

PDFSIDER Malware - Exploitation of DLL Side-Loading for AV and EDR Evasion

 resecurity.com/blog/article/pdfsider-malware-exploitation-of-dll-side-loading-for-av-and-edr-evasion

January 18, 2026

[Back](#)

Cyber Threat Intelligence

PDFSIDER malware, DLL side-loading, AES-256-GCM encrypted C2, In-memory backdoor, Anti-VM sandbox evasion

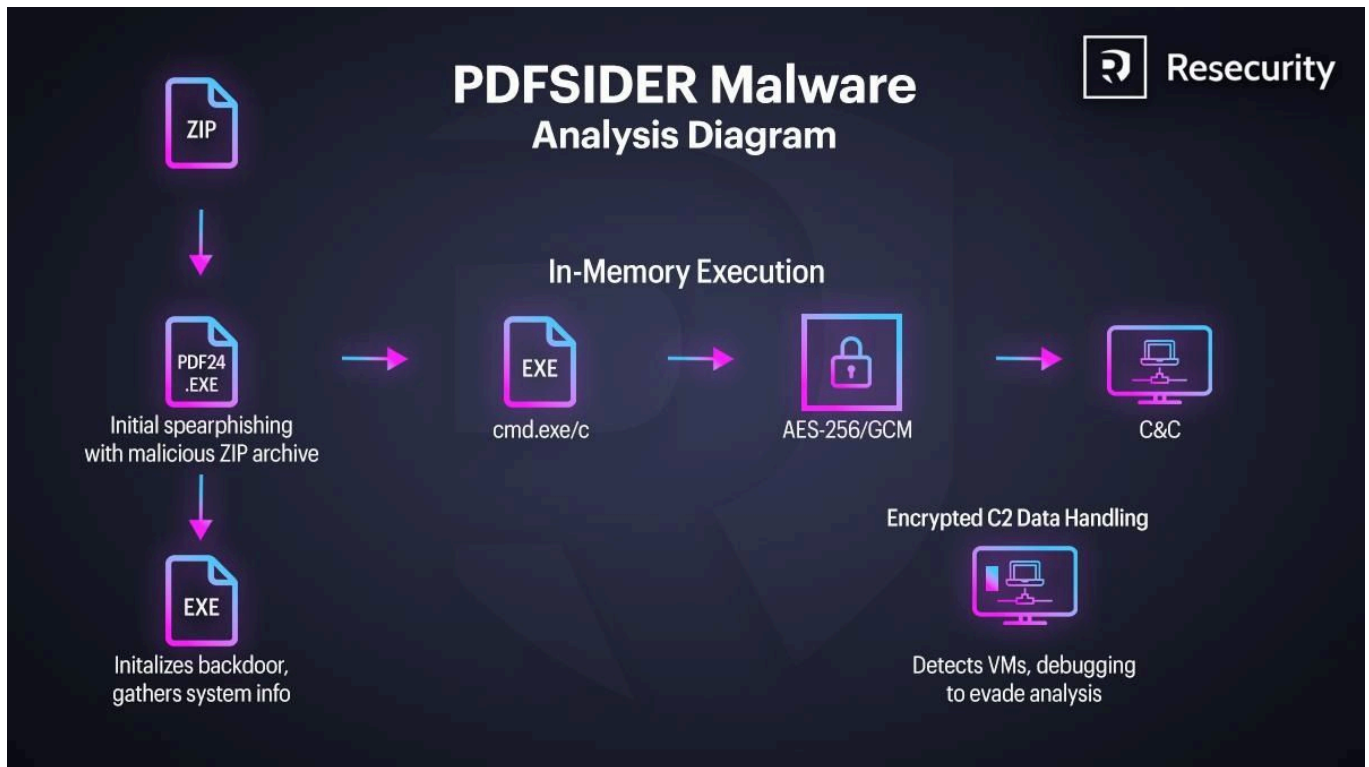


Overview

PDFSIDER is a newly identified malware variant distributed through **DLL side-loading**, designed to covertly deploy a **backdoor with encrypted command-and-control (C2)** capabilities. The malware uses a fake `cryptbase.dll` to bypass endpoint detection mechanisms.

The malware **is identified as an Advanced Persistent Threat (APT)**, having characteristics commonly associated with APT tradecraft, including stealthy execution, anti-VM checks, and encrypted communications. PDFSIDER blends traditional cyber-espionage behaviors with modern remote-command functionality, enabling operators to gather system intelligence and remotely execute shell commands covertly.

