sLoad and Ramnit pairing in sustained campaigns against UK and Italy | Proofpoint US

By October 23, 2018 Proofpoint Staff

Published: 2018-10-23 · Archived: 2026-04-05 14:58:35 UTC

Editor's note: This post has been updated to reflect a change in TTPs for the actor that occurred after the original blog was finalized.

Overview

Since May 2018, Proofpoint researchers have observed email campaigns using a new downloader called sLoad. sLoad is a PowerShell downloader that most frequently delivers Ramnit banker and includes noteworthy reconnaissance features. The malware gathers information about the infected system including a list of running processes, the presence of Outlook, and the presence of Citrix-related files. sLoad can also take screenshots and check the DNS cache for specific domains (e.g., targeted banks), as well as load external binaries. In this post we will:

- Introduce sLoad
- Describe sLoad campaigns by an actor with long history of activity, including the personalization of email messages with the recipient's name and address
- Cover geographic targeting of the UK, Italy, and Canada, particularly via geofencing, which is performed at multiple points in the infection chain.

Delivery

While initial versions of sLoad appeared in May 2018, we began tracking the campaigns from this actor (internally named TA554) since at least the beginning of 2017. Other researchers also noticed some of these campaigns [2][3][4]. The following figure shows a snapshot of the actor’s recent activity, starting slightly before the introduction of sLoad.

