Cmstar Downloader: Lurid and Enfal's New Cousin

By Robert Falcone

Published: 2015-05-18 · Archived: 2026-04-05 21:52:47 UTC

In recent weeks, Unit 42 has been analyzing delivery documents used in spear-phishing attacks that drop a custom downloader used in cyber espionage attacks. This specific downloader, Cmstar, is associated with the Lurid downloader also known as ‘Enfal’. Cmstar was named for the log message ‘CM**’ used by the downloader.

Unit 42 is aware of threat actors using two toolkits - MNKit and the Tran Duy Linh toolkit - to produce malicious documents that exploit CVE-2012-0158 in order to implant Cmstar. The Cmstar downloader itself has several unique and interesting features, as well as substantial infrastructure overlap with other tools worth discussing.

Manual Building of Import Address Table

The Cmstar downloader starts by manually building its import address table (IAT), much like shellcode would; however, it uses a rather unique technique. Instead of finding API function names based on their hashed values, this malware enumerates libraries' export address table (EAT) and searches for the name of the API function the payload needs to load by using a character to offset array. The payload pairs several comma-separated lists of characters with comma-separated lists of numbers. Each list of characters consists of the set found within the API function name the payload seeks to add to its IAT, while the corresponding list of numbers specifies the offset in the function name where those characters should be placed. For example, if the payload has "D,e,A" paired with "0,5,19", this results in the following mapping:

- "D" at offset 0 in API function name
- "e" at offset 5 in API function name
- "A" at offset 19 in API function name

The payload loads a specific Windows library's EAT by calling the ImageDirectoryEntryToData API function using the IMAGE_DIRECTORY_ENTRY_EXPORT flag. It then enumerates the library's EAT to find exported function names by checking each function name for the character and the specific offset. Once found, the payload adds the address for the specific API function to its IAT. For instance, the payload checks the EAT of "wininet.dll" using the comparisons mentioned above to find the address to the "DeleteUrlCacheEntryA" API function. One specific Cmstar payload that we analyzed used the character/offsets seen in Figure 1 to locate the API functions within three different Windows libraries to build its IAT.

Library	Characters	Offsets	Function Name
wininet.dll	D,e,A	0,5,19	DeleteUrlCacheEntryA
	e,O,A	3,8,12	InternetOpenA
	e,C,A	3,8,15	InternetConnectA

	p,O,A	3,4,15	HttpOpenRequestA
	p,S,A	3,4,15	HttpSendRequestA
	p,E,A	3,4,14	HttpEndRequestA
	p,Q,A	3,4,13	HttpQueryInfoA
	e,R,e	3,8,15	InternetReadFile
	e,C,e	3,8,18	InternetCloseHandle
advapi32.dll	S,V,A	3,6,13	RegSetValueExA
	C,s,y	3,6,10	RegCloseKey
	O,K,A	3,7,12	RegOpenKeyExA
	D,K,A	3,9,12	RegDeleteKeyA
	D,V,A	3,9,14	RegDeleteValueA
	U,r,A	3,6,11	GetUserNameA
	v,t,S	3,6,12	ConvertSidToStringSidA
	k,A,A	3,6,17	LookupAccountNameA
kernel32.dll	W,E,c	0,3,6	WinExec
	C,M,A	0,10,17	CreateFileMappingA
	U,V,e	0,5,14	UnmapViewOfFile
	M,O,e	0,7,12	MapViewOfFile

Figure 1. Character and Offset Pairs Found in Cmstar Payload and the Resulting API Function Names

Cmstar Behavior

After manually creating the IAT, Cmstar decrypts its configuration, several encrypted strings, and a piece of shellcode. The embedded configuration contains nothing more than a URL that Cmstar uses as its command and control (C2) location. The encrypted strings within the Trojan include fields used within the HTTP requests that Cmstar will create to communicate with its C2 server, as well as additional strings used to interact with the registry. The Cmstar sample associated with the MNKIT delivery document creates the following registry key to automatically execute at system startup:

