National Cyber Security Centre
a part of GCHQ

Jolly Jellyfish

Malware Analysis Report

TLP WHITE

Version 1.0

NCSC/MAR/W/00011

15th December 2021

National Cyber Security Centre
a part of GCHQ

Jolly Jellyfish

Non-persistent downloader for shellcode embedded in image files

Executive summary

- Jolly Jellyfish downloads and executes shellcode, which is hidden in legitimate bitmap image (BMP) files
- HTTP is used to download image files
- Debug information left in the samples suggest the author refers to this malware as `fishmaster`

Introduction

Jolly Jellyfish is a Windows executable which downloads and executes shellcode from a hard-coded remote server, using HTTP. Steganography is used to hide the shellcode inside BMP files. Once extracted and, where necessary, decoded, the shellcode is a Cobalt Strike stager. This report covers the analysis of multiple related samples, some of which include additional capabilities. No persistence is implemented by this malware.

National Cyber Security Centre
a part of GCHQ

Malware Details

Metadata

Filename	abcd461bdb6a6537b7a36848a87b5ea6.virus
Description	Jolly Jellyfish shellcode downloader which also downloads and executes a legitimate file
Size	393216 bytes
MD5	abcd461bdb6a6537b7a36848a87b5ea6
SHA-1	e99d5a620a488133f4da24e1f8d2d5e68542b6f3
SHA-256	f21a9c69bfca6f0633ba1e669e5cf86bd8fc55b2529cd9b064ff9e2e129525e8
Compile time	2021-02-26 07:16:23

Filename	Browser_plugin (8).exe
Description	Jolly Jellyfish shellcode downloader, of XOR-obfuscated shellcode, additionally displays a message box on execution
Size	109568 bytes
MD5	a241ff3d86925a4a12916b401536b019
SHA-1	d28eacb1b4d2e9ef54f7dff09ca03a6866fc9184
SHA-256	a7e9e2bec3ad283a9a0b130034e822c8b6dfd26dda855f883a3a4ff785514f97
Compile time	2021-03-12 06:25:25

Filename	Browser_Plugin.exe
Description	Jolly Jellyfish shellcode downloader of XOR-obfuscated shellcode, additionally downloads a legitimate file and displays a message box on execution
Size	178688 bytes
MD5	738f46546f6d4a79e2d917b26bf8a93a
SHA-1	834e80f6fa9935fd3184c25e4e37b0a068a773ee
SHA-256	4a43fa8a3305c2a17f6a383fb68f02515f589ba112c6e95f570ce421cc690910
Compile time	2021-04-06 03:11:40

Filename	bk.exe
Description	Jolly Jellyfish shellcode downloader
Size	177152 bytes
MD5	014dac67e8c32a25ccb024d1d1017b58
SHA-1	ba5558d79dadc12bbbe07e3444441d51d5e5931e
SHA-256	0df3b6e2535f8bb564183ab4e5e47d9b30ffc0204cc5bda1bae8984cdc418410
Compile time	2021-07-02 02:21:50

Filename	x37.bmp
Description	Jolly Jellyfish payload
Size	800682 bytes
MD5	a8c4ac44a5aa9d22319fe4b20cc5e790
SHA-1	7c348809e99c0be3ba5c122009a2cd15ad50b7bf
SHA-256	f0449c41bc3eebb8ea025fafc5b0cd1fcbe9a2d80c447ecc00cf3cab43e1c311

National Cyber Security Centre
a part of GCHQ

MITRE ATT&CK®

This report has been compiled with respect to the MITRE ATT&CK® framework, a globally accessible knowledge base of adversary tactics and techniques based on real-world observations.

Tactic	ID	Technique	Procedure
Defense Evasion	T1497.001	Virtualization/Sandbox Evasion: System Checks	Jolly Jellyfish checks the available memory is greater than 1GB, the size of the disk is greater than 1GB, and the number of logical processors is greater than 0. These checks are designed to avoid running on a machine with low resources, as virtual machines and sandboxes are more likely to have low resources. If these checks fail the process will exit before any malicious behaviour occurs, avoiding detection.
	T1497.003	Virtualization/Sandbox Evasion: Time Based Evasion	Jolly Jellyfish adds short sleep commands throughout execution, which could be an attempt to slow down execution enough to evade detection by automated analysis platforms.
Command and Control	T1071.001	Application Layer Protocol: Web Protocols	Jolly Jellyfish downloads shellcode over HTTP.
	T1001	Data obfuscation	Some variants of Jolly Jellyfish download XOR-encoded shellcode.
	T1001.002	Data obfuscation: Steganography	Jolly Jellyfish downloads shellcode contained within bitmap image (BMP) files.