The analyzed DLL contains a fully embedded **Botan cryptographic library**, configured for **AES-256-GCM** authenticated encryption - a strong indicator that the threat actors employ secure and structured C2 protocols. Once loaded, the malware provides attackers with an interactive, hidden command shell and can exfiltrate command output through its encrypted channel.

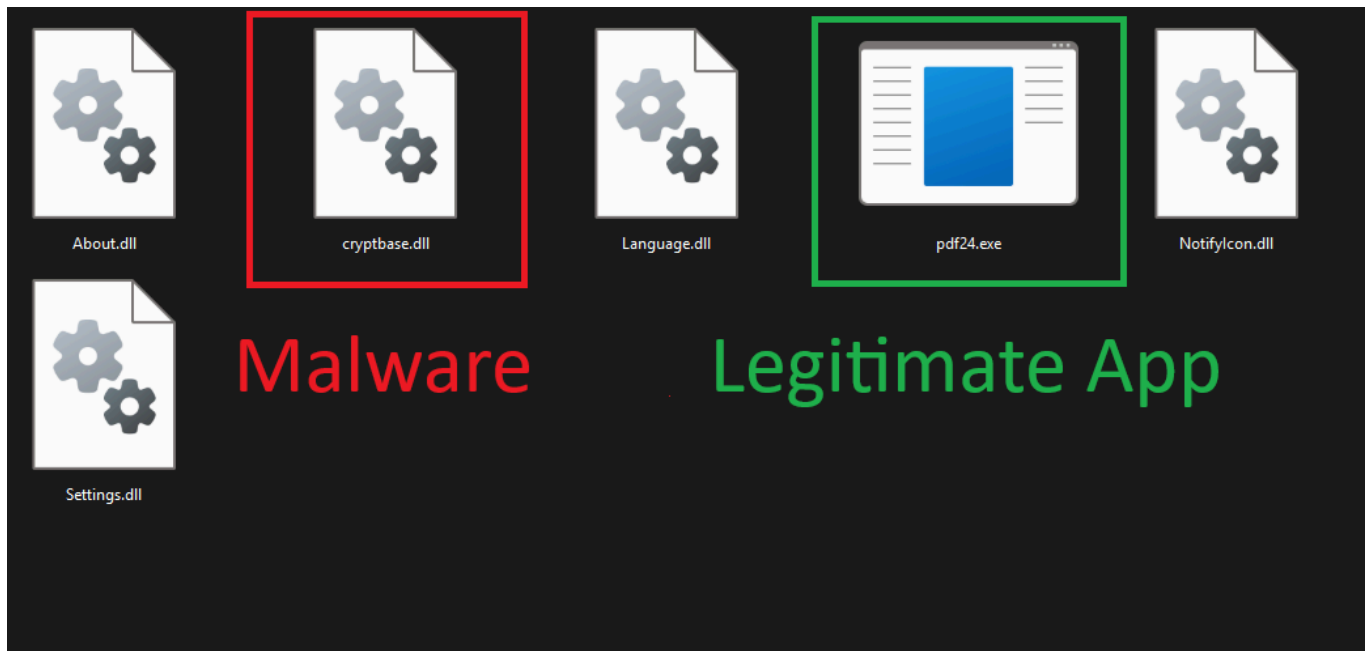


This report outlines the malware's technical components, execution workflow, evasion mechanisms, and key indicators to support detection and response efforts.