- HKCU\Software\Microsoft\Windows\CurrentVersion\Run\xpsfiltsvcs: "rundll32.exe
- C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\xpsfiltsvcs.dll,XpsRegisterServer"

Unit 42 found an additional encrypted registry key that would allow Cmstar to automatically start up after reboots. However, the code does not decrypt, reference, or use the following registry key in any way, which suggests that the malware author left this artifact in the code after swapping to the run key listed above:

- HKCU\Software\Microsoft\CTF\LangBarAddIn

Cmstar also decrypts a 752-byte piece of shellcode that carries out communications with the C2 server, specifically by sending HTTP POST requests to the following URL:

- http://happy.launchtrue[.]com:8080/cgl-bin/update.cgi

It should be noted that the C2 URL contains the string ‘cgl-bin’, which visually resembles the common cgi-bin folder used by many webservers to run server-side scripts. Unit 42 used the Palo Alto Networks AutoFocus threat intelligence service to locate additional samples using the ‘cgl-bin’ string within URLs of HTTP requests and found several samples of the Cmwhite tool associated with the LURID/Enfal downloader[1], as seen in Figure 2.

Figure 2. Cmwhite Tools Using "cgl-bin" within HTTP Requests

Cmstar’s HTTP POST requests sent to ‘happy.launchtrue[.]com’ contain data that the Trojan gathers from the infected machine that has the following structure:

*<Windows Version number>@@<CPU Architecture (2 for x64, 1 for x86)>??<boolean for elevated privileges>]]**<boolean if antivirus processes are found>!!<static version string>==*

In one example, Unit 42 observed the following data within an analysis environment, which was then encrypted using a single-byte XOR algorithm and a key of 0x45 before being sent to the C2 server:

*510@@1??1]]**0!!150316o==*

Helpfully, the malware author writes log messages to the 'DF64159.TMP' file, used for debugging purposes throughout the execution of the Cmstar downloader. The log messages are abbreviated strings that describe specific activities during the execution of the code. For instance, the downloader uses the CreateMutex to create a mutex named '{53A4988C-F91F-4054-9076-220AC5EC03F3}' to determine if another instance of the code is running. If the downloader determines another instance of itself is running, the code writes the string 'CM**' - which happens to be the basis for the name of the Trojan - to the log file. Unit 42 created a Yara signature to detect Cmstar samples based on these debugging strings, which is available in the appendix.

Hashing Process Names

As mentioned in the behavioral analysis section, the Cmstar downloader gathers system-specific information to send to the C2 server. One such piece of information is the existence of specific running processes. Many malware families and tools check for the existence of antivirus, but the Cmstar tool does so in a clever way. Rather than including a list of strings of associated processes, Cmstar enumerates the running processes and subjects these process names to a hashing algorithm. The results of this algorithm are then compared against three static values: 0x1E00AFA, 0xBEE091E8 and 0xD46FCDFA. Unit 42 reverse engineered the algorithm and created the function seen below to generate hashes in order to determine the processes Cmstar is trying to find:

```
1  def hashStr(st):
2  hash = 0
3  count = 0
4  while count < len(st):
5  h1 = (hash<<0x13)&0xFFFFFFFF
6  h2 = (hash>>0x0d)&0xFFFFFFFF
7  h3 = (h1|h2)&0xFFFFFFFF
8  hash = (h3 + (ord(st[count])^0x4a))&0xFFFFFFFF
9  count += 1
10 return hash
11
12
13
14
```

15	
16	
17	
18	
19	
20	
21	

Unit 42 found that the string ‘avp’ subjected to the algorithm above results in the value 0x1E00AFA, which suggests the Cmstar sample specifically looks for Kaspersky's Anti-virus product (avp.exe) running on the compromised system. If the Trojan finds processes whose hash matches the three values mentioned earlier, it sets a boolean value (the character ‘1’) within the data sent to the C2 server and continues carrying out its functionality. Rather than altering its activities, Cmstar only notifies the C2 server if a system is running one of these processes, suggesting that the threat actors might employ this technique as a filtering mechanism to ignore analysis systems and researchers.

