Detecting Datper Malware from Proxy Logs - JPCERT/CC Eyes

By JPCERT/CC

Published: 2017-08-20 · Archived: 2026-04-06 00:21:29 UTC

This is Yu Nakamura from Analysis Center.

This entry is to explain features of Datper, malware used for targeted attacks against Japanese organisations and how to detect it from the logs.

JPCERT/CC has been observing attacks using Datper since around June 2016. Research reports on the adversary are published from LAC [1], SecureWorks [2] and Palo Alto Networks [3]. The adversary had also conducted attacks using Daserf malware in the past, and Symantec refers to them as “Tick” in their report [4].

Attack vectors

We have confirmed that Datper infection occurs by:

- Drive-by download attacks
- Exploiting vulnerabilities in asset management software

In the former attack vector, we observed that a vulnerability of Adobe Flash Player (CVE-2016-7892) was leveraged for downloading and executing Datper. For the latter, there were cases where devices also got infected with a downloader called “wali”. Some analysis of this downloader has been published by Kaspersky [5] and Cybereason [6]. We have seen that wali can download several types of malware, and Datper is one of them.

Detailed behaviour

Datper communicates with a C&C server using HTTP protocol and operates based on the received commands. One of the characteristics is that it only communicates within a specific period of time.

Here below is a sample HTTP request that Datper sends to a C&C server. User-Agent is hard-coded in the malware.

```
GET /hoge/index.php?fnyup=940785246f0c22b41joikeddfngjokyptui HTTP/1.1
Accept: */*
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
Host: [host name]
Pragma: no-cache
Connection: close
```

The malware receives a command as a response to the above HTTP request, and it executes functions based on the commands. Functions that Datper can execute are the following:

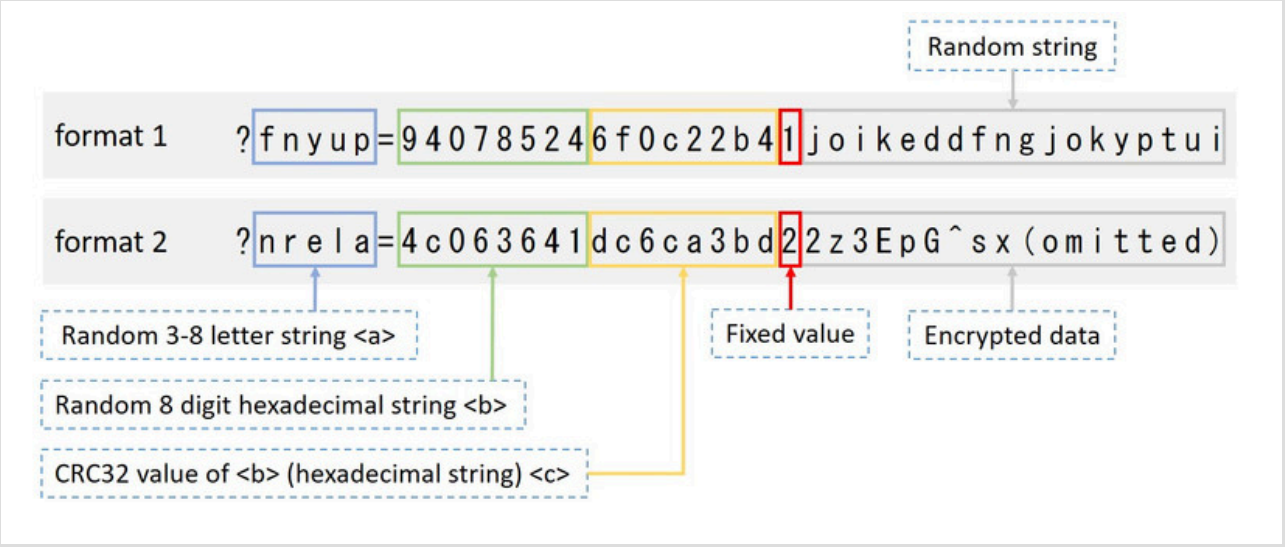
- Obtain host names, OS versions etc.
- Obtain drive information
- Configure communication intervals
- Sleep for a set period of time
- Execute a program
- Operate on files (Obtain file lists, download, upload, delete)
- Execute shell commands

After executing these functions, Datper sends the results to a C&C server.

How to detect Datper's communication

Datper sends HTTP GET requests with two types of query strings as in format 1 and 2 in the following figure.

Figure 1: Query string formats



As in the Figure 1, <a>, <b> and <c> in the query strings vary for each communication. If the fixed value which comes after <c> is "1" (as in format 1 in the Figure), it represents a request for commands, while those with "2" (format 2 in the Figure) are sent when sending command execution result to a C&C server. Command execution results are contained in the encrypted data. When the encrypted data is larger than 1024 bytes, POST method is used instead of GET.