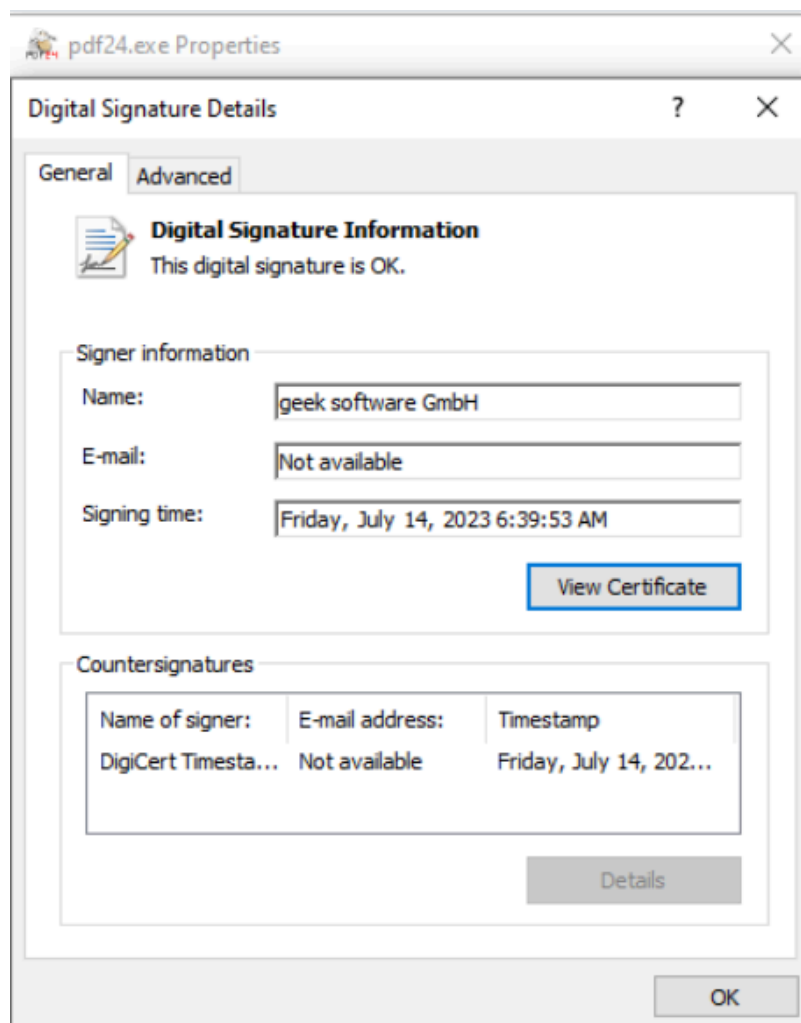
Technical Details

1. Initial Infection and Delivery

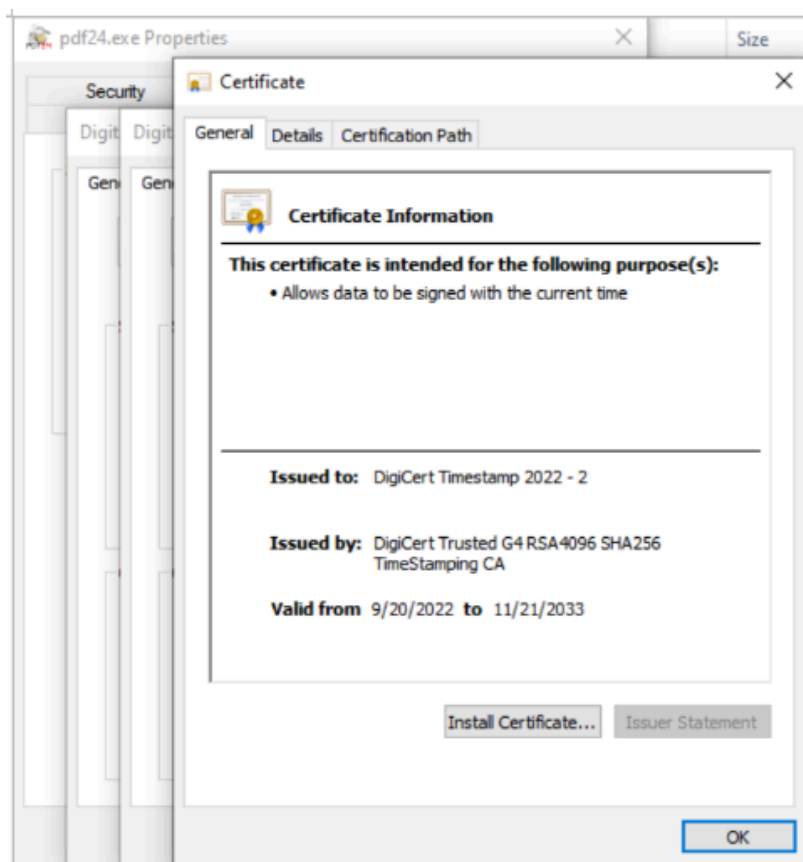
PDFSIDER launched this campaign using spear-phishing emails that directed victims to a ZIP archive attached to the message. The ZIP archive contains a legitimate EXE file, labeled 'PDF24 App.' **PDF24 Creator** is an application software by **Miron Geek Software GmbH** used for creating PDF files from any application and for converting files to the PDF format.