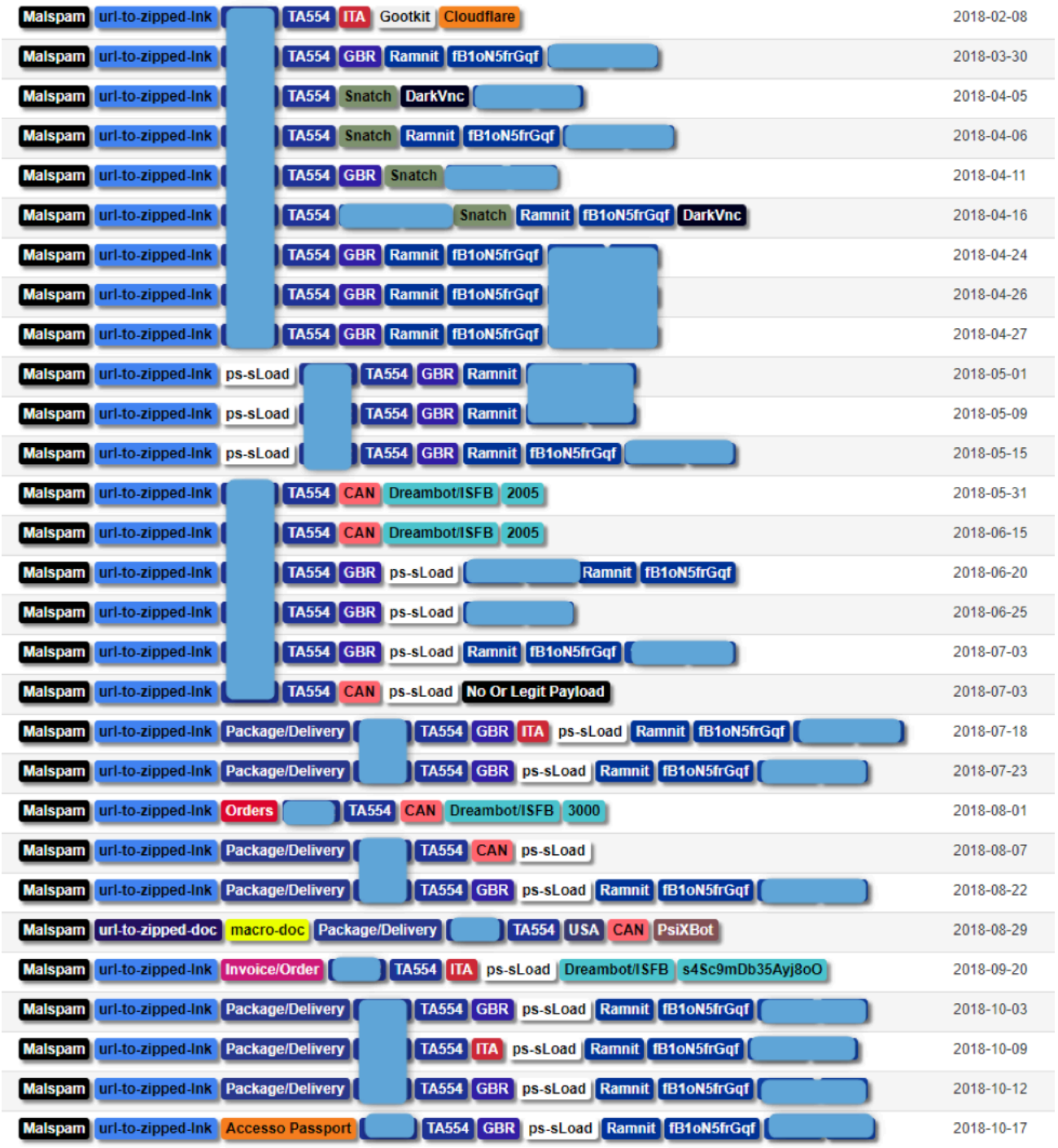

Figure 1: Snapshot of TA554's recent activity

Historically, this actor has targeted Italy, Canada, and the United Kingdom, specifically sending malicious emails to recipients in these countries. The emails are crafted in the targeted country's language and are often personalized to include recipients' names and addresses in various parts of the email such as email body and subject. TA554 frequently uses package delivery or order notification lures; the emails contain URLs linking to zipped LNK files or zipped documents. The LNK file or document macros in turn download the next stage -- typically a PowerShell script which may download the final payload or another downloader such as sLoad.

