"Cracking Open the Malware Piñata" Series: Intro to Dynamic Analysis with RedLineStealer

atomicmatryoshka.com/post/cracking-open-the-malware-piñata-series-intro-to-dynamic-analysis-with-redlinestealer

z3r0day_504

January 2, 2022

- <u>z3r0day_504</u>
- •
- ∘ Jan 2
- 0
- 6 min read



Dynamic analysis involves running a binary and observing its behavior in a controlled environment. This can be of significant benefit because some capabilities of malware come to life only at runtime, meaning that the indicators and behaviors would not be observed if only analyzed statically.

In this iteration of my latest series, we dive into RedLineStealer. At the time of writing this blog post, MalwareBazaar shows RedLineStealer as the second most prevalent malware family in the last 14 days. The variant we're analyzing is relatively noisy, as you'll see in the content below.

If you're interested in viewing my previous posts in this series, please check them out here:

"Cracking Open the Malware Piñata" Series: Analysis Environment Setup

"Cracking Open the Malware Piñata" Series: Intro to Static Analysis with Kazy Trojan

Additionally, if you'd like to get right to the counter-hacking, scroll down to the "Pushing the Big Red Button" headling. :)

BEFORE THE BOOM

Prior to double-clicking on that badness-laden executable, there are a couple of things that need to be done to prepare the environment.

Network Configuration

First and foremost, we need to check the networking between virtual machines. Verify that the network adapters are set to host only. From there, go into REMnux and identify its IP.

Making note of it, we return to the Windows VM and take the following steps:

- 1. Access the network adapter settings in Windows
- 2. Change the Windows IP to be on the same subnet as the REMnux IP
- 3. Input the REMnux IP as the default gateway and preferred DNS server in the Windows settings
- 4. Close all dialog boxes
- 5. Ping the REMnux VM from the Windows VM to make sure the connection is functional

This ensures that any time the Windows VM attempts to establish a network connection, the connection will be routed to the REMnux VM. This will be useful when it comes time to observing what the malware calls out to when detonating.

Registry Snapshot

Next, let's grab a snapshot of the registry. RegShot is a tool originally developed in 1999 by TiANWEi but contributed to by Maddes, XhmikosR, tulipfan, and Belogorokhov Youri amongst a large cohort of others over the course of two decades. The RegShot utility enables to user to grab "snapshots" of the registry prior to and after a specific event. The utility also has a "compare" feature, which highlights the differences between the "before" and "after" shots.



To grab our "before" shot, it's as simple as:

- 1. Double-click on the RegShot executable
- 2. Click on "1st Shot," then "Shot and Save"
- 3. Once complete, name your file and where to save it

VM Snapshot

After completing the registry baseline, it's a good time to take a snapshot of the VM. This will allow you to revert the Windows VM to this state after detonating the malware. I recommend taking the VM snapshot after taking the initial registry snapshot to avoid having to repeat the process again.

To take a VM snapshot in VMWare:

- 1. Go to the toolbar at the top of the window and click on "VM"
- 2. Hover over "Snapshot"
- 3. Click "Take Snapshot..."
- 4. Name it something intuitive and click "Take Snapshot"

PREPARE THE WATCH

The environment is now configured and baselines have been captured. At this stage, we start spinning up the tools that will actively be running when the malware is executed. It's important to have these tools active prior to initiating detonation so that we capture *all* events that take place.

Process Hacker

Process Hacker is one of the first tools I spin up at this point. Developed by Wen Jia Liu with help from XhimkosR and a plethora of additional contributors, it is a "task manager on steroids." The GUI is intuitive with highlighting showing which processes are spawning and terminating, modifiable columns, and additional details visible with just the hover of a mouse.

🕽 Refresh 🛞 Options 🛛 🏙 Find H	andles or	DLLs >	📲 System inf	ormation	🗆 🖬 🗙	Search Processes (Ctrl+K)
rocesses Services Network Dak						
Name	PID	CPU	I/O total	Private b	User name	Description
System Idle Process	0	83.22		60 kB	NT AUTHORITY/SYSTEM	
V System	4	1.34	4 kB/s	40 kB	NT AUTHORITY/SYSTEM	NT Kernel & System
I smss.exe	364			1.04 MB		Windows Session Manager
Memory Compression	1712			732 kB		
Interrupts		1.22		0		Interrupts and DPCs
Secure System	56			180 kB		
Registry	108			6.36 MB		
CSrss.exe	496			1.78 MB		Client Server Runtime Process
🖌 💽 wininit.ece	596			1.28 MB		Windows Start-Up Application
 services.exe 	660	0.26		5.31 MB		Services and Controller app
✓ ■ sychost.exe	852	0.03		11.59 MB		Host Process for Windows Ser
illij WmiPrvSE.exe	3808			9.59 MB		WMI Provider Host
 SearchHost.exe 	6088			119.88 MB	WINDEV2112EVAL\User	
StartMenuExperie	3436			19.31 MB	WINDEV2112EVAL\User	
RuntimeBroker.exe	5548			14.55 MB	WINDEV2112EVAL\User	Runtime Broker
RuntimeBroker.exe	6184			5.4 MB	WINDEV2112EVAL\User	Runtime Broker
dlhost.exe	6448			6.16 MB	WINDEV2112EVAL\User	COM Surrogate
dlhost.exe	740			2.43 MB	WINDEV2112EVAL\User	COM Surrogate
 RuntimeBroker.exe 	4708			2.07 MB	WINDEV2112EVAL\User	Runtime Broker
MiniSearchHost.e	2380			15.98 MB	WINDEV2112EVAL\User	
ApplicationFrame	976			9.08 MB	WINDEV2112EVAL\User	Application Frame Host
✓ III Widgets-exe	8216				DEV2112EVAL\User	

Fiddler Classic

Fiddler allows you to observe and inspect HTTP/S network requests. Developed by Telerik, this tool is one of the several tools available to observe the networking capabilities of the malware once detonated.