The infection chain begins when the victim opens the EXE file, which has no display when it runs on the machine; however, it will start to operate immediately on the machine when executed.



One of such certificates used for signing is still valid:



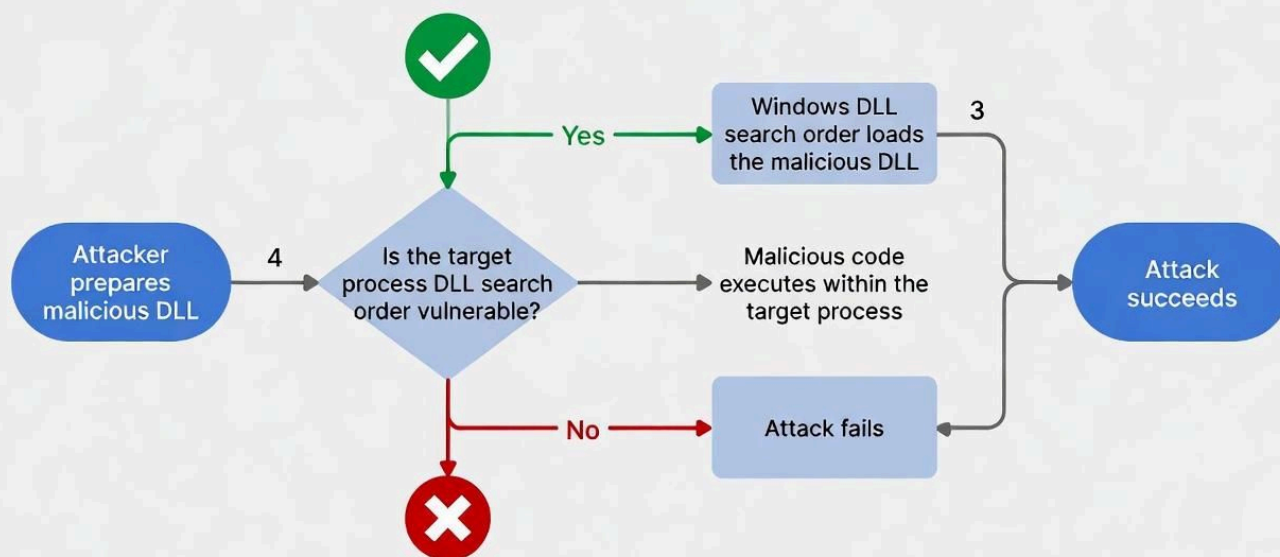
The EXE file has a legitimate signature; however, the PDF24 software has vulnerabilities that attackers were able to exploit to load this malware and bypass EDR systems effectively.

This is becoming more common nowadays due to the rise of AI coding, making it easier for cybercriminals to find vulnerable software and exploit it for their own purposes.

The malware appears to rely on **DLL Side-Loading**, a technique where a malicious DLL is placed next to a legitimate application that automatically loads it.

When the victim runs the legitimate program, The PDF24.exe loads the attacker's DLL instead of the real system `cryptbase.dll`, granting code execution.

DLL side-loading Attack



pdf24.exe	5068	RegSetInfoKey	HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\SideBySide
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Session Manager
pdf24.exe	5068	File SystemControl	C:\Users\Desktop\Desktop\pdf24 1\cryptbase.dll
pdf24.exe	5068	SetEaFile	C:\Users\Desktop\Desktop\pdf24 1\cryptbase.dll
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Srp\Gp
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Srp\Gp
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\WinSock2\Parameters
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Lsa
pdf24.exe	5068	CreateFile	C:\Users\Desktop\AppData\Local\Temp\pdf24_write_test_1660984_3934549074_0
pdf24.exe	5068	SetDispositionInformationEx	C:\Users\Desktop\AppData\Local\Temp\pdf24_write_test_1660984_3934549074_0
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\WinSock2\Parameters
pdf24.exe	5068	RegSetInfoKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Themes\Personalize
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Srp\Gp
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\Winsock\Parameters
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\Tcpip6\Parameters\Winsock
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Winsock
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\Winsock\Setup Migration\Providers
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Winsock
pdf24.exe	5068	RegSetInfoKey	HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Explorer\FolderDescriptions
pdf24.exe	5068	RegSetInfoKey	HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Explorer\FolderDescriptions
pdf24.exe	5068	RegSetInfoKey	HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Explorer\FolderDescriptions
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Srp\Gp
pdf24.exe	5068	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Hvsi
pdf24.exe	5068	RegSetValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC1C75

Once loaded, PDFSIDER initializes Winsock, sets up encrypted communications, gathers system information, and begins its backdoor loop.

