both implants embed hardcoded Google Cloud Platform (GCP) service account credentials for authentication, create per-victim spreadsheet tabs for bidirectional communication and expose cloud credentials directly within the binary.

Although the infrastructure also contained SuperShell and shared a similar C2 fingerprint with infrastructure previously attributed to SilverFox, these observations alone are insufficient to support an attribution. The observed targeting of Indian and Afghan organizations remains more consistent with APT36's documented operational focus.

Conclusion

The PATCHCORD campaign demonstrates an evolving espionage operation targeting telecom, government, defense, and critical infrastructure organizations across South Asia. The discovery of PATCHCORD, SHEETCORD, and HACKERAI C2 Agent **highlights the operator's continued evolution, from a custom C/C++ backdoor to Go-based implants that abuse legitimate cloud services**, including Google Sheets and GitHub Gists, for command-and-control.

The exposed staging server and related infrastructure provided valuable insight into the operator's tooling, campaign development, and operational practices, enabling the identification of additional infrastructure and previously undocumented malware.

At the time of publication, the campaign infrastructure remains active. Organizations across South Asia should remain vigilant for sector-specific phishing campaigns and monitor for the indicators of compromise provided in this report. TRU will continue tracking this activity and publish additional findings as the campaign evolves.

Detection by Acronis

This threat is detected and blocked by Acronis EDR / XDR:



Acronis EDR / XDR detection

Indicators of Compromise

SHA256 (Installers / Droppers):

TMS_AfghanTelecom.exe

cf7184c0dfe882dc6e3016f16e4ede32b75d7648f83d6f4f87eb6a703be7b8d6

AFTEL_VPN_Setup.exe

1774e15e8eb96eb89bc03cb4768fc0620e10c09c5f795297f36dcc2aa5d9dd94

NHPC_Fuel_Conservation_Setup.zip

ea0934472121848b80455581d289ce4480b1e5cc05678c1b90ecfc465b5ec350

nhpcfuelconservationservice.exe

5e17360d32e9b272bb7e1b97c8e4dca34622ec9ce08fd240fe2758cc3f67dc4a

MDEB_Update_Setup.exe

378484112b4e837d3850b5b0802fc509202c232bb124d6944a59fe66525ba668

MOFA_Employee_Benefits.exe

b56fab5a6834c51d85787e7c1177720dfba5a5823763f3fcf432196cd2a1bdf3

JitsiMeet-Setup.exe

2323b55ea743c813e48689318e8ed54ae838cf9e8a2adbfc2488ea8a36dd0126

N/A

2eddfebb3f7419af27493a6a3bb601372cf6c494da8df62640cce7f830b4a73b

SHA256 (PATCHCORD):

tms_launcher.exe

d46ee94d6a27ff9f02cff6fb57780acac2833ce48c95e63042a6274e24a040bb

node_service.exe

50fc220347f9e281037e831c3755dc70a8ba7f663025aea35b301226918b016b

agent.exe

0f4073d3c866bc3daf55b25f71250b96ec120db94a4f9cc8fe85b7c9f9d346b3

SHA256 (SHEETCORD):

system-agent.exe

959bbb09cd86ce3930406bf1cf32776ca477dfefe3fd63e90bf0017fccd90587

SHA256 (HACKERAI C2):

Agent.exe

74d347785dc47f8cda3876826cdd3fb3935ac55dc8e9e0c0f96d5ef4e00089a2

C2 domains:

appstoore[.]solutions

www[.]appstoore[.]solutions

afghantelecom[.]site

afghanistanupdates[.]site

www[.]afghanistanupdates[.]site

caprispine[.]health

www[.]caprispine[.]health

servicesindia[.]services

www[.]servicesindia[.]services

zala-aer[.]info

www[.]zala-aer[.]info

nicservice[.]org

www[.]nicservice[.]org

nic-support[.]site

appstoore[.]duckdns[.]org

defence[.]cdga[.]site

C2 IP:

46.30.188.13 (AS 199959, Gwy IT Pty Ltd, Netherlands)

Registry persistence:

HKCU\Software\Microsoft\Windows\CurrentVersion\Run\BeaconBrowserHijack
(PATCHCORD)

HKCU\Software\Microsoft\Windows\CurrentVersion\Run\SystemHelper (SHEETCORD)

Startup persistence (SHEETCORD):

%APPDATA%\Microsoft\Windows\Start Menu\Programs\Startup\SystemHelper.vbs

User-Agent:

Beacon/1.0.0

© 2003–2026 Acronis International GmbH.