National Cyber Security Centre
a part of GCHQ

Functionality

Overview

Jolly Jellyfish downloads a bitmap image file containing embedded shellcode over HTTP and extracts, decodes and executes it. The format is described in the 'Functionality (Steganography)' section below. The malware also employs several techniques to evade detection, which vary across different samples and are described in the 'Functionality (Defence evasion)' section of this report.

Persistence

This malware does not implement any persistence mechanisms.

Steganography

Jolly Jellyfish downloads a file with a '.bmp' file extension, which contains embedded shellcode. The downloaded image is shown below in Figure 3.

Figure 1 shows a legitimate BMP image, which begins with `0x0A` bytes of data (corresponding to the BMP header), followed by a 4-byte little-endian offset value. In the BMP format specification, this value is the offset to the start of the pixel array for the image.

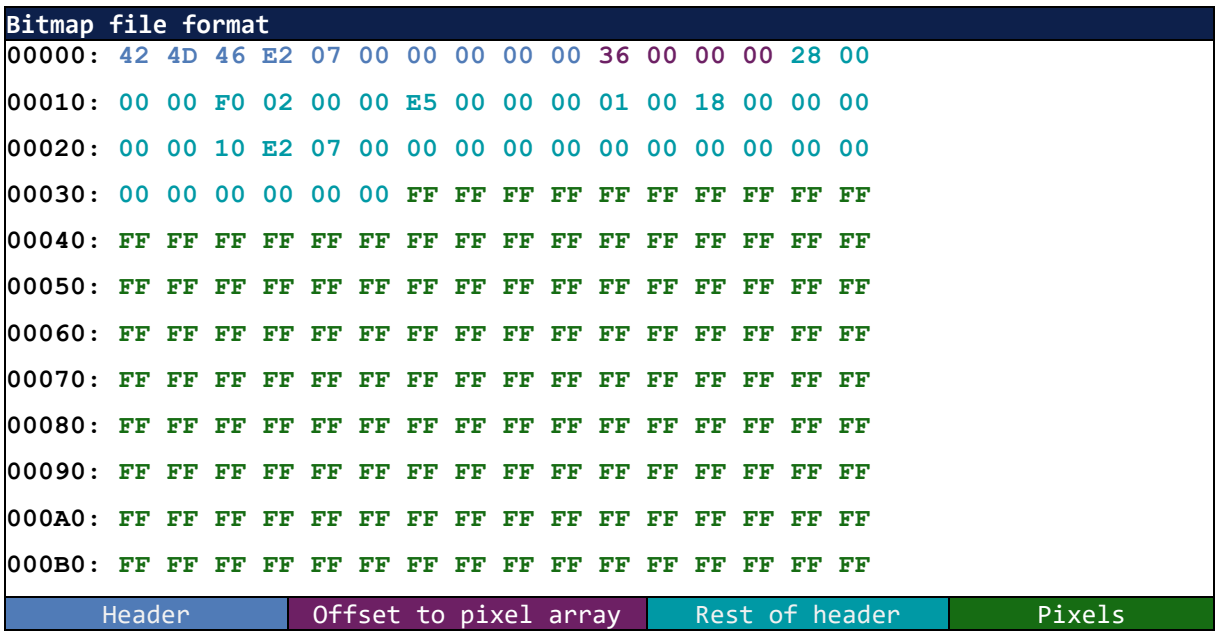


Figure 1: BMP file format

In the downloaded BMP image, shellcode is embedded in, and extracted from, this pixel array by reading every fourth byte, as shown in Figure 2. The shellcode is always null-terminated and has a maximum size of 1900 bytes.

