Hamas-Affiliated Ashen Lepus Targets Middle Eastern Diplomatic Entities With New AshTag Malware Suite

By Unit 42

Published: 2025-12-11 · Archived: 2026-04-05 14:26:54 UTC

Executive Summary

In recent months, we have been analyzing the activity of an advanced persistent threat (APT) known for its espionage activities against Arabic-speaking government entities. We track this Middle Eastern threat actor as Ashen Lepus (aka WIRTE).

We share details of a long-running, elusive espionage campaign targeting governmental and diplomatic entities throughout the Middle East. We discovered that the group has created new versions of their previously documented custom loader, delivering a new malware suite that we have named AshTag. The group has also updated their command and control (C2) architecture to evade analysis and blend in with legitimate internet traffic.

Ashen Lepus remained persistently active throughout the Israel-Hamas conflict, distinguishing it from other affiliated groups whose activities decreased over the same period. Ashen Lepus continued with its campaign even after the October 2025 Gaza ceasefire, deploying newly developed malware variants and engaging in hands-on activity within victim environments.

This campaign highlights a tangible evolution in Ashen Lepus's operational security and tactics, techniques and procedures (TTPs). While its operations over the years have demonstrated only moderate sophistication, the group has recently adopted more advanced tactics that include:

- Enhanced custom payload encryption
- Infrastructure obfuscation using legitimate subdomains
- In-memory execution to minimize forensic artifacts

Palo Alto Networks customers are better protected from the threats described in this article through the following products and services:

- Advanced WildFire
- Advanced URL Filtering and Advanced DNS Security
- Cortex XDR and XSIAM

If you think you might have been compromised or have an urgent matter, contact the Unit 42 Incident Response team.

Ashen Lepus Background

We investigated a campaign waged by a Hamas-affiliated threat group that has been active since 2018. Their operations focus on cyber-espionage and intelligence collection, targeting government entities across the Middle East.

We attribute this activity with high confidence to Ashen Lepus. Our attribution is based on Unit 42's Attribution Framework, and takes into account the network infrastructure, modus operandi and malware that the group has used throughout their campaigns. The attribution artifacts are detailed in Appendix A.

Ashen Lepus Ops: Victimology and Motivation

Ashen Lepus is known for targeting entities in close geographical proximity, such as the Palestinian Authority, Egypt and Jordan. Recent campaigns show a significant expansion in operational scope – according to recent uploads to VirusTotal, the group is now targeting entities in other Arabic-speaking nations, including Oman and Morocco.

Despite the broader geographic footprint seen in their recent attacks, the group's lure themes remain largely consistent. The majority of lure themes continue to relate to Middle East geopolitical affairs, mainly those involving the Palestinian Territories. However, the current campaign shows an increase in lures related to Turkey and its relationship with the Palestinian administration. Table 1 details these themes.

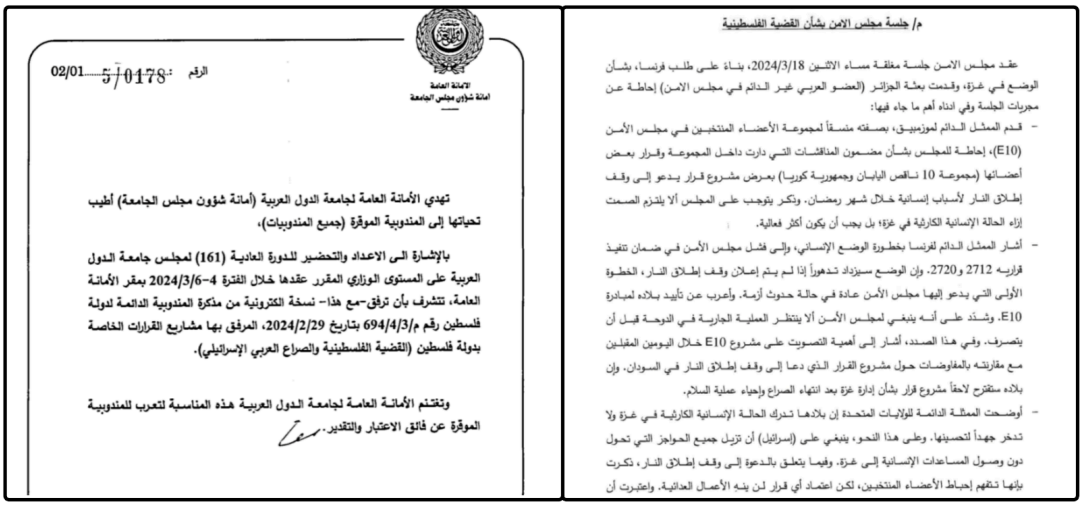
Lure Theme	Machine Translation
اتفاقية الشراكة بين المغرب وتركيا	Partnership agreement between Morocco and Turkey
وزير الدفاع التركي غيرنا استراتيجيتنا في 1302 مكافحة التنظيمات الارهابية	1302 Turkish Minister of Defense We changed our strategy in combating terrorist organizations
أنباء عن تدريب عناصر من حماس في سوريا تحديدا في الجنوب بدعم تركي	Reports of Hamas elements training in Syria, specifically in the south, with Turkish support
تقرير عن مقترح حماس لتوحيد السلاح الفلسطيني تحت مظلة السلطة	Report on Hamas's proposal to unify Palestinian arms under the umbrella of the Authority
مشاريع القرارات الخاصة بدولة فلسطين سري للغاية	Draft resolutions concerning the State of Palestine Top Secret