Threat Infrastructure

In order to determine the intrusion set involved with the Cmstar, Unit 42 enumerated infrastructure used by the downloader for its C2 servers. The related infrastructure chart in Figure 3 shows a rather large cluster of related entities with one small set of entities that do not share any related entities with the larger cluster.

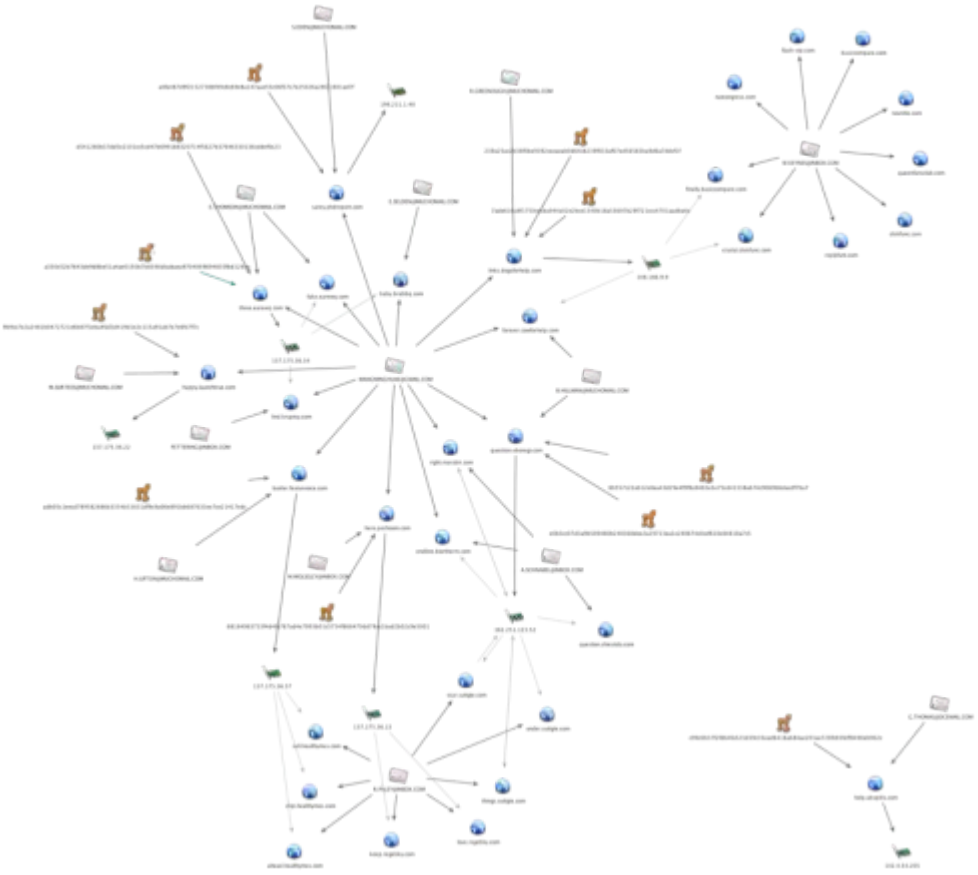

Figure 3. Infrastructure and Entites Related to Cmstar

As seen in the chart above, the C2 domain ‘happy.launchtrue[.]com’ was originally registered using the email address ‘WANGMINGHUA6@GMAIL[.]COM’. When Unit 42 used the Palo Alto Networks AutoFocus threat intelligence service to locate additional Cmstar samples, we found several with C2 domains that also had the same original registrant:

- links.dogsforhelp[.]com
- three.earewq[.]com
- question.eboregi[.]com
- here.pechooin[.]com
- sarey.phdreport[.]com
- bakler.featurvoice[.]com

The only known Cmstar C2 domain not initially registered by the email address was help.ubxpi0s[.]com. Further analysis revealed that additional domains related to Cmstar C2 domains were also originally registered using the email address ‘WANGMINGHUA6@GMAIL[.]COM’ and updated to the current information within a few days. In addition, this was the original registrant for C2 domain used in our Google Code blog[2], indicating this registrant email is likely a re-seller, and/or someone who initially sets up infrastructure for particular APT threat actors.