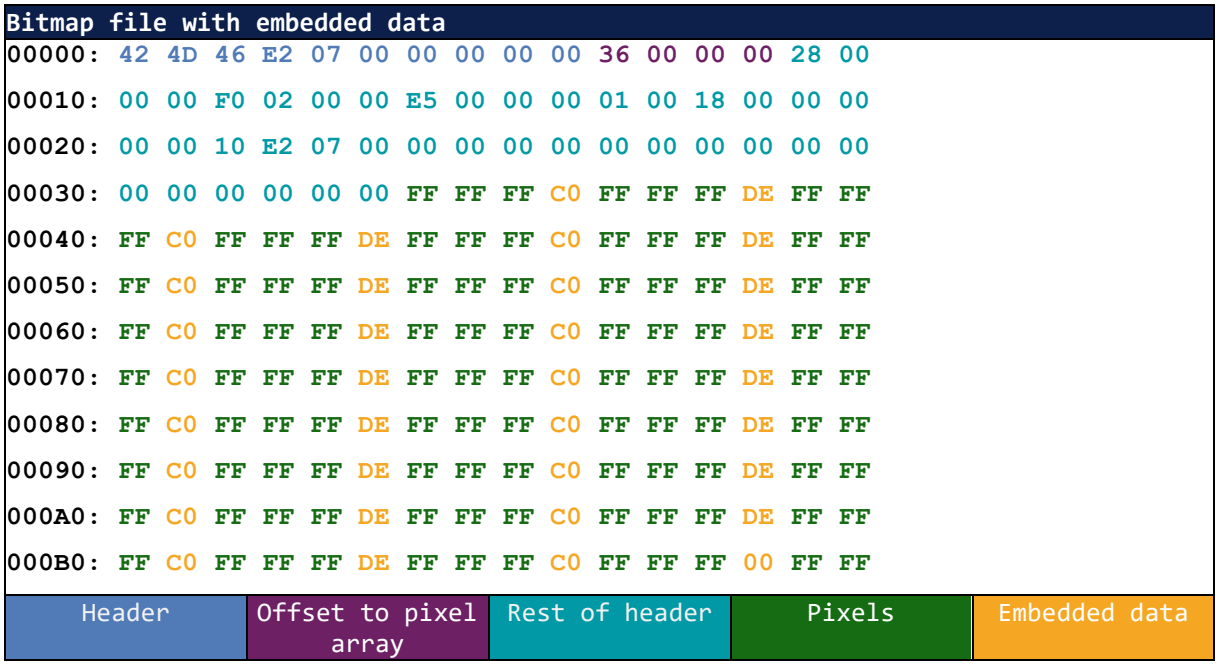


Figure 2: BMP file format with embedded data

Due to the comparatively small number of pixels affected by this encoding method, any difference from a legitimate image is likely to be hard to notice unless looked for. Figure 3 shows the modified image, and Figure 4 shows a closeup of a section containing embedded data.

Figure 3: downloaded BMP image with embedded data

Figure 4: downloaded BMP – closeup to highlight embedded data

Shellcode

Once extracted, the shellcode is a Cobalt Strike stager with the following configuration:

Type	Values
C2 server	download.google-images[.]ml
GET URI	/kXe5
User agent	Mozilla/5.0 (Windows NT 6.1; rv24.0) Gecko/20100101 Firefox/24.0

Message box

In some variants of Jolly Jellyfish, a message box is displayed.

It is not clear what the purpose of this message box is, but the samples containing this functionality have file names similar to 'Browser_plugin.exe'. This could indicate that they are intended to be deployed using social engineering, with the message box intended to make the target think that an error has occurred. This is further supported by several samples appearing to masquerade as legitimate files, as described in the 'Functionality (Defence evasion)' section of this report.

The message data is as follows, which is neither standard ASCII nor valid Unicode:

```
E4 AF C0 C0 C6 F7 B2 E5 BC FE D2 D1 BE AD B3 C9 B9 A6 B8 FC D0 C2 A3 AC C7
EB D6 D8 C6 F4 E4 AF C0 C0 C6 F7 A3 A1
```

In any case, the `MessageBoxA` API call is used, which expects the provided message to be ASCII characters, so the message is displayed as random characters as shown in Figure 4.

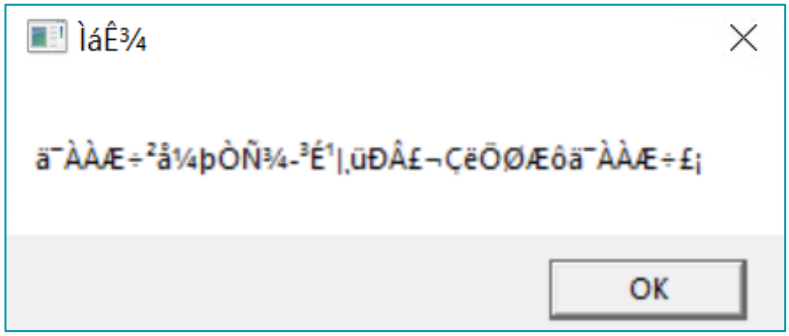


Figure 4: Message box containing random characters

Defence evasion

Shellcode obfuscation

In some samples, the shellcode is XOR-encoded, using a fixed multi-byte key. The keys used in the analysed samples are `misgat_mg` and `mait_mg`.