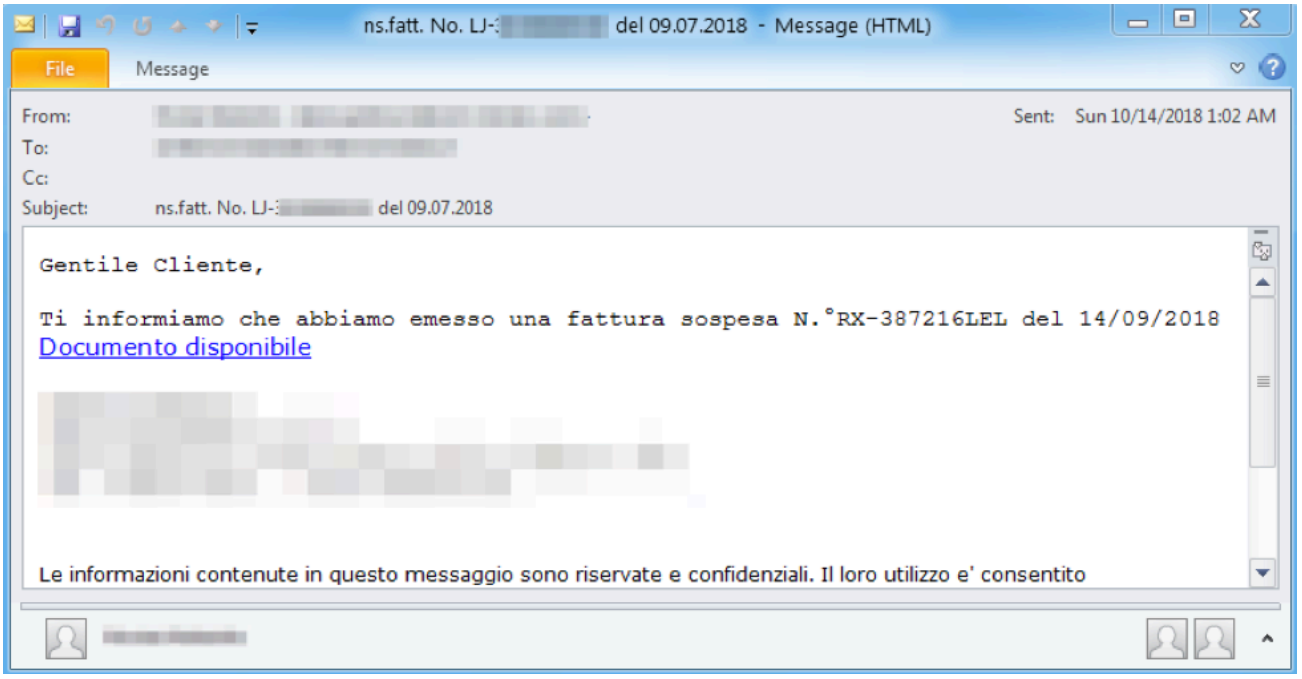

Figure 2: Email targeting Italian recipients on October 14, 2018

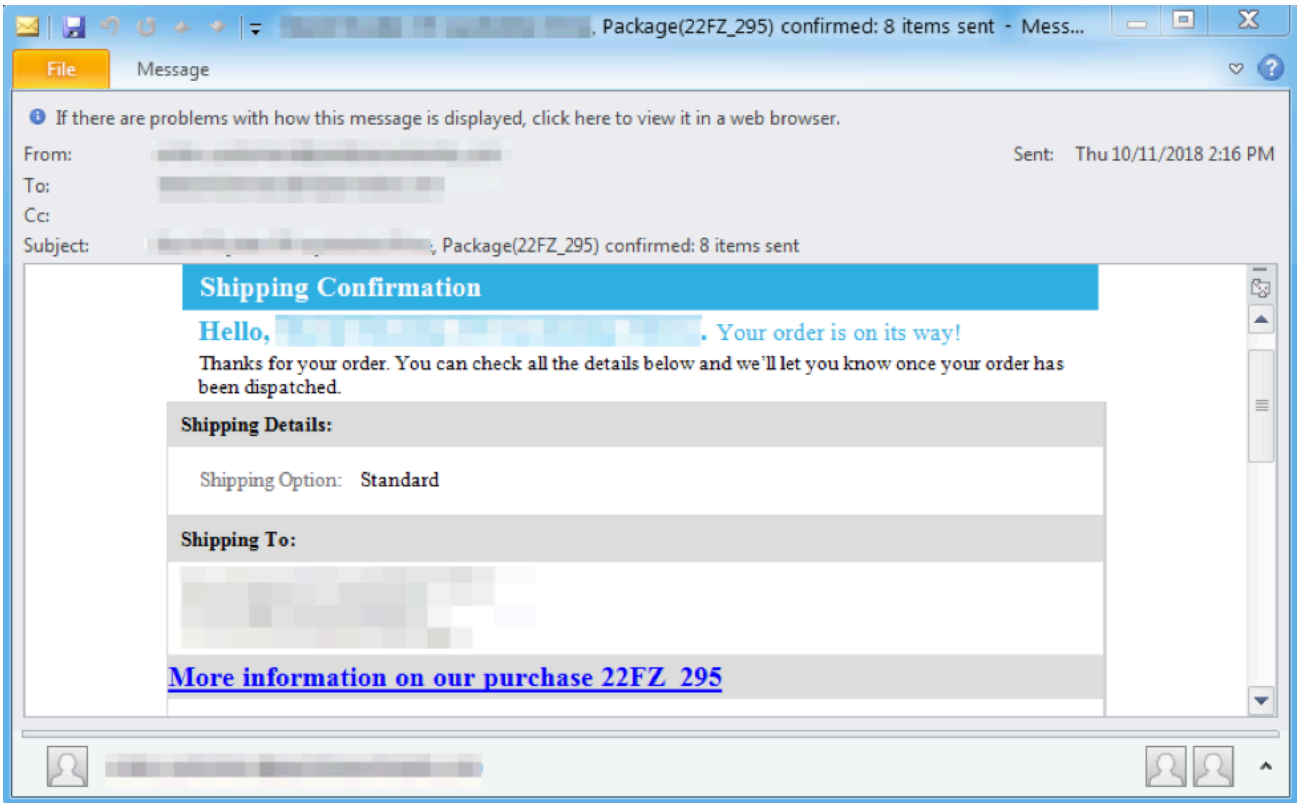

Figure 3: Email targeting United Kingdom recipients on October 11, 2018. This email was personalized to include the recipient's name and address

The actor frequently, but not always, uses one or more intermediate downloader, such as an as yet unnamed PowerShell script, sLoad, Snatch, or Godzilla. We have observed final payloads including Ramnit, Gootkit, DarkVNC, Ursnif, and PsiXBot.

Geofencing -- restricting access to content based on the user's location, determined via the source IP address -- is performed at all steps of the infection chain. For example, we observed checks being performed at:

1. Download of the zipped-LNK
2. LNK downloading PowerShell
3. PowerShell downloading sLoad
4. sLoad communications with its command and control (C&C)
5. sLoad receiving a task/command (base64-encoded binary)

Steps 2 and 5 are additionally "Headers-fenced", meaning that the request must also match those of BITS (Background Intelligent Transfer Service).

Malware Analysis

Overview

Figure 4 shows an overview of the network traffic in this infection chain.

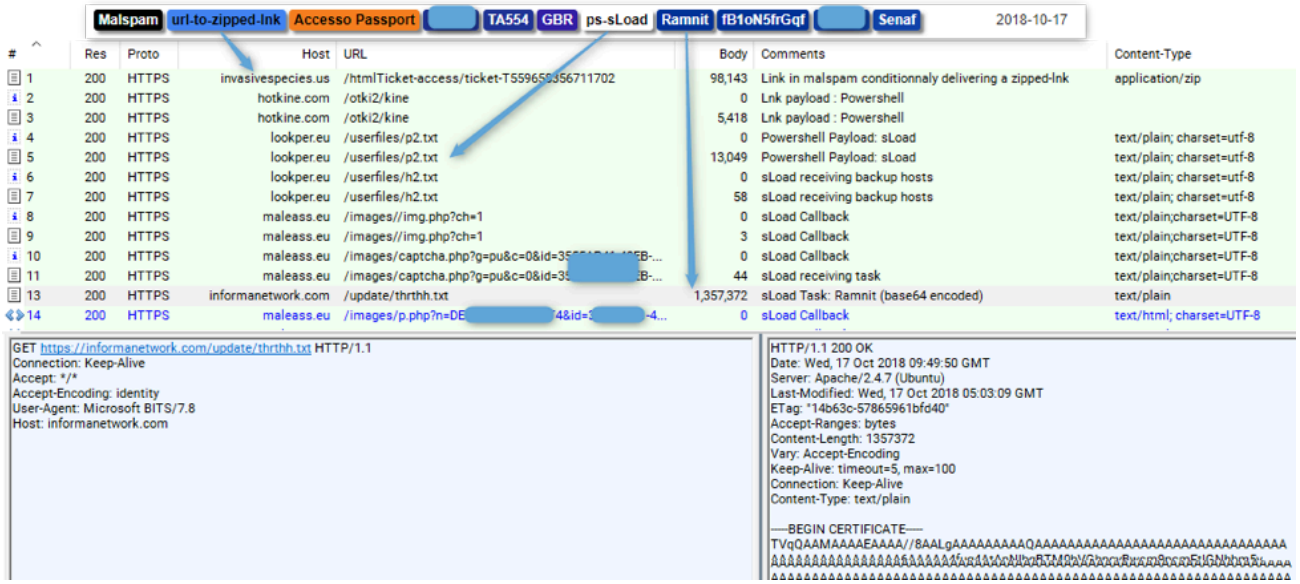

Figure 4: The full infection chain on October 17, starting from the user's click on a link in a malicious email, to the download of PowerShell and sLoad, to subsequent sLoad C&C traffic