- forever.cowforhelp[.]com
- question.shiesiido[.]com

- endline.biortherm[.]com
- right.marubir[.]com
- baby.brabbq[.]com
- lind.kruptcy[.]com

The rest of the domains related to the Cmstar infrastructure did not use the original registrant noted above, but instead kept the same information initially used to register them. The difference in domain registration patterns could indicate threat actor preference, or could indicate there are at least two groups using this malware whose infrastructure at times overlaps.

- under.suttgte[.]com
- help.ubxpi0s[.]com
- finally.basiccompare[.]com
- crystal.diskfunc[.]com
- queenfansclub[.]com
- novnitie[.]com
- flash-vip[.]com
- replyfunt[.]com
- natcongress[.]com
- keep.regebky[.]com
- love.regebky[.]com

Interestingly, the updated registrant information (or original, in the cases where it wasn't changed) for all of the C2 domains in this blog has also been used to register scam sites, most purporting to sell knock-off designer products like shoes, software, or cell phones. The contact emails and contact names can vary, but the address is re-used. Blue Coat noted this pattern as well in a blog published late last year, which also noted the 'WANGMINGHUA6@GMAIL[.]COM' registrant email.[3] It is not known whether the threat actors conducting the malicious activity are also behind the scam sites.

Conclusion

The Cmstar tool has several interesting features, including a previously unseen method of manually creating its import address table using an API function name character to offset mapping techniques, and a hashing algorithm used to find antivirus processes on an infected system. Both of these features are noteworthy and may provide the ability to correlate future tools to the same group and/or malware authors.

The URL used by Cmstar to communicate with its C2 server, as well as significant infrastructure overlap, show a direct relationship between the Cmstar downloader, Lurid/Enfal and Cmwhite tools. In a majority of the cases, threat actors using the Cmstar downloader initially register the C2 domains using the email address 'WANGMINGHUA6@GMAIL[.]COM' and later change the registration information to include a different email address. Unit 42 cannot positively confirm that the threat actors control the 'WANGMINGHUA6@GMAIL[.]COM' email address, or if the email address belongs to a reseller that the threat actors buy domains from to create their infrastructure; however, we do believe this is an interesting TTP worth tracking in future infrastructure enumeration.

1. THE “LURID”DOWNLOADER - Nart Villeneuve and David Sancho - http://la.trendmicro.com/media/misc/lurid-downloader-enfal-report-en.pdf
2. Attacks on East Asia using Google Code for Command and Control – Jen Miller-Osborn and Rob Downs -- http://blog.paloaltonetworks.com/2014/08/attacks-east-asia-using-google-code-command-control/
3. Linking APTs from 2011 and 2014 to an Active Scam Network – Kiel Wadner - https://www.bluecoat.com/security-blog/2014-10-08/linking-apts-2011-and-2014-active-scam-network

Appendix

Known Cmstar Downloaders

Filename: xpsfiltsvcs.tmp
SHA256: 239a25ac2b38f0be9392ceeaeab0d64cb239f033af07ed56565ba9d6a7ddcf1f
C2: links.dogsforhelp.com

Filename: xpsfiltsvcs.tmp
SHA256: 6b557c22ab12e8ea43d29e4f9f8a9483e3e75cd41338a674c9069b6dacdf7ba7
C2: question.eboregi.com

Filename: xpsfiltsvcs.tmp
SHA256: 7ade616a8f1750cecba944a02e2bce1340b18a55697b29f721ccc4701aadba6e
C2: links.dogsforhelp.com

Filename: xpsfiltsvcs.tmp
SHA256: 88184983733f4d4fa767ad4e7993b01c5754f868470dd78ac1bad2b02c9e5001
C2: here.pechooin.com