FakeDNS

Possible overkill, but I normally have this *and* Fiddler running concurrently. FakeDNS is a command-line tool available on REMnux and developed by Verisign's iDefense group. It captures all DNS requests sent to the device its running on and lists them in the terminal in real-time.

remnux@remnux: ~/malware ×	remnux@remnux: ~
remnu@remnux:-/malwareakedns[INF0]:Response:dakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:ipv6.msftconnecttest.comobserveddakedns[INF0]:Response:fakedns[INF0]:Response:fakedns[INF0]:Response:fakedns[INF0]:Response:glive.comobservedfakedns[INF0]:Response:dakedns[INF0]:Response:dakedns[INF0]:Response:dakedns[INF0]:Response:dakedns[INF0]:Response:dakedns[INF0]:Response:dakedns[INF0]:	remnux@remnux: ~
<pre>Takedns[INF0]: Response: www.msftconnecttest.com -> 192.168.22.128 Takedns[INF0]: Response: ipv6.msftconnecttest.com -> 192.168.22.128</pre>	

WireShark

While I won't be using it in this specific scenario, Wireshark is an incredibly useful packet capture tool. Packet inspection is a very useful technique, especially when characterizing communications between a piece of malware and the infrastructure it's "calling out" to. Dissecting this information can better help paint the picture of what information the malware is sending or receiving.

Process Monitor

Developed by Mark Russinovich as part of Windows Sysinternals, Process Monitor captures file, process, registry, and networking events all in real-time. In dynamic analysis, this is magnificent in that in catches a lot of artifacts that may no longer be present by the time an analyst goes digging for evidence. It is also very effective at painting a timeline of events.

🜉 Process Monitor - Sysi	nternals: www.sysinter	hals.com			-	×
File Edit Event Filte	r Tools Options	Help				
6 🖯 🗔 🔂 🚺	1 7 🛛 🎯	퉒 🗲 🔎 利 🖪	i 🗈 🖵 📽 🖊			
Time Process Name	PID Operation	Path	Result	Detail		
8:09:0 The MsMpEnglexe	3060 ReadFile	C:\ProgramData\Microsoft\W	Indows DeSUCCESS	Offset: 14,770,176,		
8:09:0 Trychost.exe	2828 ReadFile	C:\Windows\System32\State	RepositorySUCCESS	Offset: 704,512, Le		
8:09:0 := ctfmon.exe	4944 💽 ReadFile	C:\Windows\System32\Core	UlCompon SUCCESS	Offset: 2,798,080,		
8:09:0 Terretaria	4944 🎬 RegQueryKe	y HKLM	SUCCESS	Query: Handle Tag		
8:09:0 := ctfmon.exe	4944 🎬 RegOpenKe	HKLM\Software\Microsoft\In	put\LocaleSUCCESS	Desired Access: R		
8:09:0 : ctfmon.exe	4944 🎛 RegQueryVa	ue HKLM\SOFTWARE\Microsof	ft\Input\LoSUCCESS	Type: REG_DWO		
8:09:0 Transferrer 8:09:0	4944 🎬 RegCloseKe	HKLM\SOFTWARE\Microsof	ft\Input\LoSUCCESS			
8:09:0 : ctfmon.exe	4944 🖄 Query Namel	foC:\Program Files (x86)\Proces	ss Monitor\SUCCESS	Name: \Program Fil		
8:09:0 Transferrer 8:09:0	4944 🏨 RegQueryKe	y HKCU	SUCCESS	Query: Handle Tag		
8:09:0 : ctfmon.exe	4944 🎬 RegOpenKe	HKCU\SOFTWARE\Microsof	ft\Input\SeSUCCESS	Desired Access: R		
8:09:0 Terctfmon.exe	4944 🎬 RegQueryVa	ue HKCU\Software\Microsoft\In	put\Settin NAME NOT FO	UND Length: 16		
8:09:0 Ter ctfmon.exe	4944 🎬 RegCloseKe	HKCU\Software\Microsoft\In	put\Settings SUCCESS			
8:09:0 : ctfmon.exe	4944 🎬 RegQueryKe	y HKLM	SUCCESS	Query: Handle Tag		
8:09:0 Transferrer 8:09:0	4944 🎬 RegOpenKe	HKLM\SOFTWARE\Microsof	ft\Input\SeSUCCESS	Desired Access: R		
8:09:0 : ctfmon.exe	4944 🌃 RegQueryVa	ue HKLM\SOFTWARE\Microsof	ft/Input/SeSUCCESS	Type: REG_DWO		
8:09:0 Tothon.exe	4944 🎬 RegCloseKe	HKLM\SOFTWARE\Microsol	It\Input\SeSUCCESS			
8:09:0 = ctfmon.exe	4944 🎬 RegQueryKe	y HKCU	SUCCESS	Query: Handle Tag		_
Showing 30,767 of 47,981 ev	ents (64%)	Backed by virtual memory				

Now we're locked and loaded to let the badness run rampant. Here we'll start our analysis of RedLine Stealer with a sample I've pulled from MalwareBazaar. If you're not familiar, you can check them out <u>here</u>. IOCs identified through this process will be summarized at the bottom of the post.



Right out of the gate, here's the evolution of Process Hacker after initial execution:

💙 🜉 Procmon64.exe	7096	
🜉 Procmon64.exe	3692	
🕞 setup_x86_x64_install.exe	10184	
OneDrive.exe	3976	
🕓 msedge.exe	1280	
💽 msedge.exe	1088	
避 setup_installer.exe	5788	



setup_x86_x64_install.exe spawns setup_installer.exe, which in turn spawns several instances of Sat19d470e8e0597fc47.exe. Process Hacker allows you to take a look at the memory strings during runtime. This can be done by right-clicking on the process of interest, clicking "Properties," navigating to the "Memory" tab, and clicking on "Strings..." Below are screenshots of what this looked like for Sat19d470e8e0597fc47.exe.