Table 1. Lure themes used in a recent Ashen Lepus campaign.

Breaking Down Ashen Lepus's Recent Campaign Developments

Decoy Archive Analysis

Since at least 2020 [PDF], Ashen Lepus has employed a consistent, multi-stage infection chain delivering a new malware suite that we call AshTag. The chain typically starts with a benign PDF decoy file that guides targets to a file-sharing service to download a RAR archive containing a malicious payload. Figure 1 shows two lure examples, relating to discussions conducted by the League of Arab States and United Nations Security Council.



الرقم: 02/01 5/0178

الأمانة العامة
أمانة شؤون مجلس الجامعة

تهدي الأمانة العامة لجامعة الدول العربية (أمانة شؤون مجلس الجامعة) أطيب تحياتها إلى المندوبية الموقرة (جميع المندوبيات)،

بالإشارة الى الاعداد والتحضير للدورة العادية (161) لمجلس جامعة الدول العربية على المستوى الوزاري المقرر عقدها خلال الفترة 4-2024/3/6 بمقر الأمانة العامة، تتشرف بأن ترفق-مع هذا- نسخة الكترونية من مذكرة المندوبية الدائمة لدولة فلسطين رقم م/694/4/3 بتاريخ 2024/2/29، المرفق بها مشاريع القرارات الخاصة بدولة فلسطين (القضية الفلسطينية والصراع العربي الإسرائيلي).

وتغتنم الأمانة العامة لجامعة الدول العربية هذه المناسبة لتعرب للمندوبية الموقرة عن فائق الاعتبار والتقدير.

م/ جلسة مجلس الامن بشأن القضية الفلسطينية

عقد مجلس الامن جلسة مغلقة مساء الاثنين 2024/3/18، بناءة على طلب فرنسا، بشأن الوضع في غزة، وقدمت بعثة الجزائر (العضو العربي غير الدائم في مجلس الامن) إحاطة عن مجريات الجلسة وفي ادناه أهم ما جاء فيها:

- قدم الممثل الدائم لموزمبيق، بصفته منسقاً لمجموعة الأعضاء المنتخبين في مجلس الأمن (E10)، إحاطة للمجلس بشأن مضمون المناقشات التي دارت داخل المجموعة وقرار بعض أعضائها (مجموعة 10 ناقص اليابان وجمهورية كوريا) بعرض مشروع قرار يدعو إلى وقف إطلاق النار لأسباب إنسانية خلال شهر رمضان. وذكر بتوجب على المجلس ألا يلتزم الصمت إزاء الحالة الإنسانية الكارثية في غزة؛ بل يجب أن يكون أكثر فعالية.
- أشار الممثل الدائم لفرنسا بخطورة الوضع الإنساني، والى فشل مجلس الأمن في ضمان تنفيذ قراريه 2712 و2720. وإن الوضع سيزداد تدهوراً إذا لم يتم إعلان وقف إطلاق النار، الخطوة الأولى التي يدعو إليها مجلس الأمن عادة في حالة حدوث أزمة. وأعرب عن تأييد بلاده لمبادرة E10. وشدد على أنه ينبغي لمجلس الأمن ألا ينتظر العملية الجارية في الدوحة قبل أن يتصرف. وفي هذا الصدد، أشار إلى أهمية التصويت على مشروع E10 خلال اليومين المقبلين مع مقارنته بالمفاوضات حول مشروع القرار الذي دعا إلى وقف إطلاق النار في السودان. وإن بلاده ستقترح لاحقاً مشروع قرار بشأن إدارة غزة بعد انتهاء الصراع وإحياء عملية السلام.
- أوضحت الممثلة الدائمة للولايات المتحدة إن بلادها تدرك الحالة الإنسانية الكارثية في غزة ولا تدخر جهداً لتحسينها. وعلى هذا النحو، ينبغي على (إسرائيل) أن تزيل جميع الحواجز التي تحول دون وصول المساعدات الإنسانية إلى غزة. وفيما يتعلق بالدعوة إلى وقف إطلاق النار، ذكرت بأنها تتفهم إحباط الأعضاء المنتخبين، لكن اعتماد أي قرار لن ينه الأعمال العدائية. واعتبرت أن