Filename: xpsfiltsvcs.tmp
SHA256: b9d597aea53023727d8564e47e903b652f5e98a2c32bdc23bc4936448fb2d593
C2: question.eboregi.com

Filename: xpsfiltsvcs.tmp
SHA256: e0b3cc07d3a9b509480b240368dee2a29713ea1e240674c0ccf610c84810a7c5
C2: question.eboregi.com

Filename: xpsfiltsvcs.txt
SHA256: f4b8f71c0e10a345a855763e01033e2144e949c8f98c271755cc025e3f55b7da
C2: three.earewq.com

Filename: xpsfiltsvcs.tmp
SHA256: 2e00a98212c5a2015d12612f0d26039a0c2dfee3e1b384675f613e683f276e02
C2: bakler.featurvoice.com

Filename: xpsfiltsvcs.dll
SHA256: 42ed2edc37b957266ff7b02955a007dd82d955c09ef7be23e685d938e40ad61d
C2: turber.xoxcobbs.com

Filename: xpsfiltsvcs.dll
SHA256: 9b9cc7e2a2481b0472721e6b87f1eba4faf2d419d1e2c115a91ab7e7e6fc7f7c
C2: happy.launchtrue.com

Filename: xpsfiltsvcs.tmp
SHA256: a330c52b7643de9d8be51a4ae0150b7b8390dbabaea9704069694835fbd3298e
C2: three.earewq.com

Filename: xpsfiltsvcs.tmp
SHA256: a8fa487d9f2152738bf49c8c69e8a147aae55c06f37c7e25026a28f21601ad7f
C2: sarey.phdreport.com

Filename: xpsfiltsvcs.tmp
SHA256: c99c0b37f2fd64fa523d39c35ead6416a684ae203ae728feb5feff8490eb902c
C2: help.ubxpi0s.com

Filename: xpsfiltsvcs.tmp
SHA256: d541280b37dd5e2101cc5cd47b0991b8320714f5627b37646330136cddef0c23
C2: three.earewq.com

Filename: coyote_load.dll
SHA256: adb05c1eecd789582886b3354b53831df9c9a06e891bb687633ee7ce21417edc
C2: bakler.featurvoice.com

Delivery Documents Installing Cmstar

Filename: Какая реформа армии нужна Украине.doc (What is needed reform of the army Ukraine.doc)
MD5: 76ffb9c2d8d0ae46e8ea792ffacc8018
SHA256: c26c67eac20614038aaadfda19b604862926433333893d65332928b5e36796aa
Type: MIME entity text
Toolkit: MNKit
Author: User123
Last Saved By: User123
Created: 2012-05-01T14:08:00Z
Modified: 2012-05-01T14:12:00Z

Filename: запуск ракеты-носителя Союз.doc (launch of the carrier rocket Soyuz.doc)
MD5: 6fdeadacfe1dafd2293ce5c4e178b668
SHA256: e39b0e777ef0135c1f737b67988df70c2e6303c3d2b01d3cdea3efc1d03d9ad9
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: РФ_КНР_сельское_хозяйство.doc (RF China Agriculture and Economy.doc)
MD5: 9da10a36daf845367e0fc2f3e7e54336
SHA256: a0aeb172a72442d2c2c02e1d32b48accb9975c4da7742df24d9350a8ccd401f2
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: Ерөнхий сайд асан Н.Алтанхуягийг шалгаж эхэлжээ.doc (Former Prime Minister started to check with their lethargy.doc)
MD5: f7d47e1de4f5f4ad530bca0fc080ea53
SHA256: 4883286b8229a2c43db17eb1e1c5bd79d1933e840cdfedff80d5b99a84c9e39f
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: запуск ракеты-носителя Союз.doc (launch of the carrier rocket Soyuz.doc)
MD5: 6fdeadacfe1dafd2293ce5c4e178b668
SHA256: e39b0e777ef0135c1f737b67988df70c2e6303c3d2b01d3cdea3efc1d03d9ad9
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