2. In-Memory Execution

PDFSIDER operates primarily **in memory**, minimizing disk artifacts:

The malware provides full remote command execution upon launch, performing several actions. It creates anonymous pipes and constructs a command string using `cmd.exe /C <attacker-supplied command>`.

```

6ddc1b15 HANDLE hObject = nullptr
6ddc1b1f int32_t var_1a8 = 0
6ddc1b29 int32_t var_1a4 = 0
6ddc1b33 int32_t var_198
6ddc1b33 int32_t* var_1a0 = &var_198
6ddc1b39 __builtin_strncpy(dest: &var_198, src: "cmd.exe /C ", n: 0xc)
6ddc1b4d uint32_t _Size_8 = 0xb
6ddc1b5e int32_t var_1c8 = 0
6ddc1b68 int32_t var_1c4 = 0
6ddc1b72 int32_t var_1c0 = 0
6ddc1b72
6ddc1b7f if (ebx + 0xc < 0)
6ddc2a4e     sub_6df5e580("vector::reserve")
6ddc2a4e     noreturn
6ddc2a4e

```

This command is launched with the `CREATE_NO_WINDOW` flag, ensuring no visible console appears.

The malware then extracts all information from your system, creates a unique ID from your system, and transmits the output back to the attacker.

CRYPTBASE.DLL	<code>strlen ("53")</code>
CRYPTBASE.DLL	<code>memcpy (0x034cfd30, 0x00b911f8, 2)</code>
CRYPTBASE.DLL	<code>strcpy ("53", 0x034cfd34, 10)</code>
CRYPTBASE.DLL	<code>getenv ("COMPUTERNAME")</code>
CRYPTBASE.DLL	<code>strcpy (0x034cfb14, "DESKTOP-[REDACTED]")</code>
CRYPTBASE.DLL	<code>getenv ("USERNAME")</code>
CRYPTBASE.DLL	<code>strcpy (0x034cfb54, "mypc")</code>
CRYPTBASE.DLL	<code>GetCurrentProcessId ()</code>
CRYPTBASE.DLL	<code>calloc (500, 1)</code>
CRYPTBASE.DLL	<code>strlen ("DESKTOP-[REDACTED]")</code>
CRYPTBASE.DLL	<code>strlen ("Nov 24 2025")</code>
CRYPTBASE.DLL	<code>strlen ("21:50:41")</code>
CRYPTBASE.DLL	<code>strlen ("mypc")</code>

2.2 Encrypted C2 Data Handling

The malware utilizes the **Botan 3.0.0** cryptographic library for encryption, incorporating several key components such as **AES-256/GCM**, **GHASH**, and checks for **GCM tag failures**.

```

6df7df21 data_6df7df21:
6df7df21      69 6e 20 00 00 00 00
in ....

6df7df28 char const data_6df7df28[0xc1] = "/home/john/.info/python/prof/reverse-proxy-
udp/src/win32-build/_deps/botan_src-src/botan_all.cpp", 0
6df7df89 char const data_6df7df89[0xe] = "aes_encrypt_n", 0
6df7df97 char const data_6df7df97[0xc] = "Key was set", 0

6df7dfa3

6df7dfa4 char const data_6df7dfa4[0x36] = "EK.size() == 44 || EK.size() == 52 ||
EK.size() == 60", 0
6df7dfda char const data_6df7dfda[0xe] = "aes_decrypt_n", 0
6df7dfe8 char const data_6df7dfe8[0x36] = "DK.size() == 44 || DK.size() == 52 ||
DK.size() == 60", 0