Figure 1. Lure examples presented to targets.

Downloading and opening the RAR archive initiates the chain of events that leads to an infection. This infection involves the following three files:

- A binary file masquerading as a sensitive or political document
- A malicious loader, which runs in the background
- An additional decoy PDF file named Document.pdf

When the targeted individual runs the binary in order to read the article, the binary side-loads the first malicious loader (netutils.dll), which in turn opens the decoy PDF file for viewing. Figure 2 illustrates the initial infection chain in Cortex XDR, showing alerts triggered by the Windows executables responsible for DLL side-loading and persistence.

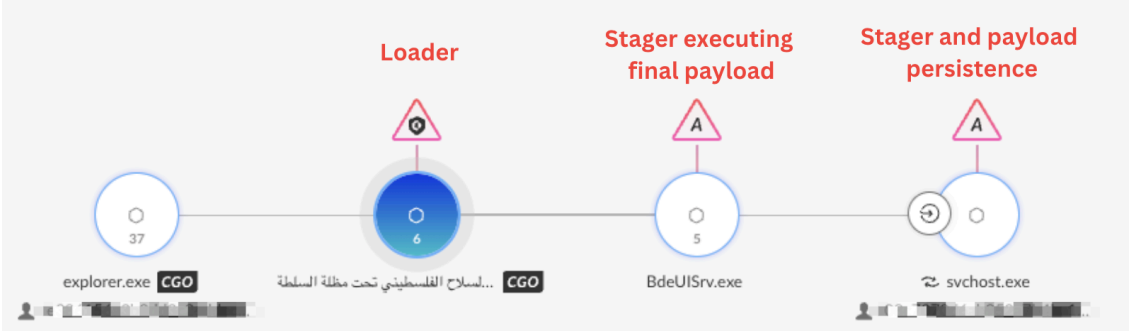


Figure 2. AshTag's initial infection chain and persistence, as seen in Cortex XDR.

C2 Architecture Evolution

Comparing this campaign with past campaigns shows that there has been a change in the group's C2 domain naming convention. Instead of hosting its C2 servers on its own domains, the group now registers new API and authentication-related subdomains of legitimate domains. This change is part of the group's shift to adopt better operational security (OpSec), and helps its activity blend in with benign network activity. The domains often have technology or medical themes, such as api[.]healthylifefeed[.]com, api[.]softmatictech[.]com and auth[.]onlinefieldtech[.]com.

We also observed a clear separation between different servers for different tools within the execution chain. The domains have varying formats and are hosted in multiple autonomous system numbers (ASNs). Since the servers are geofenced, automatic analysis tools cannot execute the entire chain to link between the different stages.

In this campaign, the group took several cautionary measures to avoid detection and analysis. For instance, the secondary payloads are embedded within HTML tags of a seemingly benign webpage. Also, the C2 server performs initial checks on the victim's endpoint, to avoid sending the payload to sandbox environments. The server checks the victim’s geolocation, and checks specific User-Agent strings in the traffic that are unique to the malware.

The New AshTag Malware Suite and Campaign Evolution

The AshTag campaign marks a significant upgrade to the group's traditional tooling. In previous campaigns, the actors did not deliver a full payload, and instead terminated the parent process using a simple .NET DLL. We assess that previous campaigns observed in the wild were a testing phase in the development of the attack chain. However, in this campaign, Ashen Lepus is deploying a more sophisticated, fully featured malware suite, which we have named AshTag. Unit 42 designates the name “Lepus” to threat groups associated with the Palestinian Territories, and we labeled the malware components “Ash” to reflect the basic, gritty attack resources that accumulate to choke system defenses, allowing the full attack to take hold.

AshTag is a modular .NET toolset currently in active development, with extensive features, including file exfiltration, content download and in-memory execution of additional modules.