Address	Length	Result	
0x-9000-9d	44	This program cannot be run in DOS mode. \$	
0x-90b0b8	10	v4.0.30319	
x4108ea	44	5#<&=3>CF/RIANERCaDbVc	
x41143e	156	WIX'Y'Z\$(\$\\$)%^^ E'Eatbeddeeftotheet8etemeneoEoEoEdesteuevewexe/eteXe	
x4115b5	28	StatcArrayIntTypeSze=10	
x411607	28	StaticArrayInitTypeSize=20	
0x411624	40	359A00EF6C789FD4C18644F56C5D3F97453FFF20	
0x411652	28	StatcArravInitTypeSize=30	
x41166f	28	StaticArrayInitTypeSize=40	
x41168c	40	77A9683FAF2EC9EC3DABC09D33C3BD04E8897D60	
0x4116b5	40	A8F9862160DF0858926D5ED70E280F6C95A25280	
0x4116de	40	F413CEA98AA458730567FE47F57CC3C94DDF63C0	
0x411710	15	<trvfind>b 0 0</trvfind>	
0x41172f	13	<.ctor>b 0 0	
0x41173d	20	<>c DisplayClass0 0	
0x411752	41	<getwindowsversion>g HRLM GetString[11.0</getwindowsversion>	
0x411785	11	<run>b 10</run>	
0x411791	29	<getdefaultipv4address>b 1.0</getdefaultipv4address>	
0x4117b8	12	<init>b 2.0</init>	
0x4117ce	20	<domainexists>b 4 0</domainexists>	
0x4117e3	20	<>c DisplayClass4 0	
0x411801	19	<getscanargs>b 5.0</getscanargs>	

Address	Length	Result
0x412e13	40	B14D74C51EAE4F88F8F39888D07DA392799FCAAF
0x412e3c	40	78F285852D43939E0FBD786C5592189AF986E88F
0x412e65	40	3DB6DAD76E13B54DC03AF1C6092C40388E57F8BF
0x412e98	34	BCRYPT_INIT_AUTH_MODE_INFO_VERSION
0x412ed4	14	TryInitNordVPN
0x412ee3	14	TryInitOpenVPN

Results - Sat192603358e4defd.exe (4800)

Address	Length	Result
0x2cccd84	74	http://tempuri.org/Entity/Id4Respo
0x2ccd678	52	http://tempuri.org/Entity/
0x2ccd704	44	http://tempuri.org/Ent
0x2ccd748	58	http://tempuri.org/Entity/Id5
0x2ccd7ac	52	http://tempuri.org/Entity/
0x2ccd85c	44	http://tempuri.org/Ent
0x2ccd8a0	74	http://tempuri.org/Entity/IdSRespo
0x2cce178	52	http://tempuri.org/Entity/
0x2cce204	44	http://tempuri.org/Ent
0x2cce248	58	http://tempuri.org/Entity/Id6
0x2cce2ac	52	http://tempuri.org/Entity/
0x2cce35c	44	http://tempuri.org/Ent
0x2cce3a0	74	http://tempuri.org/Entity/Id6Respo
0x2cced38	52	http://tempuri.org/Entity/
0x2ccedc4	44	http://tempuri.org/Ent
0x2ccee08	58	http://tempuri.org/Entity/Id7

We see references to "tempuri[.]org" as well as references to "TryInitNordVPN" and "TryInitOpenVPN." Based on cursory research, the last two references don't appear to be existing open-source libraries/APIs, so these could've been developed by the malware author.

Taking a look over in fakeDNS, we have quite a few attempts to contact domains, aside from general Windows noise:

fakedns[INF0]:	Response:	www.msftconnecttest.com -> 192.168.22.128
fakedns[INF0]:	Response:	www.msftconnecttest.com -> 192.168.22.128
fakedns[INF0]:	Response:	wdcp.microsoft.com -> 192.168.22.128
fakedns[INF0]:	Response:	wdcpalt.microsoft.com -> 192.168.22.128
fakedns[INF0]:	Response:	184.246.69.159.in-addr.arpa -> 192.168.22.128
fakedns(INF0):	Response:	45.30.193.212.in-addr.arpa -> 192.168.22.128
fakedns[INF01:	Response:	dns.msftncsi.com -> 192.168.22.128
fakedns[INF01:	Response:	dns.msftncsi.com -> 192.168.22.128
fakedns[INF0]:	Response:	dns.msftncsi.com -> 192.168.22.128
fakedns[INF01:	Response:	one-mature-tube.me -> 192.168.22.128
fakedns[INF0]:	Response:	cloudiah.com -> 192.168.22.128
fakedns[INF0]:	Response:	<pre>tlu.dl.delivery.mp.microsoft.com -> 192.168.22.128</pre>
fakedns[INF0]:	Response:	57.225.144.45.in-addr.arpa -> 192.168.22.128
fakedns[INF0]:	Response:	kelenxz.xvz -> 192.168.22.128
fakedns[INF0]:	Response:	ad-postback.biz -> 192.168.22.128
fakedns[INF0]:	Response:	www.listincode.com -> 192.168.22.128
fakedns[INF0]:	Response:	ip-api.com -> 192.168.22.128
fakedns[INF0]:	Response:	iplogger.org -> 192.168.22.128
fakedns[INF01:	Response:	gp.gamebuy768.com -> 192.168.22.128
fakedns[INF0]:	Response:	www.hhiuew33.com -> 192.168.22.128
fakedns[INF0]:	Response:	umwatson.events.data.microsoft.com -> 192.168.22.128
fakedns[INF0]:	Response:	168.69.108.65.in-addr.arpa -> 192.168.22.128
fakedns[INF0]:	Response:	teams,live.com -> 192.168.22.128

Over the course of time, the malware consistently attempted to call out to hhieuw33[.]com. We were able to take a closer look at that communication in Fiddler:

#	Result	Protocol	Host	URL	Body	Caching	Content-Type	Process	Comments
135	502	HTTP	Tunnel to	statics.teams.cdn.live.net	512	no-cac	text/html; c	msedg	
台 136	502	HTTP	Tunnel to	one-mature-tube.me:443	512	no-cac	text/html; c	sat194	
A 137	502	HTTP	www.hhiuew33.com	/check/?sid=08key=8e56	512	no-cac	text/html; c	sat194	
前 138	502	HTTP	Tunnel to	umwatson.events.data.mi	512	no-cac	text/html; c	wermgr	
139	502	HTTP	Tunnel to	iplogger.org:443	512	no-cac	text/html; c	sat19f	
<u>m</u> 140	502	HTTP	Tunnel to	one-mature-tube.me:443	512	no-cac	text/html; c	sat194	
A 141	502	HTTP	www.hhiuew33.com	/check/?sid=08key=8e56	512	no-cac	text/html; c	sat194	
142	502	HTTP	Tunnel to	slscr.update.microsoft.co	512	no-cac	text/html; c	sychos	
⁽¹⁾	502	HTTP	Tunnel to	one-mature-tube.me:443	512	no-cac	text/html; c	sat194	
A 144	502	HTTP	www.hhiuew33.com	/check/?sid=08key=8e56	512	no-cac	text/html; c	sat194	
尚 145	502	HTTP	Tunnel to	umwatson.events.data.mi	512	no-cac	text/html; c	wermgr	
146	502	HTTP	Tunnel to	one-mature-tube.me:443	512	no-cac	text/html; c	sat194	
A 147	502	HTTP	www.hhiuew33.com	/check/?sid=08key=8e56	512	no-cac	text/html; c	sat194	
148	502	HTTP	Tunnel to	umwatson.events.data.mi	512	no-cac	text/html; c	wermgr	
149	502	HTTP	Tunnel to	slscr.update.microsoft.co	512	no-cac	text/html; c	sychos	
150	-	HTTP	Tunnel to	iplogger.org:443	-1			sat19f	
151	-	HTTP	Tunnel to	one-mature-tube.me: 443	-1			sat194	
152	-	HTTP	www.hhiuew33.com	/check/?sid=08key=8 Micros	oft Edge			sat194	

\overline FiddlerSa	ipt	🗄 Log	Filters	Timeline
Statistics	🖳 Inspectors	🐐 AutoRespond	ier 📝 Composer	FO Fiddler Orchestra Beta
Headers TextView	SyntaxView V	WebForms HexView	Auth Cookies	Raw JSON XML
POST http://www. Proxy-Connection Content-Type: ap User-Agent: Mozi Content-Length: Host: www.hhiuew bHphvzv6ZEdGc23D	hhiuew33.com/d 1: Keep-Alive plication/x-ww 11a/5.0 (Windo 472 (33.com STZJQ013SW1320	:heck/?sid=0&key= ww-form-urlencode ws NT 10.0; Wind DltMwhZeuk2SUNJNV	8656becd9ed99edf57 d i4; x64) ApplewebKi /luvTBNekkxTwpFd016	<u>d41e1dd73118c5</u> HTTP/1.1 t/537.36 (KHTML, like Geck RxlaakvSTldRMlpqvxpNRE00Wm
Find (press Ctrl+En	ter to highlight all)			View in Notepad

In the URL, the malware is attempting to pass some information in the parameters named "sid" and "key." Additionally, the content of the POST request is encoded in multiple layers, with the outermost layer being base64.

Moving to the RegShot comparison, output:

Files added: 171 Files added: 171 C:\Program Files (x86)\Process Monitor\Logfile.CSV C:\Program Files (x86)\FarLabUhinstaller\unins000.dat C:\Program Files (x86)\FarLabUhinstaller\unins000.exe C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.@.cmd_edd8c564b2e33d8f5d03143c8eaf05d6e819754_e6c52f83_cab_872cedee-6936-4E C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash.Sat1983c7a137841_c7f3a667854f329fb45264ead98727b1ad860de_7acd47b4_cab_a0eaa C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash_Sat1983c7a137841_c7f3a667854f329fb45264ead98727b1ad860de_7acd47b4_cab_a0eaa C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash_Sat1983c7a137841_c7f3a667854f329fb45264ead98727b1ad860de_7acd47b4_cab_a0eaa C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash_Sat1983c7a137841_c7f3a667854f329fb45264ead98727b1ad860de_7acd47b4_cab_a0eaa C:\ProgramData\Microsoft\Windows\WER\ReportQueue\AppCrash_Sat1983c7a137841_c7f3a667854f329fb4

🗮 ~res-x64 - Notepad			;	×
File Edit Format View Help				
HKU\S-1-5-21-1516808570-3660347512-3657706960-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Ext	ensibl	le Ca	che	١.
Keys added: 50				
<pre>HRLM\SOFTWARE\Microsoft\SystemCertificates\AuthRoot\Certificates\82FAF3E291435468607857694DF5E45868851868 HRLM\SOFTWARE\Microsoft\Tracing\Sat1944466031ace9ca_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat19416446031ace9ca_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat194164460831ace9ca_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\Tracing\Sat1941c04426464e86_RASAP132 HRLM\SOFTWARE\Microsoft\TextorNet\Nindows Error Reporting\TermReason\18084 HRLM\SOFTWARE\Microsoft\Twindows\TextorNet Reporting\TermReason\4864 HRLM\SOFTWARE\Microsoft\Twindows\TextorNet Reporting\TermReason\4864 HRLM\SOFTWARE\Microsoft\Twindows\TextorNet Reporting\TermReason\9592 HRLM\SOFTWARE\Microsoft\Twindows\Terror Reporting\TermReason\9592 HRLM\SOFTWARE\Microsoft\TextorNetTYCR\ClockData HRLM\SOFTWARE\WM6432Nde\Microsoft\TextorTyCR\ClockData HRLM\SOFTWARE\WM6432Nde\Microsoft\TextorTyCR\ClockData HRLM\SOFTWARE\WM6432Nde\Microsoft\SistemCertificates\AuthRoot\Certificates\82FAF3E291435468687857694DF5E45868851868 HRLM\SYSTEM\ControlsEt801\Services\PROCM0X4 HRLM\SYSTEM\ControlsEt801\Services\PROCM0X4 HRLM\SYSTEM\ControlSEt801\Services\PROCM0X24\Instances</pre>				
Values added: 152 	00 80 15 86 1 5 04 0 79 89 1	1 01 8 C 21		

RegShot does an awesome job tracking everything related to the registry, and even goes as far as tracking file and folder events. We can see several that stand out in relation to the activity we've already seen, such as the keys which contain the malware executable names.