```

20 69 73 20	6E 6F 74 20	76 61 6C 69	64 00 73 74	is not valid st
6F 69 00 41	45 53 2D 32	35 36 2F 47	43 4D 00 00	oi.AES-256/GCM.
30 30 30 31	30 32 30 33	30 34 30 35	30 36 30 37	0001020304050607
30 38 30 39	31 30 31 31	31 32 31 33	31 34 31 35	0809101112131415
31 36 31 37	31 38 31 39	32 30 32 31	32 32 32 33	1617181920212223
32 34 32 35	32 36 32 37	32 38 32 39	33 30 33 31	2425262728293031
33 32 33 33	33 34 33 35	33 36 33 37	33 38 33 39	3233343536373839
34 30 34 31	34 32 34 33	34 34 34 35	34 36 34 37	4041424344454647
34 38 34 39	35 30 35 31	35 32 35 33	35 34 35 35	4849505152535455
35 36 35 37	35 38 35 39	36 30 36 31	36 32 36 33	5657585960616263
36 34 36 35	36 36 36 37	36 38 36 39	37 30 37 31	6465666768697071
37 32 37 33	37 34 37 35	37 36 37 37	37 38 37 39	7273747576777879
38 30 38 31	38 32 38 33	38 34 38 35	38 36 38 37	8081828384858687
38 38 38 39	39 30 39 31	39 32 39 33	39 34 39 35	8889909192939495
39 36 39 37	39 38 39 39	00 00 00 00	00 00 00 00	96979899.....
5B 2A 2A 2A	20 4C 4F 47	20 45 52 52	4F 52 20 23	[*** LOG ERROR #

This strongly suggests that all inbound and outbound data handled by the malware is **decrypted in memory**, ensuring it is **never written to disk**. The data is also **authenticated using AEAD (Authenticated Encryption with Associated Data)** in **GCM** mode, providing a high level of security. This type of cryptographic implementation is typical of remote shell malware used in **targeted attacks**, where maintaining the integrity and confidentiality of communications is critical.

3. Anti-VM Technology

PDFSIDER features a multi-stage environment validation routine designed to detect sandboxes, virtual machines (VMs), and analysis labs. The malware uses the GlobalMemoryStatusEx function to assess the system's total RAM, with low-RAM systems (commonly associated with sandboxes or virtual environments) triggering an early exit.

```

6ddc18f7  int32_t lpBuffer 0x40
6ddc1902  GlobalMemoryStatusEx(&lpBuffer)
6ddc1919  bool c = unimplemented {sbb eax, dword [esp+0x1c]}
6ddc1926  return zx.d(not(c))

6ddc1590  int32_t sub_6ddc1590()

6ddc159a  if (sub_6ddc18f0() == 0 && sub_6ddc1930() == 0 && sub_6ddc1990() == 0)
6ddc15ae      int32_t result = sub_6ddc19c0()
6ddc15ae
6ddc15b5      if (result == 0)
6ddc15ba          return result
6ddc15ba
6ddc15c2  exit(_Except: 0xfffffffffe)
6ddc15c2  noreturn

```

Additionally, it includes checks for the presence of a debugger, further ensuring that the malware avoids execution in a controlled or monitored environment.

```

6ddf4fbd      if (edx_1 != 0)
6ddf4fc5          free(_Block: edx_1)
6ddf4fc5
6ddf4fcd      *(eax_2 + 0x34) = eax_5
6ddf4fd0      int32_t eax_6 = *(eax_2 + 0x3c)
6ddf4fd3      int32_t lpArguments = 0x1000
6ddf4fda      char* var_28_1 = arg2
6ddf4fdd      int32_t var_24_1 = eax_6
6ddf4fe0      int32_t var_20_1 = 0
6ddf4fe0
6ddf4ff8      if (IsDebuggerPresent() == 0 && data_6e0461dc == 0)
6ddf503e          return result
6ddf503e
6ddf5018      RaiseException(dwExceptionCode: 0x406d1388, dwExceptionFlags:
6ddf5018      noreturn
6ddf5018

```

During our investigation, we discovered that the hackers leased a VPS server from the following company and set up a C&C server that sent the encrypted data to DNS port 53 at the specified location, which was used to exfiltrate compromised data.

CRYPTBASE.DLL	strlen ("NDUUnzYuOS4yNDg=")
CRYPTBASE.DLL	calloc (12, 1)
CRYPTBASE.DLL	memset (0x00c21250, 0, 12)
CRYPTBASE.DLL	malloc (4)
CRYPTBASE.DLL	memset (0x00c21250, 0, 12)
CRYPTBASE.DLL	free (0x00c257f8)
CRYPTBASE.DLL	_strdup ("45.76.9.248")
CRYPTBASE.DLL	_strdup ("53")
CRYPTBASE.DLL	_lock (17)
CRYPTBASE.DLL	fputc (73, 0x75894620)

PDFSIDER is a robust and stealthy backdoor designed for long-term covert access, flexible remote command execution, and encrypted communications - aligning more with espionage tradecraft than financially motivated malware.