The AshTag infection chain unfolds as follows:

- A targeted victim clicks the binary file, expecting to open a document.
- The binary file side-loads a DLL in the background. This DLL is the first malicious loader, which we call AshenLoader.
- AshenLoader opens the decoy PDF document on the desktop.
- In the background, AshenLoader retrieves and runs another side-loaded DLL: a stager that we call AshenStager.
- AshenStager retrieves and runs the AshTag payload.
- AshenStager also sets its persistence via a scheduled task, executed by svchost.exe.

Figure 3 depicts the complete attack chain.

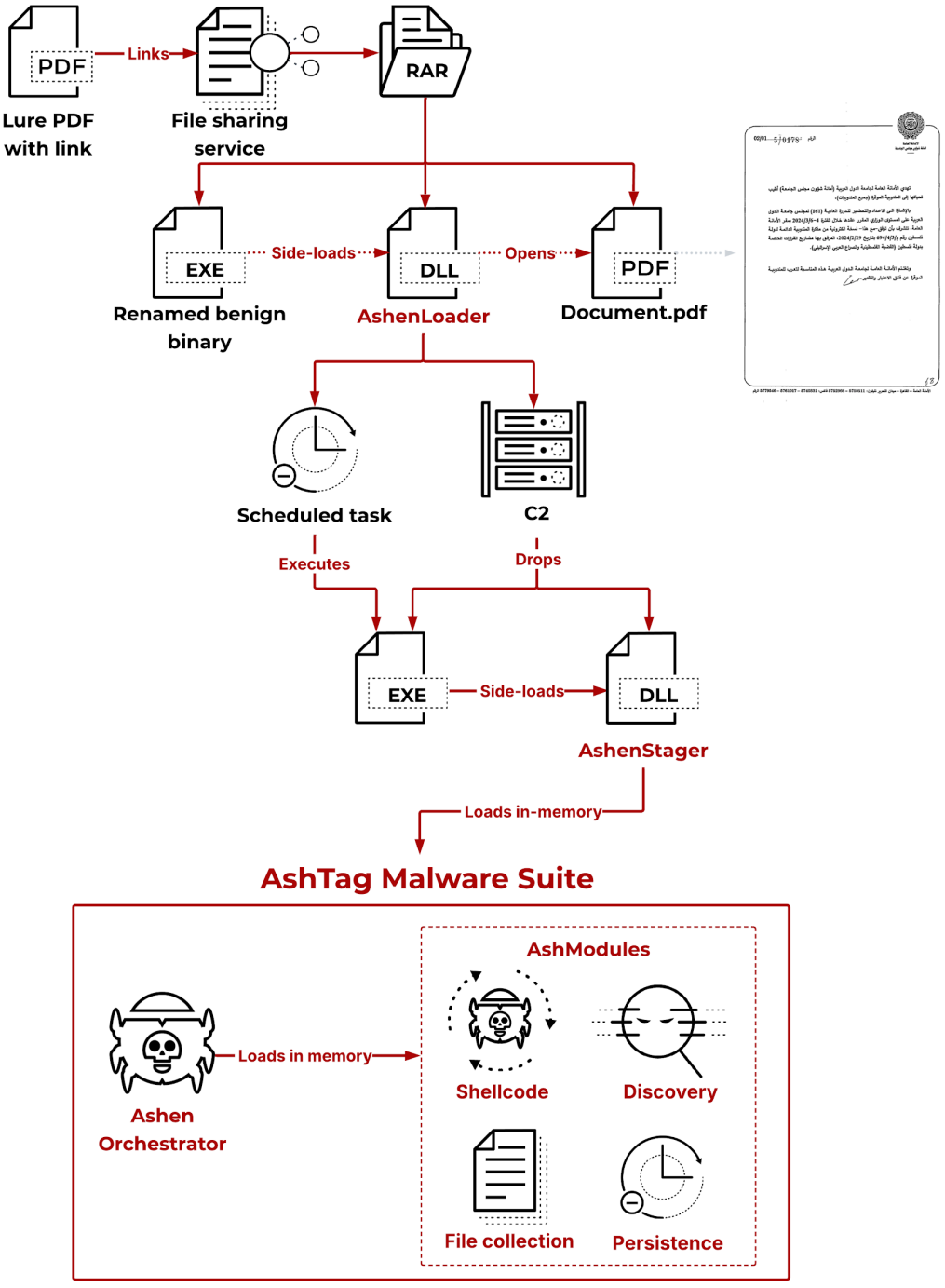


Figure 3. The full AshTag Malware infection chain.