The main elements of the infection chain are detailed below:

- Line 1: the initial user click on the URL in email, resulting in the download of a zipped LNK (a Windows shortcut file [5]) from invasivespecies[.]us
- Line 3: the LNK, which was run by the user, downloads the next stage (PowerShell) from hotline[.]com/otki2/kine
- Line 5: PowerShell downloads sLoad (from lookper[.]eu/userfiles/p2.txt)
- Line 7: PowerShell downloads a file containing sLoad C&C hosts (from lookper[.]eu/userfiles/h2.txt)
- Line 8-9: sLoad initial beacon
- Line 10-11: sLoad reporting infected system information and polling for commands
- Line 13: sLoad downloading Ramnit, after receiving a command to do so. Note that we have observed extended waits -- more than one day -- until sLoad receives a command to download the next stage
- Line 14: sLoad sending screenshots to the C&C

LNK

Typically when we see LNK files used as the first-stage downloader, they tend to point to a PowerShell command that performs the download, all inside the link target field. With files like this, it is easy to extract and analyze the PowerShell command. For example, on Windows this can be performed manually by right-clicking on the shortcut file, selecting Properties, and analyzing the command in the "Target:" box.

Less commonly, data can be appended to the end of a LNK file after the termination block (four NULL bytes) [7] as Windows will stop reading data in the LNK after seeing a termination block. So it is possible to add [malicious] data to the end of the file which can be parsed externally using PowerShell / Certutil / external tools to execute code. We have observed this used to hide long series' of commands such as described in [6].

In our case, the additional commands are appended after the end of the LNK file. Hence, the link target field essentially contains a short "carving script" that finds and executes commands located after the end of the LNK file. The actual LNK is 1528 bytes long and additional 1486 bytes of PowerShell code is added at the end.

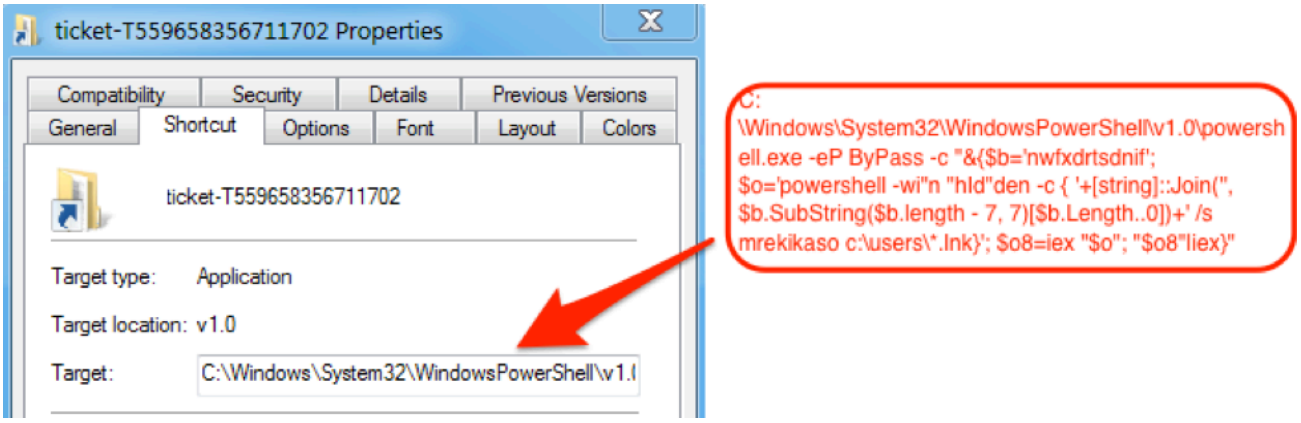

Figure 5: Screenshot of the example LNK properties

The "Target:" field contains an obfuscated command that uses the "findstr" ("nwfxdrtsdnif" reversed), a Windows grep-like command, to find the malicious code appended at the end of the LNK file, which is marked with the "mrekikaso" string.