In a similar but more robust "all encompassing" ability, Process Monitor grabs a plethora of data as well:

Process Monitor - Sysintemals: www.sysintemals.com			
File Edit Event Filter Tools Options Help			
🗁 🔚 🖸 🛱 🖉 🖉 🎯 👗 🐓 오 기 📾 🕓 🖵 & 🗳			
Time Process Name PID Operation Path	Result	Detail	TID
6:37.0 Sat 196470e8e 4464 TCP Reconnect 192 168 22 133 57405 o 155 69 246 184 13127	SUCCESS	Length: 0, segnant	0
6:37.0., a Sat 19d470e8e., 4464 TCP Reconnect 192.168.22.133:57405-> 159.69.246.184:13127	SUCCESS	Length: 0. segnum:	0
6.37.1 TSat15d470e8e 4464 CTCP Reconvect 192.168.22.133.57405 -> 155.69.246.184.13127	SUCCESS	Length: 0, segnum	0
6.37.1 K Sat 191709/9xd 9988 CTCP Reconnect 192.168.22.133.57407 -> 212.193.30.45.80	SUCCESS	Length: 0. segnum	0
6:37.1 ESst19170999d 9988 CTCP Reconnect 192.168.22.133.57407 -> 212.193.30.45:00	SUCCESS	Length: 0, segnum:	0
6.37.1 ■ Sat 15d470e8e 4464 @ TCP Reconnect 192.168.22.133.57405 > 159.69.246.184.13127	SUCCESS	Length: 0, segnum	0
6:37.2 🛤 Sat 191709f3bd 9988 😨 TCP Reconnect 192.168.22.133.57407 -> 212.193.30.45:80	SUCCESS	Length: 0. segnum:	0
6:37.2 If Sat 154644603 76	SUCCESS	Longth: 0, segnum	Ú.
6:37.2 ■ Sat15d470e8e 4464 @ TCP Disconnect. 192.168.22.133.57405 > 159.69.246.184.13127	SUCCESS	Length: 0. segnum:	0
6:37:24.8535508 AM #603 76 2 TCP Reconnect 192.168.22.133.57408 -> 192.168.22.128.443	SUCCESS	Length: 0, segnum:	0
6372. mtSat19484603 76 2 TCP Reconnect 192.168.22.133.57408 > 192.168.22.128.443	SUCCESS	Length: 0, segnum	0
6:37.2 Sat 194644603 76	SUCCESS	Length: 0. segnum:	0
6:37.2 In Sat 194644603 76 @ TCP Deconnect. 192.168.22.133.57408.⇒ 192.168.22.128.443	SUCCESS	Length: 0, segnum:	0
6:37.2 Sat 194644603 76 CTCP Reconnect 192:168:22:133:57410 ⇒ 192:168:22:128:443	SUCCESS	Length: 0, segnum:	0
6:37.2. 55 Sat 194544603. 75 📿 TCP Reconnect 192:168.22 133:57410 -> 192:168.22 128:443	SUCCESS	Length: 0, segnum:	0
6.37.2. Set 194644603 76 CTCP Reconnect 192.168.22.133.57410> 192.168.22.128.443	SUCCESS	Length: 0, segnum	0
6:37.2	SUCCESS	Length: 0. segnum:	0
6:37.2 🔁 Sat 19170999bd 9988 💭 TCP Reconnect 192.168.22.133.57407 -> 212.193.30.45.80	SUCCESS	Length: 0, segnum:	0
6:37.2 Sat 194644603 76 CTCP Deconnect. 192.168.22.133.57410 ⇒ 192.168.22.128.443	SUCCESS	Length: 0, segnum:	0
6:37.2 55 Sat 194644603 75 QTCP Reconnect 192.168.22.133.57412 > 192.168.22.128.443	SUCCESS	Length: 0, segnum:	0
6:37.2	SUCCESS	Length: 0, segnum:	0
6:37.3 5at194644603 75 TCP Reconnect 192.168.22.133.57412 > 192.168.22.128.443	SUCCESS	Length: 0. segnum:	0
6:37.3	SUCCESS	Length: 0, segnum:	0
6:37.3 In Sat 194644603 76 GTCP Reconnect 192.168.22.133.57412.⇒ 192.168.22.128.443	SUCCESS	Length: 0, segnum:	0
6:37:3 Sat 194644603 75 😨 TCP Deconnect. 192 168:22:133:57412 -> 192 168:22:128:443	SUCCESS	Length: 0. segnum:	0
6:37.3 Sat 154644603 76 🖵 TCP Reconnect 192:168:22:133:53040 -> 192:168:22:128:443	SUCCESS	Length: 0, segnum:	0
6:37.3 #Sat 195518974 8992 @ TCP Reconnect 192.168.22.133:53041 ⇒ 192.168.22.128:80	SUCCESS	Length: 0, segnum	0
6:37.3 mssat19464603 75 ♀ TCP Reconnect 192.168.22.133.53040 -> 192.168.22.133.53040 -> 192.168.22.128.443	SUCCESS	Length: 0, segnum:	0
6.37.3. T Set 194470eSe4654 CTCP Reconnect 192.168.22.133.57413 159.69.246.184.13127	SUCCESS	Length 0 secrem	0