Initial Loader Execution Flow

When AshenLoader is executed, it tries to collect and send initial reconnaissance data to the attacker's C2 server. The AshenStager payload is embedded within the C2's webpage, between the custom <headerp> HTML tags – an embedding method that has been documented in the past. In addition to these similarities, we identified new features of AshenLoader, described in Appendix B.

AshenLoader retrieves and executes a stager that we dub AshenStager. In past campaigns, this stager was named Stager-X64, following its internal naming by the attackers. We now track AshenStager as part of the AshTag malware suite. AshenStager is side-loaded by a legitimate executable paired with a malicious custom DLL, named wtsapi32.dll.

AshenStager is designed to send an HTTP request to its C2 server, where it parses the HTML response to extract another encrypted payload that is hidden within <article> tags. After extracting the payload, AshenStager decodes, parses and injects the payload in memory. The final payload in this chain is a malware suite, which is orchestrated by a tool that we call AshenOrchestrator. Figure 4 shows the orchestrator's Base64-encoded payload embedded in HTML content from the C2 server.

Figure 4. AshenOrchestrator's Base64-encoded payload embedded within the article HTML tags.

AshTag Malware Suite

AshTag is a modular .NET backdoor designed for stealthy persistence and remote command execution. AshTag masquerades as a legitimate VisualServer utility to evade suspicion. In reality, this backdoor is a multi-feature malware suite that uses AshenOrchestrator to conduct communication and to execute other payloads in memory.

When AshenStager retrieves AshenOrchestrator's payload, the stager receives a Base64-encoded JSON file. The JSON file contains the payload and the payload's configuration. The configuration contains parameters such as specific URL paths that lead to different modules, encryption keys and the C2 domain. The configuration also includes sleep time buffers (jitter), mn and mx, which are used to avoid detection of the C2 beaconing. Figure 5 shows an example of such a configuration.

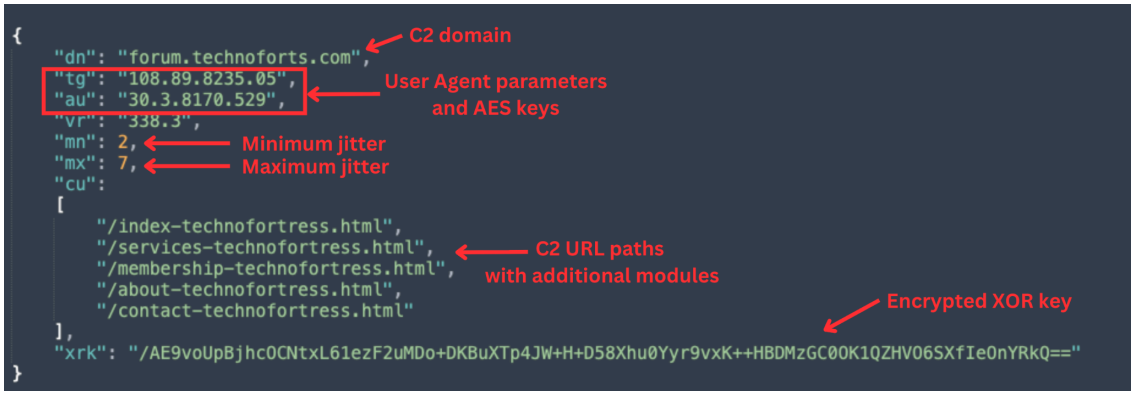


Figure 5. Decoded AshenOrchestrator configuration.

Like most of the tools used in this campaign, AshenOrchestrator extracts its next payload from embedded HTML tags. However, in this instance, the payload is even more well hidden. Instead of using a hardcoded tag name, the

stager searches for a specific commented-out tag within the HTML page that contains the relevant tag name. Figure 6 demonstrates the payload embedding scheme.

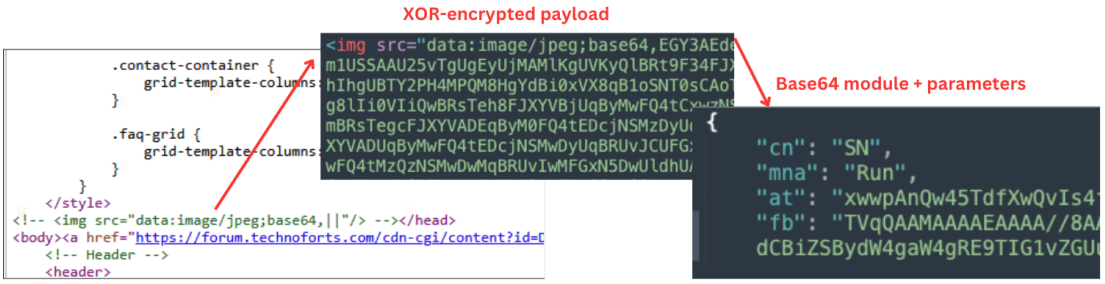


Figure 6. AshTag module decoding process.