Strings as in the above Figure is typical for Datper's communication and barely observed during usual web browsing. Based on the characteristics, it is possible to detect Datper's communication by checking for logs that match the format - that strings are aligned in the order of <a>=<b><c> format and <b>'s CRC32 value matches <c>. For easy verification, the following is an example of Python script for checking proxy server logs. Regular expressions need to be modified according to the log format.

```
import re
import sys
from binascii import crc32
from ctypes import c_uint

filter_1 = re.compile('(http://[\da-z\.-]+\.[a-z\.]{2,6}/[\/\w_\.-]+\?[\da-z]{3,8}=([\da-f]{8})([\da-f]{8})[1-2]

def main():
    for line in sys.stdin:
        m1 = filter_1.search(line)
        if m1:
            url = m1.group(1).lower()
            d1 =  m1.group(2).lower()
            d2 =  m1.group(3).lower()
        else:
            continue
        d1_crc32 = "%08x" % c_uint(crc32(d1)).value
        if d1_crc32 == d2:
            print "hit: %s" % line
if __name__ == '__main__':
    main()
```

Change in compression algorithm

As mentioned above, Datper's communication contains encrypted data. More precisely, plain text data is compressed, encrypted and then encoded. As for the compression algorithm, LZNT1 had been used, however, it was replaced with LZRW1/KH around November 2016. Below is the list of compression and encryption methods that Datper uses.

Table 1: List of compression and encryption methods

	Compression algorithm	**Encryption algorithm**	**Encode algorithm**
Datper (Until October 2016)	LZNT1	RC4	Base64 (alternative table)
Datper (After November 2016)	LZRW1/KH	xor + RC4	Base64 (alternative table)

The adversary has often used LZNT1 for attacks using Datper and other types of malware (xxmm/Minzen). While LZNT1 is easy to use with a Windows API "RtlDecompressBuffer", LZRW1/KH is not covered in Windows API. The reason for this inconvenient choice is unclear, however, this change together with the slight update in the encryption algorithm may be due to the intention of the adversary to disturb the malware analysis processes.

Conclusion

The adversary using Datper had conducted targeted attacks using Daserf malware for a long period of time against Japanese organisations. Activity with Datper is also likely to continue for a while, and we will carefully watch the malware and its attack activity.

- Yu Nakamura

(Translated by Yukako Uchida)

References

[1] CYBER GRID VIEW Vol.2 | Security Information | LAC Co. Ltd. (Japanese)

http://www.lac.co.jp/security/report/pdf/20160802_cgview_vol2_a001t.pdf

[2] A whole picture of cyber attacks targeting Japanese companies – BRONZE BUTLER (Japanese)

https://www.secureworks.jp/%7E/media/Files/JP/Reports/SecureWorksBronzeButlerReport.ashx

[3] “Tick” Group Continues Attacks

https://researchcenter.paloaltonetworks.com/2017/07/unit42-tick-group-continues-attacks/

[4] Tick cyberespionage group zeros in on Japan

https://www.symantec.com/connect/blogs/tick-cyberespionage-group-zeros-japan

[5] Old Malware Tricks To Bypass Detection in the Age of Big Data – Securelist

https://securelist.com/blog/research/78010/old-malware-tricks-to-bypass-detection-in-the-age-of-big-data/

[6] ShadowWali: New variant of the xxmm family of backdoors | Cybereason

https://www.cybereason.com/labs-blog/labs-shadowwali-new-variant-of-the-xxmm-family-of-backdoors

Appendix A SHA-256 Hash value of Datper Samples

- Datper(LZNT1)

efa68fcbd455a72276062fb513b71547ea11fedf4db10a476cc6c9a2fa4f67f7

12d9b4ec7f8ae42c67a6fd030efb027137dbe29e63f6f669eb932d0299fbe82f

331ac0965b50958db49b7794cc819b2945d7b5e5e919c185d83e997e205f107b

90ac1fb148ded4f46949a5fea4cd8c65d4ea9585046d66459328a5866f8198b2

2384e8ad8eee6db1e69b3ee7b6b3d01ae09f99a86901a0a87fb2788c1115090c

7d70d659c421b50604ce3e0a1bf423ab7e54b9df361360933bac3bb852a31849

- Datper(LZRW1/KH)

7bc042b9a599e1024a668b9921e2a42a02545429cf446d5b3d21f20185afa6ce

1e511c32cdf8abe23d8ba7c39da5ce7fc6c87fdb551c9fc3265ee22ac4076e27

2f6745ccebf8e1d9e3e5284a895206bbb4347cf7daa2371652423aa9b94dfd3d

---