Sandbox detection

This malware implements some basic anti-sandbox/anti-virtual machine (VM) techniques. This includes checking the physical memory size of the machine exceeds 1GB and that the disk is more than 1GB in size. It also checks that the number of logical processors exceeds zero, although it is unknown why this check occurs. If any of these checks fail it will exit.

Anti-dynamic analysis

There are several sleep commands throughout the code, which slow down execution. This could be an attempt to prevent dynamic analysis solutions successfully detecting the malicious behaviour, although they are likely to be short to be effective.

There is a one second sleep between each API call involved in allocating and copying memory, as well as various sleep commands with durations between 1-2 seconds throughout each of the analysed samples.

Masquerading as legitimate file

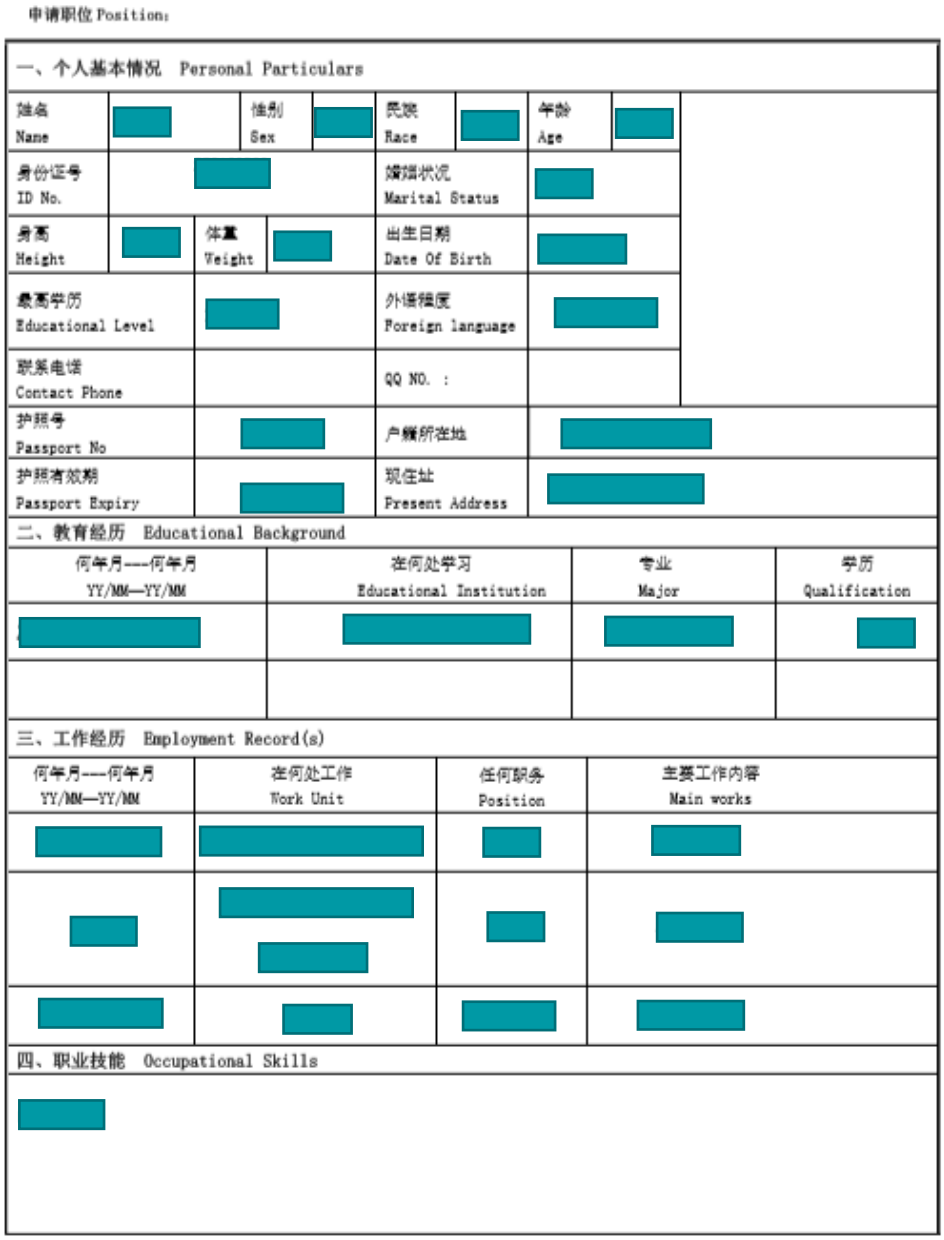
Some variants of Jolly Jellyfish download and execute what are apparently legitimate files in addition to the shellcode. This may indicate that targets are tricked into executing these files, which masquerade as legitimate applications or resources.