AshenOrchestrator creates a unique AES key from the tg and au parameters, and decrypts the xrk XOR encryption key. The decrypted XOR key is then used to decrypt the embedded HTML value that contains the payload. The payload itself is a specific module contained in another Base64-encoded JSON that has additional configuration parameters. These parameters determine the module's loading method name (mna) and class name (cn). Table 2 lists the different class names that AshenOrchestrator expects and their inferred functionalities.

Class Name (cn)	**Inferred Purposes**
PR1, PR2, PR3	Persistence Process Management
UN1, UN2, UN3	Uninstall Update Removal
SCT	Screen Capture
FE	File Explorer File Management
SN	System Fingerprinting

Table 2. Different Ashen modules and their inferred purposes.

The mna value dictates the action that AshenOrchestrator performs for each module that it retrieves. There are four possible actions:

- Upload additional content
- Download the module to disk
- Execute the module as a .NET assembly

- Inject the module into memory

Analyzing the injection method revealed that its code was not actually implemented, and only returned false, indicating that certain aspects of the AshTag malware suite are still in active development.

Retrieving the different modules for analysis was a complicated task, in part because Ashen Lepus appears to be actively rotating the modules that are hidden within webpage content. This would explain why not all modules are available at the same time. In addition, we found that different encryption keys open different types of modules.

Despite these complicating factors, we were able to retrieve one of the modules responsible for system fingerprinting – internally named the SN module. The module is an extremely simple .NET program that executes WMI queries and sends a unique victim ID back to the attackers. Figure 7 shows the main function of the SN module.

```
// Token: 0x02000002 RID: 2
public class SN
{
    // Token: 0x06000001 RID: 1 RVA: 0x00002050 File Offset: 0x00000250
    public static object Run()
    {
        Hashtable hashtable = new Hashtable();
        try
        {
            string text = string.Empty;
            try
            {
                string text2 = WMIHelper.Select("root\\cimv2", "SystemDrive", "Win32_OperatingSystem", null);
                text = WMIHelper.Select("root\\cimv2", "volumeserialnumber", "Win32_logicalDisk", "DeviceID='" + text2 + "'");
            }
            catch
            {
                text = "20A8631F";
            }
            hashtable.Add("computer_id", text);
        }
        catch (Exception ex)
        {
            hashtable.Clear();
            hashtable.Add("error", ex.ToString());
        }
        return hashtable;
    }
}
```

Figure 7. Code from the SN fingerprinting module.

We identified the threat actor's operations in our telemetry, which indicated that they used additional modules to stage and exfiltrate files.

Ashen Lepus's Hands-On Activity

Following the initial automated infection, the threat actor accessed the compromised system to conduct hands-on data theft. A few days after the original infection, the attackers loaded a custom module via AshenOrchestrator and began staging specific documents in the C:\Users\Public folder.

Our analysis indicates that the threat actor downloaded these documents directly from a victim's mail accounts, revealing the group's main objective: obtaining specific, diplomacy-related documents. This aligns with past reports of the group's practice of obtaining intelligence relating to regional geopolitical conflicts.

To exfiltrate the staged files, Ashen Lepus downloaded the Rclone open-source tool, transferring the data to an attacker-controlled server. This appears to be the first time this threat group has been observed using Rclone for data exfiltration. In doing so, Ashen Lepus joins a growing number of actors who leverage legitimate file transfer tools to blend their malicious activity with benign network traffic and avoid detection.

Conclusion

Ashen Lepus remains a persistent espionage actor, demonstrating a clear intent to continue its operations throughout the recent regional conflict – unlike other affiliated threat groups, whose activity significantly decreased. The threat actors' activities throughout the last two years in particular highlight their commitment to constant intelligence collection.

During this campaign, Ashen Lepus has begun to deliver its new malware suite, AshTag. AshTag is a modular .NET suite, capable of data exfiltration, command execution and in-memory payload execution.

While the group's core TTPs are not highly sophisticated, this campaign reveals an evolution in its approach. We observed a clear effort to improve operational security by enhancing payload encryption, shifting infrastructure to innocent-looking subdomains and executing payloads in memory. This "low-cost, high-impact" methodology allows the threat actors to effectively evade static defenses and thwart analysis.