Description Contractor descriptions de seree						
Froces Montor - Systematic www.systemat.com					0	
0 🗄 (: : : : : : : : : : : : : : : : : : :	○ / ▦ 🔛 🗗 🗞 🗗					
Time Process Name PID Operation Path		Beaut	Detail		TID	
6.37.1. Mpowenheil.exe 7448 CueryStandard	System32/CatRost\[F750EBC3-38EE-11D1-85E5-00C04FC295EE]\Microsoft-Windows-Cle	SUCCESS	AlocationSize: 217.		7075	
6.37.1. Appowenheil.exe 7448 CouteFieMappC/Windows/	System32%EatRoat%[F750EBC3-38EE-11D1-85E5-00C04FC295EE]/Microsoft-Windows-Cle .	FILE LOCKED WI.	Sync Type: Sync Ty		70.75	
6.17.1. Approximate and 7448 [CountyStandard]C.Windows) 6.17.1. Approximate and 7448 [CountyStandard]C.Windows)	System 32-Cat Root (F750EBC3-38EE-11D1-85E5-00CD4FC255EE)/Microsoft-Windows-Cae System 32-Cat Root (F750EBC3-38EE-11D1-85E5-00CD4FC255EE)/Microsoft-Windows-Cae	SUCCESS	AllocationSize: 217.		70.75	
637.1. Showenhell ass 7448 CountrientioC/Windows/	System32/CatRoot/JF750E9C3-38EE-11D1-85E5-00C04FC255EE//Mcrosoft-Windows-Cle	SUCCESS	VolumeCreation Tim.		7875	
6:37.1. Browenhelless 7448 QueryAlinforms. C/Windows/	System32/CatRoot/(F750EBC3-38EE-11D1-85E5-00C04FC295EE)/Microsoft-Windows-Cle .	BUFFER OVERFL	Creation Time: 12/6		7875	
6:37:1. Approvembell.cos 7448 Constellis C/Windows/	System32\CatRost\F750E6C3-38EE-11D1-85E5-00C04FC295EE}\Microsoft-Windows-Cle	SUCCESS	Desired Access: R.		7875	
6.37.1. Sat155183/4. 9728 PC-ceateFile C/Usern/Use	r/AppData/Local/Temp/725043/PC50/Sat1955183/4c.exe Sub-2017/s/Death/E726E8/19.98EE.510/1.92EE.00/162EE2/38-accels/Western /Se	SUCCESS	Desited Access: G.		1652	
6.37.1. Reported and 7448 P.Couefile C/Windows	System 20 Cat Root/ JF750EBC3/38EE.11D1-85E5.00C04FC295EE7 Mechanist Windows Clin.	SUCCESS	Charlen line: 12/5		7876	
6:37.1. 45 Sat 195518974. 9728 Query Standard L.C./Usern/Use	r\AppData\Local\Temp\72S0437FCS0/Sat195518974c.exe	SUCCESS	AlocationSize: 1.5.		1682	
6:37:1 @Sat195518974 9728 @QueryStandardL.C/Usern/Use	r\/ppData\Local\Temp\7rS0437FC50\Sat195518974c.exe	SUCCESS	AlocationSize: 1,5.		1682	
6:37.1. Sat 195518974 9728 Read File C:/Users/Use	r\AppData\Local\Temp\7zS0437FC50\Sat195518974c.exe	SUCCESS	Offset: 870,425. Le.		1692	
6.37.1. #95at195518374. 9728 ENHeadHe COlleen/Use 6.33.1 # Curtists195519378 9728 ENDeadEte Colleen/Use	r/AppData/Local/Temp1/25043/HC50/Sat1955189/Ac.exe // AppData/Local/Temp1/25043/HC50/Sat19551897Ac.exe	SUCCESS	Offset: 870,490, Le.		1682	
6.37.1 dl Sat 1955 (8574 9728 PC) we Standard Children Use	Proppose Cock Temp (2000) Cold STRC 50 (Sec 1955) For the cold (Vanishing Cock) Temp (2000) Cock (1955) For the cold (Vanishing Cock) Temp (2000) Cock (1955) For the cold (Vanishing Cock) (1955) For the cock (1955) For the cold (1955) For the cock (1955) For the cock (1955) For the cold (1955) For the cock (1955) For	SUCCESS	Monation Spec 15		1652	
6.37.1. Sat 195518974. 9728 En Read File CAUsers/Use	http://www.acah.temp17/250437FC501/Sat195518974c.exe	SUCCESS	Offset: 870,499, Le		1692	
6:37.1 🌉 Sat 195518974 9728 🖺 Read File C./Users/Use	rV/ppData1Local/Temp17z50437FC501Sat195518974c.exe	SUCCESS	Offset: 870,503, Le.		1692	
6.37.1. Sat 195518974. 9728 Read File C/Users/Use	<pre>iVAppData\Local\Temp\72S0437FC5D\Sat195518974c.exe</pre>	SUCCESS	Offset: 874,599, Le		1692	
6.37.1. #Sat195518974. 9728 PreadFile Critices/Use	rVppData1LocahTemp17z90437FC501Sat199518974c.exe	SUCCESS	Offset: 874,603, Le.		1692	
6.37.1. (# Sat 1950 18974 3728 En Headrie Critisers Use 6.37.1. (# Sat 1955 19538	ProppData (Local) Temp/ /2508/379C30/Sat 1955189780 even	SUCCESS	Offset 878,639, Le		1632	
6.37.1. Sat 195518974. 9728 CreateFile C/Users/Use	(AppEata'Local/Temp	SUCCESS	Desired Access: R.		1652	
6.37.1. 25at 195518974. 9728 Cuery Basic Infor. C'Users/Use	(AppData)Local/Temp	SUCCESS	CreationTime: 12/6		1692	
6.37.1. #Sat195518974. 9728 CoseFile C/Users/Use	(AppEata)Local/Temp	SUCCESS			1632	