Examples of downloaded, apparently legitimate, files are shown below:

Download URL	http://37.61.205[.]212:8880/dow/Aili.pdf
Description	Chinese and English language form, shown in Figure 5
Size	103402 bytes
MD5	a465f18c7e50500c6b6f94741ef56b2f
SHA-1	b5053ceba7e45c956f601e77ed1ca4546f372221
SHA-256	17b4cf337cf4fa466a4a1bdc69795c2f96ef7b42464839dafbaf8502e28a3193
Executed?	No

Download URL	https://monpass[.]mn/storage/uploads/download_files/Monpass.Client.Install.exe
Description	Apparently legitimate application from MonPass, a Mongolian Certificate Authority (CA).
Executed?	Yes

National Cyber Security Centre
a part of GCHQ



申请职位 Position:

一、个人基本情况 Personal Particulars

姓名 Name	[redacted]	性别 Sex	[redacted]	民族 Race	[redacted]	年龄 Age	[redacted]
身份证号 ID No.	[redacted]			婚姻状况 Marital Status	[redacted]		
身高 Height	[redacted]	体重 Weight	[redacted]	出生日期 Date Of Birth	[redacted]		
最高学历 Educational Level	[redacted]			外语程度 Foreign language	[redacted]		
联系电话 Contact Phone				QQ NO. :			
护照号 Passport No	[redacted]			户籍所在地	[redacted]		
护照有效期 Passport Expiry	[redacted]			现住址 Present Address	[redacted]		

二、教育经历 Educational Background

何年月---何年月 YY/MM—YY/MM	在何处学习 Educational Institution	专业 Major	学历 Qualification
[redacted]	[redacted]	[redacted]	[redacted]

三、工作经历 Employment Record(s)

何年月---何年月 YY/MM—YY/MM	在何处工作 Work Unit	任何职务 Position	主要工作内容 Main works
[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]

四、职业技能 Occupational Skills

[redacted]

Figure 5: PDF form downloaded by Jolly Jellyfish sample

Communications

Jolly Jellyfish uses a HTTP GET request to retrieve the obfuscated shellcode.

Conclusion

Jolly Jellyfish is a low sophistication downloader containing some basic anti-dynamic analysis functionality. It contains easily signatured fixed strings and uses HTTP communications, which present straightforward detection opportunities. Of particular note, however, is the use of steganography to hide the downloaded shellcode in otherwise correctly formatted bitmap files.

The variation of functionality across different samples suggests that Jolly Jellyfish may often be deployed using social engineering.

The common PDB path:
`C:\Users\test\Desktop\fishmaster\x64\Release\fishmaster.pdb`
is also used in a different piece of malware that contains a Cobalt Strike Stager. This stager is configured to connect to the same domain as some Jolly Jellyfish variants.

National Cyber Security Centre
a part of GCHQ

Detection

Static strings

Each of the analysed samples contain the string `BidenHappyHappyHappy`. This string doesn't appear to have any practical functionality but can be used as an indicator to detect these binaries.