The expansion of Ashen Lepus's victimology beyond their traditional geographic targets, coupled with new lure themes, suggests a broadening of its operational scope. We assess that Ashen Lepus will continue to adapt its toolset and targeting to pursue its geopolitical intelligence objectives. Organizations in the Middle East, particularly in the governmental and diplomatic sectors, should remain vigilant against this evolving threat.

Palo Alto Networks customers are better protected from the threats described in this article through the following products and services:

- The Advanced WildFire machine-learning models and analysis techniques have been reviewed and updated in light of the indicators shared in this research.
- Advanced URL Filtering and Advanced DNS Security identify known domains and URLs associated with this activity as malicious.
- Cortex XDR and XSIAM
    - Cortex XDR helps to prevent the threats described in this blog, by employing the Malware Prevention Engine. This approach combines several layers of protection, including Advanced WildFire, Behavioral Threat Protection and the Local Analysis module, to help prevent both known and unknown malware from causing harm to endpoints.

If you think you may have been compromised or have an urgent matter, get in touch with the Unit 42 Incident Response team or call:

- North America: Toll Free: +1 (866) 486-4842 (866.4.UNIT42)
- UK: +44.20.3743.3660
- Europe and Middle East: +31.20.299.3130
- Asia: +65.6983.8730
- Japan: +81.50.1790.0200
- Australia: +61.2.4062.7950
- India: 000 800 050 45107
- South Korea: +82.080.467.8774

Palo Alto Networks has shared these findings with our fellow Cyber Threat Alliance (CTA) members. CTA members use this intelligence to rapidly deploy protections to their customers and to systematically disrupt malicious cyber actors. Learn more about the Cyber Threat Alliance.

Indicators of Compromise

SHA256 Hashes of Malware Samples

RAR Archives

- 3502c9e4896802f069ef9dcdba2a7476e1208ece3cd5ced9f1c4fd32d4d0d768
- 1f3bd755de24e00af2dba61f938637d1cc0fbfd6166dba014e665033ad4445c0
- 4e1f7b48249dd5bf3a857d5d017f0b88c0372749fa156f5456056767c5548345
- 3d445c25752f86c65e03d4ebed6d563d48a22e424ba855001ad2db2290bf564c
- 7e5769cd8128033fc933fbf3346fe2eb9c8e9fc6aa683546e9573e7aa01a8b6b

AshenLoader Variant #1

- f554c43707f5d87625a3834116a2d22f551b1d9a5aff1e446d24893975c431bc - dwampi.dll
- a17858f40ff506d59b5ee1ba2579da1685345206f2c7d78cb2c9c578a0c4402b - dwampi.dll
- ebe3b6977f66be30a22c2aff9b50fec8529dfa46415ea489bd7961552868f6b5 - dwampi.dll
- 8870bd358d605a5685a5f9f7785b5fee5aebdcb20e4e62153623f764d7366a3c - dwampi.dll
- 2d71d7e6ffecab8eefa2d6a885bcefe639fca988bdcac99e9b057e61698a1fd6 - dwampi.dll
- 8c44fa9bf68341c61ccaca0a3723945543e2a04d9db712ae50861e3fa6d9cc98 - wtsapi32.dll
- f380bd95156fbfb93537f35941278778819df1629cb4c5a4e09fe17f6293b7b7 - wtsapi32.dll

AshenLoader Variant #2

- f9816bc81de2e8639482c877a8defcaed9b15ffdce12beaef1cff3fea95999d4 - srvcli.dll
- e71a292eafe0ca202f646af7027c17faaa969177818caf08569bd77838e93064 - srvcli.dll
- 739a5199add1d970ba22d69cc10b4c3a13b72136be6d45212429e8f0969af3dc - netutils.dll
- b00491dc178a3d4f320951bccb17eb85bfef23e718b4b94eb597c90b5b6e0ba2 - netutils.dll

AshenStager

- 6bd3d05aef89cd03d6b49b20716775fe92f0cf8a3c2747094404ef98f96e9376 - wtsapi32.dll

AshenOrchestrator

- 30490ba95c42cefcca1d0328ea740e61c26eaf606a98f68d26c4a519ce918c99

AshTag Module Designated as "SN"