MD5: 5aeb8a5aa8f6e2408016cbd13b3dfaf0
SHA256: df34aa9c8021f1f0bdf33249908efc4a9628941453ad79b281b3a46bf9a7f37f
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: Путины урилга.doc (Putin’s invitation.doc)
SHA256: 45027d11ab783993c413f97e8e29759d04b04564f8916f005f5c632f291697bb
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

MD5: 46bf922d9ae07a9bc3667a374605bdbb
SHA256: 7dc78caf515d1d3d2b84be7c023ccbd0b4fd670a42babcbcbd5a5ba65bbdd166
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: Армия-2015.doc (Army-2015.doc)
MD5: 783a423f5e285269126d0d98f53c795b
SHA256: 5b338decffe665a2141d1079c32b2d612057d1fdbfddf198cc28003dae7f0516
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: С днём 70 лет победы.doc (Happy 70 years of victory.doc)
MD5: 510b3272342765743a202373261c08da
SHA256: 0a10d7bb317dceccd05d18408fd6b8b12c784910e5f7e035ee22c2c5d7e4cbf5
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: new resume.doc
MD5: c5ae7bd6aec1e01aa53edcf41962ac04
SHA256: 87bcc6d18c6a81d92d826b232703dee84b522bd1d0cae56f74bcf58fdca0930e
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

MD5: 3d41e3c902502c8b0ea30f5947307d56
SHA256: b65dd4da9f83c11fcb5beaec43fabd0df0f7cb61de94d874f969ca926e085515
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation

Filename: Центр-2015.doc (Center-2015.doc)
MD5: 94499ff857451ab7ef8823bf067189e7
SHA256: 671dfc4d47a43cf0bd9205a0f654dcd5050175aef54b69388b0c5f4610896c6a
Type: Microsoft Office Word 97-2003 Document
Toolkit: Tran Duy Linh
Created: 2012:11:23 04:35:00
Modified: 2012:11:23 04:39:00
Company: DLC Corporation
Related Cmwhite Tools

MD5: 3fff0bf6847d0d056636caef9c3056c3
SHA256: 13c1d7eb2fd64591e224dec9534d8252f4b91e425e8f047b36605138d15cbf2d
C2: stone.timmf.com

MD5: 30a6c3c7723fe14c4b6960fa3e4e57ba
SHA256: ab934c6177be0fdc3b6dfbf21f60ce7837a30e6599dcfb111b43008c75ceb91f
C2: xphome.mailru-vip.com
C2: error.yandex-pro.com

MD5: e0417547ba54b58bb2c8f795bca0345c
SHA256: 1cf44815f9eb735e095f68c929d5549e0ebc44af9988cccaf1852baeb96bb386
C2: dns.thinkttun.com

MD5: d05f012c9c1a7fb669a07070be821072
SHA256: a37f337d0bc3cebede2039b0a3bd5afd0624e181d2dcc9614d2f7d816b5a7a6b
C2: help.redhag.com
C2: mssage.hotoicq.com
C2: new.hoticq.com

Cmstar Yara Rule

```
1  rule ce_enfal_cmstar_debug_msg
2  {
3  meta:
4  author = "rfalcone"
5  description = "Detects the static debug strings within CMSTAR"
6  reference = "9b9cc7e2a2481b0472721e6b87f1eba4faf2d419d1e2c115a91ab7e7e6fc7f7c"
7  date = "5/10/2015"
8  strings:
```

```
$d1 = "EEE\x0d\x0a" fullword
$d2 = "TKE\x0d\x0a" fullword
$d3 = "VPE\x0d\x0a" fullword
$d4 = "VPS\x0d\x0a" fullword
$d5 = "WFSE\x0d\x0a" fullword
$d6 = "WFSS\x0d\x0a" fullword
$d7 = "CM**\x0d\x0a" fullword
condition:
uint16(0) == 0x5a4d and all of ($d*)
}
```

35	

Source: https://researchcenter.paloaltonetworks.com/2015/05/cmstar-downloader-lurid-and-enfals-new-cousin/