Several samples also contain the PDB path
`C:\Users\test\Desktop\fishmaster\x64\Release\fishmaster.pdb`

Indicators of compromise

Type	Description	Values
URL	URL for downloading shellcode	http://download.google-images[.]ml:8880/downloa/37.bmp
URL	URL for downloading shellcode	http://download.google-images[.]ml:8880/download/x37.bmp
URL	URL for downloading shellcode	http://micsoftin[.]us:2086/dow/83.bmp
URL	URL for downloading shellcode	http://172.20.10[.]6/save.bmp

Rules and signatures

Description	Detects the "Bidenhappyhappyhappy" string used by the Jolly Jellyfish malware
Precision	No false positives seen from Virus Total retrohunts.
Rule type	YARA

```
rule JollyJellyfish_unique_string_Bidenhappyhappyhappy
{
    meta:
        author = "NCSC"
        reference = " NCSC/MAR/W/00011"
        description = "Detects the "Bidenhappyhappyhappy" string used by
the Jolly Jellyfish malware"

    strings:
        $1 = "Bidenhappyhappyhappy"

    condition:
        uint16(0) == 0x5A4D and
        uint32(uint32(0x3c)) == 0x00004550 and
        all of them
}
```

National Cyber Security Centre
a part of GCHQ

Description	Detects the string displayed by the message box in some variants of Jolly Jellyfish
Precision	No false positives seen from Virus Total retrohunts.
Rule type	YARA

```
rule JollyJellyfish_unique_messagebox_display_string
{
    meta:
        author = "NCSC"
        reference = "NCSC/MAR/W/00011"
        description = "Detects the string displayed by the message box in
some variants of Jolly Jellyfish"

    strings:
        $popuptext = {E4 AF C0 C0 C6 F7 B2 E5 BC FE D2 D1 BE AD B3 C9 B9
A6 B8 FC D0 C2 A3 AC C7 EB D6 D8 C6 F4 E4 AF C0 C0 C6 F7 A3 A1}

    condition:
        uint16(0) == 0x5A4D and
        uint32(uint32(0x3c)) == 0x00004550 and
        all of them
}
```

Description	Detects Jolly Jellyfish check for memory being greater than 1GB
Precision	No false positives seen from Virus Total retrohunts.
Rule type	YARA

```
rule JollyJellyfish_check_memory_greater_1gb
{
    meta:
        author = "NCSC"
        reference = "NCSC/MAR/W/00011"
        description = "Detects Jolly Jellyfish check for memory being
greater than 1GB"
    strings:
        $1 = {33 D2 48 8B 44 ?? 38 B9 00 04 00 00 48 F7 F1 33 D2 B9 00 04
00 00 48 F7 F1 89 44 ?? ?? 81 7C ?? ?? 00 04 00 00}
        $2 = {48 8B 44 ?? 38 48 C1 E8 14 ?? 00 04 00 00}
    condition:
        uint16(0) == 0x5A4D and
        uint32(uint32(0x3c)) == 0x00004550 and
        any of them
}
```

National Cyber Security Centre
a part of GCHQ

Description	Detects the Jolly Jellyfish PDB string
Precision	No non-malicious results, but also detects malware containing Cobalt Strike stager
Rule type	YARA

```
rule JollyJellyfish_pdb_string
{
    meta:
        author = "NCSC"
        reference = "NCSC/MAR/W/00011"
        description = "Detects the Jolly Jellyfish pdb string"

    strings:
        $pdb = "fishmaster.pdb"

    condition:
        uint16(0) == 0x5A4D and
        uint32(uint32(0x3c)) == 0x00004550 and
        any of them
}
```

Description	Detects Jolly Jellyfish finding the start address of the shellcode in the downloaded data
Precision	No false positives seen from virus total retrohunts.
Rule type	YARA

```
rule JollyJellyfish_identify_shellcode_start_addr
{
    meta:
        author = "NCSC"
        reference = "NCSC/MAR/W/00011"
        description = "Detects Jolly Jellyfish finding the start address
of the shellcode in the downloaded data"

    strings:
        $1 = {48 89 84 24 ?? 00 00 00 48 8B 84 24 ?? 00 00 00 8B 40 0A 48
8B 4C 24 ?? 48 8D 44 01 03}
        $2 = {8B 43 0A 48 83 C0 03 48 03 D8}
    condition:
        uint16(0) == 0x5A4D and
        uint32(uint32(0x3c)) == 0x00004550 and
        any of them
}
```

National Cyber Security Centre
a part of GCHQ

Credits

Secureworks for sharing a copy of the downloaded Bitmap file.

Disclaimer

This report draws on information derived from NCSC and industry sources. Any NCSC findings and recommendations made have not been provided with the intention of avoiding all risks and following the recommendations will not remove all such risk. Ownership of information risks remains with the relevant system owner at all times.

This information is exempt under the Freedom of Information Act 2000 (FOIA) and may be exempt under other UK information legislation.

Refer any FOIA queries to ncscinfoleg@ncsc.gov.uk.