- 66ab29d2d62548faeaeadaad9dd62818163175872703fda328bb1b4894f5e69e

AES Keys and Nonce

AshenLoader Variant #1

- Key: {9a 20 51 98 4a 2b b1 76 ef 98 87 e3 be 87 f9 ca 44 ba 8c 19 a8 ef ba 55 62 98 e1 2a 39 21 ea 8b}
- Nonce: {44 ba 8c 19 a8 ef ba 55 62 98 e1 2a 39 21 ea 8b}

AshenLoader Variant #2

- Key: {60 3d eb 10 15 ca 71 be 2b 73 ae f0 85 7d 77 81 1f 35 2c 07 3b 61 08 d7 2d 98 10 a3 09 14 df f4} (generic default key)
- Nonce: {f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff} (generic default nonce)
- AshenStager XOR Key: msasn1.dll

C2 Domains

Backdoor

- forum.techtg[.]com
- forum.technoforts[.]com

Exfiltration Server

- api.technology-system[.]com

Loaders Variant #1

- api.healthylifefeed[.]com
- api.softmatictech[.]com
- apiv2.onlinefieldtech[.]com
- auth.onlinefieldtech[.]com
- status.techupinfo[.]com
- api.medicinefinders[.]com
- account.techupinfo[.]com

Loaders Variant #2

- api.systemsync[.]info
- api.widetechno[.]info

Scheduled Task Names

- C:\Windows\System32\Tasks\Windows\WindowsDefenderUpdate\Windows Defender Updater
- C:\Windows\System32\Tasks\Windows\WindowsServicesUpdate\Windows Services Updater
- C:\Windows\System32\Tasks\Automatic Windows Update

Appendix A: Attribution

Our assessment utilizes the Unit 42 Attribution Framework, which provides a systematic, evidence-based methodology to connect observed malicious activity to specific threat groups. This approach moves beyond subjective assessments, allowing us to rigorously evaluate multiple dimensions of threat data, including TTPs, tooling, OpSec, network infrastructure and victimology.

Tactics, Techniques and Procedures (TTPs)

There is a significant overlap between this campaign and Ashen Lepus's established modus operandi. The group consistently crafts lures written in Arabic that focus on the developing political and military situation in the Middle East, with a specific emphasis on the Palestinian Territories.

While public reporting on the group's post-compromise activity is limited, the hands-on espionage actions observed in this incident – specifically, the targeted theft of diplomatic documents – strongly correlate with the group's known intelligence collection interests and sophistication level.

Infrastructure Overlaps

We identified clear infrastructure overlaps with historic reporting on the group. Specifically, the URL structure observed in this campaign aligns with findings from Check Point. For example, the URL cited in their report has the same subdomain naming scheme and URL parameter structure that we observed in previous loader versions (api/v1.0/account?token=):

- hxxps://**support-api**[.]financecovers[.]com/**api/v1.0/account?token=**{encrypted_recon_data}

A similar URL was also documented in OWN Security's report:

- hxxps://**cdn**[.]techpointinfo[.]com/**api/v1.0/account?token=**{encrypted_recon_data}

Malware Artifacts

Analysis of the loader reveals key features consistent with previous campaigns from this group, as documented by Check Point. Notably, the loader continues to embed next-stage payloads within HTML tags of seemingly benign webpages and utilizes similarly structured execution lures to initiate the infection chain. The group also uses the same file names for their payloads – both their SharpStage .NET backdoor and previous versions of their loader were named wtsapi32.dll.

Appendix B: The Development of New Loader Versions

AshenLoader is a possible evolution of the group's previous IronWind loader. Throughout 2025, Ashen Lepus was actively tweaking AshenLoader, which for the most part retained the same functionality. In addition to AshenLoader's ability to communicate to the C2 server to download and execute additional payloads, the following features were updated:

- **Encryption algorithm:** The threat actors implemented an AES-CTR-256 cipher in versions of the malware that they compiled from early to late 2025, in contrast to the TEA algorithm mentioned in previous research. In samples that were compiled from mid to late 2025, the actors modified the encryption key and counter value (nonce) values. In both variants, the nonce and AES keys are hardcoded into the binaries.
- **Fingerprinting additional data from infected endpoints:** The new variants provide the threat actors with more detailed information about the infected endpoint than previous versions – such as listing files under the ProgramFiles directory.
- **URI updates:** Variants discussed in previous public research used the token parameter sent in the initial beaconing GET request. The earlier 2025 variants shifted toward using id= and q= parameters. Late 2025 variants then changed the scheme again and started using auth=. Additionally, part of the URI changed from /v1/ to /v2/.

Although these features do not significantly change the loader's functionality, they are simple and effective ways to avoid static detection engines.