6:37:1. Sat195518974. 9728 CeateFile C'Users/Use	('AppData'Local\Temp\is GIS4H tmp	NAME NOT FOUND	D Desired Access: R.		1632	
COMPACT COMPACT AND COMPACT	rvepptuita utocan tempria sicoli Hamp	SUCCESS	Dealed Access: R.		1602	
Time Process Name PID Operation	Path			Result		
6:37:1 Testss.exe 640 RegOpenKey	HRLM\SOFTWARE\Morosoft\Windows\CurrentVersion\SdeBySde\Wi	mers/w86_micros	soft windows.c	SUCCESS		
6:37:1 Torss.exe 640 ERegQueryValue	HRLM-SOFTWARE/Microsoft/Windows/CurrentVersion/SideBySide/Winners1x85_microsoft.windows.cSUCCESS					
6.37:1 Transa exe 640 PegCloseKey	HKI,M\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\We	ners/x85_micros	soft windows.c	SUCCESS		
6:37:1 Tesrss.exe 640 RegCloseKey	HKLM\SOFTWARE\Morosoft\Windows\CurrentVersion\SideBySide\Wi	mers/w86_micros	soft windows.c	SUCCESS		
6:37:1 Torss.exe 640 ERegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Wi	ners		SUCCESS		
6:37:1 Tesetup_install.exe 2480 EPegOpenKey	HKI,M\Software\WOW6432Node\Microsoft\Windows\CurrentVersion\SideBySide					
6:37:1 Tesetup_install.exe 2480 PegSetInfoKey	HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\SideBySide					
6:37:1 Tsetup_instal.exe 2480 EPRegQueryValue	HKLM\SOFTWARE\W0W5432Node\Mcrosoft\Windows\CurrentVersio	n\SideBySide\Pr	eferExternalMa	NAME NOT	FOUND	
6:37:1 Testup_instal.exe 2480 EPegCloseKey	HXI,M\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVension\SideBySide					
6:37:1 Tesras exe 640 PegQueryValue	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Pub	ksherPolicyChan	ige Time	SUCCESS		
6:37:1 🗱 Sat 195518974 9728 🎬 RegOpenKey	HKLM\System\CurrentControlSet\Control\Ns\CodePage			REPARSE		
6:37:1 1 Set 195518974 9728 E RegOpenKey	HKI,M\System\CurrentControlSet\ControlWis\CodePage			SUCCESS		
6:37:1 #Sat 195518974 9728 PegQueryValue	HKLM\System\CurrentControlSet\ControlWa\CodePage\ACP			SUCCESS	1	
6:37:1 #Sat195518974 9728 #RegQueryValue	HKLM\System\CurrentControlSet\Control\Ns\CodePage\OEMCP			SUCCESS	1	
6:37:1 18 Sat 195518974 9728 E RegCloseKey	HKLM\System\CurrentControlSet\ControlWis\CodePage			SUCCESS		
6.37.1 1 Sat 195518974 9728 PegOpenKey	HKLM\System\CurrentCentrolSet\Control\Seasion Manager			REPARSE		
6:37:1 1 Sat 195518974 9728 PegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager			SUCCESS	1	
6:37:1., 18 Sat 195518974., 9728 ReqQueryValue	HKLM\System\CurrentControlSet\Control\Session Manager\RaiseExcept	ionOnPossibleDe	adlock	NAME NOT	FOUND	
6:37:1 195518974 9728 D RedCloseKey	HKLM\System\CurrentControlSet\Control\Session Manager			SUCCESS		
6:37:1. 15518974. 9728 PeoOpenKey	HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Segment H	leap		REPARSE		
6:37:1 #Sat195518974 9728 B RecOpenKey	HR, M\System\CurrentControlSet\Control\Session Manager\Segment He	80		NAME NOT	FOUND	
6.37.1 1 Set 195518974 9728 B RecOver Key	HKI, M\SYSTEM\CurrentControlSet\Control\Section Metacore	-		REPARSE	1	
6:37:1 44 Sat195518974 9728 10 BenOpenKey	HIC M/Sectors/ControlSet/Control/Service Manager			SUCCESS		
6-17-1 4 Set195511974 9728 #70-0-0-01/date	100 M/Sustan/Outer/Centre/Set/Centre/Session Manager	icies.		NAME NOT	COUND	
6.37.1 d 5.4165519974 9738 PP	HKI Mi Sustami Cameri Cantroli Sati Cantroli Session Manager Viesou Cerro			SUCCESS		
6-37-1 Curtok 19374 0710 1970-00-01/14	House System Control Control Control (2010) Second - ET-Control 2017.	0011-040-000	4620-47	NAME NOT	COUND	
6.37.1 (Cat105510074 0710 0710 0710 0710 0710 0710 0710	HALP CAMERICAN CONTRACT AND A CONTRA	201102929709	e pozitica /	DCDADSC	- NAME OF	
6.37.1 d c.4166519974 9739 BPD-0	HALP IN Sectors (Carter) Carter (Carter) Sector (Carter) Sector (Carter)			SUCCESS		
5.57.1 Cutocatory 0710 m270	HALM System Control Carbon Carbon Settion Markager	Loine		NAME NOT	COUND.	
eral. 1 Margari 1300163/4 31/28 Mil Medcher/Agne	HALM Gyaleh Laherellere ookt Leheor Seaach Mahager Vesourcero	LOC3		INVINE NO I	FOUND L	