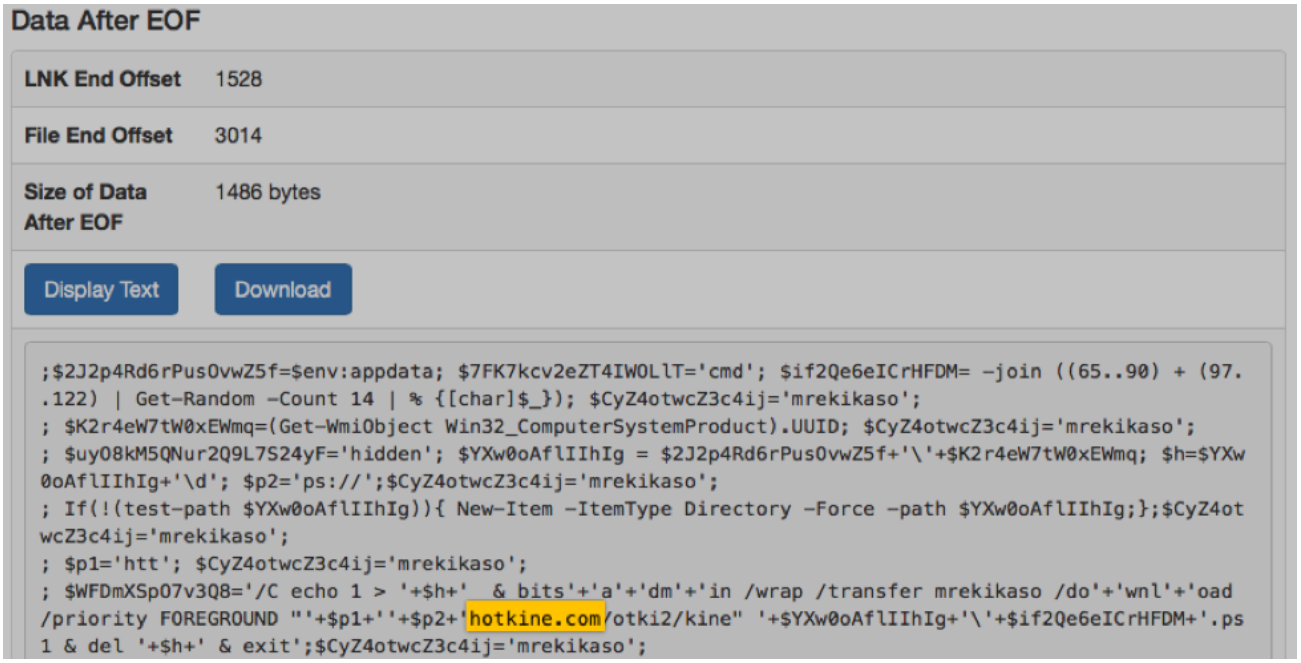

Figure 6: This screenshot shows the PowerShell code appended after the end of the LNK. This code performs the download of the next stage (more PowerShell)

sLoad Downloader

The LNK downloads a small PowerShell script (unnamed) which itself contains a few notable features:

- It performs a check to see if any security processes are running on the system and exits if found
- Downloads sLoad (e.g., from lookper[.]eu/userfiles/p2.txt) and stores it encrypted with a hardcoded key as "config.ini"
- Downloads sLoad C&C hosts file (e.g. from lookper[.]eu/userfiles/h2.txt) and stores it encrypted with a hardcoded key as "web.ini"
- Uses a Scheduled Task to execute sLoad

```
$p = @("windbg*","dumpcap*","Regshot*","windump*","ollydbg*","commview*",
"tcpdump*","Dbgview*","netsniffer*","Tcpview*","Fiddler*","win_dump*",
"regmon*","joeboxcontrol*","winspy*","joeboxserver*","wireshark*","idag*",
"sniff_hit*","smsniff*","idag64*","winapioverride32*","apimonitor*",
"ProcessHacker*","ImmunityDebugger*","plugin_host*","HashMyFiles*",
"pestudio*");
for ($i=0;$i -lt $p.length; $i++){ $r=Get-Process -name $p[$i]; if ($r){
stop-process -name powershell* }};
```

Figure 7: PowerShell (sLoad downloader) searching for security tools prior to performing any further action

sLoad

sLoad is also written in PowerShell. At the time of this writing, the latest version of sLoad was 5.07b, which we will analyze here. It includes noteworthy features such as:

- Collection of information to report to the C&C server that includes:
  - A list of running process

    - Presence of .ICA files on the system (likely Citrix-related)
    - Whether an Outlook folder is present on the system
    - Additional reconnaissance data
- The ability to take screenshots
- Checking the DNS cache for specific domains (e.g., targeted banks)
- Loading external binaries

sLoad's network communication begins with an initial C&C beacon to path "/img.php?ch=1", which is an empty request. It may receive an "sok" from the server.

After the initial beacon, sLoad enters a loop in which it pushes extensive information about the victim's system to the C&C, expects and executes commands from the server, and sends screenshots to the server. In this loop, it first performs a request to "captcha.php" and sends information about the infected system via the URL parameters.

Table 1: Breakdown of URL parameters and values in the "captcha.php" request

Parameter	**Example Value**	**Explanation**
g	"pu"	Hardcoded value
c	"0"	If any files with .ICA extension are found on the system, searched starting from the "C:\users" folder, this value is "1". Otherwise this value is "0". We assume .ICA files are the most likely Citrix-related.
id		System's UUID generated with: (Get-WmiObject Win32_ComputerSystemProduct).UUID
v	"Microsoft Windows 7 Ultimate"	OS caption generated with: (gwmi win32_operatingsystem).caption
c	"GLklWOaPjmVuQiCD"	Random string of 16 upper and lower letters, generated for each such request
a	"*armsvc*cmd*cmd*conhost"	"*"-separated list of running processes
d		The point of this parameter is to count the number of computers in the current domain or network. This parameter could be empty if there are none, or can have a value such as "{in network:1}"

n	“MARK-PC”	Computer name generated with: $env:ComputerName
bu	“*nwolb.com*barclays.co.uk”	“*”-separated list of hostnames from the system’s DNS cache that match the hostnames from a hardcoded list of targeted banks
cpu	“Intel(R) Core(TM) i5-780HQ CPU @ 2.91GHz”	System processor information
o	“0”	If "\..\Microsoft\Outlook\" (starting from current working directory) exists then “1”, else “0”

sLoad reads and saves the server’s response to the “captcha.php” request. If any response is returned, sLoad checks it and acts upon it. The response can begin with:

Table 2: Explanation of possible responses from the C&C to the “captcha.php” request

Server Response (begins with)	**Explanation**
“run=”	This is followed by a URL which is downloaded and its PowerShell content executed
“updateps=”	This is followed by a URL which is downloaded and its PowerShell content saved. Essentially this implements the “update self” functionality. The contents of the file storing sLoad on disk are replaced, and the current sLoad instance is stopped
Any other response with length greater than 3	Is expected to be a URL, whose content is downloaded, decoded with “certutil”, and saved as an executable, at which point the executable is started

Near the end of the main loop, sLoad will upload the screenshots it took of the victim’s Desktop to the “p.php” URI. sLoad executes a long sleep of 10 minutes before it polls the server again for commands and to upload additional screenshots.