4. Decoy

Threat actors may leverage various decoys to lure victims into opening malicious documents. In one of the recent episodes of the same campaign, the actor also used a fake document designed to appear as if authored by the PLA Intelligence Bureau of the Joint Staff Department (中央军委联合参谋部情报局).



Notably, the majority of artifacts identified from the PDFSIDER campaign evade popular AV/EDR solutions. Currently, Resecurity interprets PDFSIDER as a tradecraft used in targeted

attacks rather than mass-scale delivery.

Indicators of Compromise (IOCs)

File Name	MD5 Hash	Status
About.dll	e0e674ec74d323e0588973aae901b5d2	Clean
Cryptbase.dll	298cbfc6a5f6fa041581233278af9394	Malicious
Language.dll	80e4a29270b828c1f97d9cde9475fcbd	Clean
NotifyIcon.dll	96ff508f9be007062b1770691f489e62	Clean
Pdf24.exe	a32dc85eee2e1a579199050cd1941e1d	Clean
Settings.dll	9f9dd5a432b4dde2160c7a7170e0d069	Clean

C2 IPs:

45.76.9.248

MITRE ATT&CK Mapping

Tactic	Technique	Mapping to PDFSIDER
Initial Access	T1574.002 - DLL Side-Loading	Malicious cryptbase.dll hijacks legitimate applications.
Execution	T1059.003 - Windows Command Shell	Executes hidden cmd.exe /C commands.
	T1204 - User Execution	User must run the parent executable that loads the DLL.
Defense Evasion	T1497 - Virtualization/Sandbox Evasion	CPU, RAM, timing, VHD boot checks.
	T1622 - Debugger Evasion	Uses IsDebuggerPresent .
Discovery	T1082 - System Information Discovery	Collects username, computer name, PID.
Command & Control	T1095 - Non-Application Layer Protocol	Custom Winsock C2 with AES-GCM.
	T1041 - Exfiltration Over C2 Channel	Sends captured command output over encrypted channel.

Tactic	Technique	Mapping to PDFSIDER
Execution	T1106 - Native API	Uses low-level Win32 APIs for pipe and process control.

Significance

Resecurity has learned about **PDFSIDER** during an investigation of a network intrusion attempt that was successfully prevented by a **Fortune 100 corporation**. The threat actor contacted their staff, impersonating technical support, and used social engineering tactics with **QuickAssist** in an attempt to gain remote access to their endpoint.

Considering DLL sideloading technique, this attack vector can be effectively exploited by advanced actors to **bypass antivirus (AV) and endpoint detection and response (EDR) systems**. According to our HUNTER team, PDFSIDER is already being actively used by several **ransomware actors** as a payload delivery method.

Resecurity may outline a continuing trend of targeted spear-phishing, favoring reliable execution techniques such as DLL sideloading over exploit-based initial access. For example, one of the recent malware campaigns identified by [Acronis](#), targeting the U.S. government and attributed to **LOTUSLITE**, leveraged such a vector combined with geopolitical narratives between the United States and Venezuela as thematic lures. Infrastructure analysis and execution patterns show a moderate-confidence overlap with **Mustang Panda** tradecraft, including delivery style, loader–DLL separation, and infrastructure usage. In another campaign identified by [Trellix](#), DLL sideloading was used to distribute a wide assortment of malware, such as Agent Tesla, CryptBot, Formbook, Lumma Stealer, Vidar Stealer, Remcos RAT, Quasar RAT, DCRat, and XWorm. DLL sideloading vulnerability was exploited in a legitimate binary associated with the open-source c-ares library to bypass security controls and deliver a broad range of commodity trojans and stealers.

Share Article

[Cyber Threat Intelligence](#)

[Doomsday for Cybercriminals — Data Breach of Major Dark Web Forum](#)

[Cyber Threat Intelligence](#)

[Cyber Counterintelligence \(CCI\): When 'Shiny Objects' trick 'Shiny Hunters'](#)

Newsletter

Keep up to date with the latest cybersecurity news and developments.

Email Address

By subscribing, I understand and agree that my personal data will be collected and processed according to the [Privacy](#) and [Cookies Policy](#).