Using some of the pre-built filters located on the toolbar makes it easier to parse through some of the data. This information could be further whittled down if necessary, but during the initial glance I prefer to keep as much digestible information as necessary. Process Monitor has a "process tree" feature, pictured below:

Unity show processes still running at end of current trace				
timetines cover displayed events only				
Process	Description	Image Path	Life Time	Company
🖂 🎇 tetup_x86_x64_instal.exe (5312)		C:\Usen\User\D		
🖂 🚰 setup_installer.exe (832)	7z Setup SFX	C:\Users\User\Ap		Igor Pavlov
setup_install exe (2400)		C:\Users\User\Ap		
Conhost exe (9796)	Console Window	C:\Windows\Syst		Microsoft Corp
□ 225 cmd.exe (3776)	Windows Comma	C.\Windows\Sys		Microsoft Corp
powershell exe (10164)	Windows PowerS	C:\Windows\Sys		Microsoft Corp
Em and.exe (4356)	Windows Comma	C:\Windows\Sys		Microsoft Corp
powenheil exe (7443)	Windows PowerS.	C:\Windows\Sys		Microsoft Corp
cmd.exe (8332)	Windows Comma	C:\Windows\Sys		Microsoft Corp
Sat 1984/0e8e059 Ac4/.exe (10156)		C:\Users\UserVp		_
Sal 1304/06/003/104/ exe (4464)		C. Users User Vp		
Cmd.exe (10156)	Windows Comma	Children Uhren An		Microsoft Corp
Sat 1917/23/363361 exe (2366)	Jao mpo	Critical Control Contr		Lashina Manada Care
Child Cat (2620)	Evel shi bired der	CALINEROUNS COVE		Fiel abl kinets
C 4195518974c leve (3720)	Stellingtal	Ciller/Jer/Je		FarLab Uninste
 Will Sat 1955 19974c ann (9376) 	Earl abilitization	C\Uber\Uber\do		FarLabUninsta
ill Sat 195518974c two (8992)	Setup (Ikindal)	C\lben\lber\do		Farl abl binsta
- 25 cmd eue (1112)	and a second		_	
escription:				
motor				
db: C\User\User\Decktoe\mahuare\a12d7db1756	449531-2167556-62040-4		<781	
amount "Colline of load package and and all of the 175	LANS21-71/754-470.40	A.C. 925260924-604	6.79	
ommanu: Cityserstusertu/esktopt/mawareta120/401/3	10493316211/3316120498	00003020708340094	3610	
ver: WINDEV2112EVAL\User				

In this image, it can be observed that the processes associated with the malware spawn cmd.exe, which then spawns powershell.exe. There's an event in the Process Monitor data showing PowerShell with a "Process Start" operation after spawning from cmd.exe. This caught my interest because it was a unique operation in comparison with the bulk of PowerShell events. Further analysis resulted in catching this:



For those unfamiliar, Set-MpPreference is a PowerShell cmdlet associated with Windows Defender scans. The malware authors developed the specimen such that it would leverage PowerShell to disable Defender monitoring on the directory where the malware resides. In layman's terms, they're using PowerShell to tell Defender "nothing to see here" when it comes to the folder where all of the badness is. This is what could be considered an evasive maneuver.

A useful tool related to Process Monitor is ProcDOT. Developed by Christian Wojner, it provides correlation and visualization of Process Monitor and PCAP data. If there's a ton of data, the visual can seem convoluted from a zoomed out view, but it helps in showing how processes, file writes, and other events are related.





Without being able to connect back to its "mothership," the malware doesn't do much more than what's already been displayed. In order to further analyze a specimen of this nature, being reliant on a network connection, it would be useful to utilize dispensable hardware, a public network, and a VPN solution, with the intent being to do a full system restoration after detonation. This method is labor intensive and risky and not recommended for folks with minimal experience.

SO...WHAT IS REDLINESTEALER?

RedLineStealer is a credential stealer that targets web browsers. Access to the tool is available on the forums for several hundred dollars. The website HavelBeenPwned <u>recently added almost half a million</u> <u>entries</u> related to RedLineStealer credential theft. Check it out <u>here</u> if you have concerns regarding your creds and whether or not they've been compromised in a malicious campaign.

Indicators of Compromise (IOCs) for RedLineStealer

File name: setup_x86_x64_install.exe

File hash: a12d74b1756d49531e21f755fef2049ab6c83626f0834cb945c781c39d40a177

File name: Sat19d470e8e0597fc47.exe (or similarly named matching the same alphanumeric pattern)

File path: C:\Users\User\AppData\Local\Temp\7zS4441B019\Sat19d470e8e0597fc47.exe

File hash: BC118B7708D56B93707A9BB025D3BF62D723B7932435A08299F59249C1C37DBE

File name: @.cmd

File path: C:\Users\User\AppData\Local\Temp\IXP000.TMP\@.cmd

File hash: 286227287F1FA79D5D5D909C2F457FC4D0AEFA6BE9E940F9A1F214D113FF88B4

File name: Sat195518974c.exe

File path: C:\Users\User\AppData\Local\Temp\7zS0437FC5D\Sat195518974c.exe

File hash: 13357A53F4C23BD8AC44790AA1DB3233614C981DED62949559F63E841354276A

File name: IXP000.TMP

File path: C:\Users\User\AppData\Local\Temp\IXP000.TMP

Directory: C:\Program Files (x86)\FarLabUninstaller*

Domains associated:

www.hhiuew33[.]com

gp.gamebuy768[.]com

one-mature-tube[.]com

cloudjah[.]com

kelenxz[.]xyz

ad-postback[.]biz

IPs associated:

212.193.30[.]45

159.69.246[.]184

Registry keys:

HKLM\SOFTWARE\Microsoft\Tracing\Sat194d446031aec9ca_RASAPI32 HKLM\SOFTWARE\Microsoft\Tracing\Sat194d446031aec9ca_RASMANCS HKLM\SOFTWARE\Microsoft\Tracing\Sat19f1c04426464e86_RASAPI32 HKLM\SOFTWARE\Microsoft\Tracing\Sat19f1c04426464e86_RASMANCS

HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Uninstall\FarLabUninstaller.exe_is1

TOOLS AND REFERENCES

- Process Monitor
- ProcDOT
- WireShark
- Fiddler
- Process Hacker
- FakeDNS
- <u>RegShot</u>