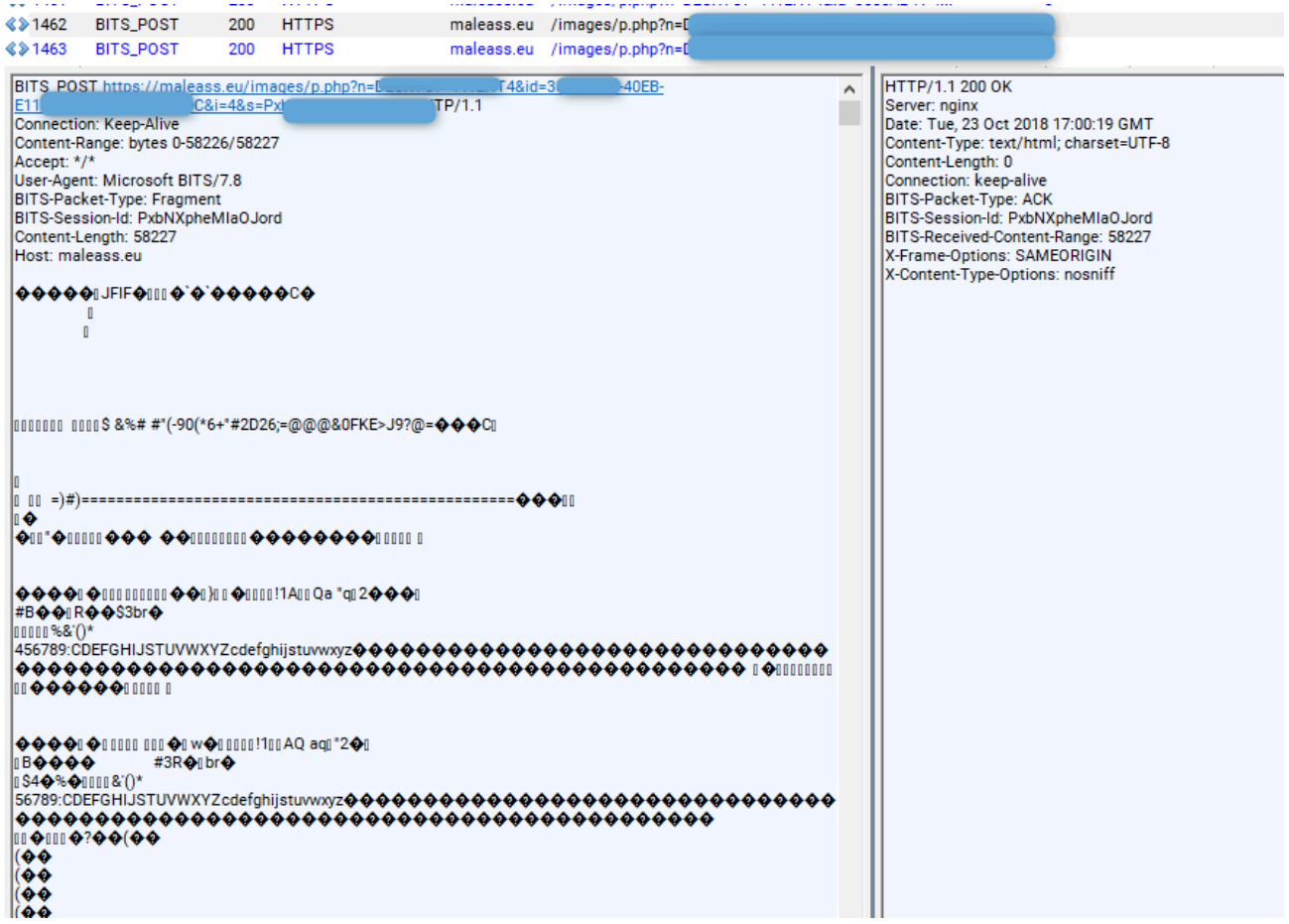

Figure 8: sLoad posting a screenshot to its C&C

```
$oB="";
$b = @("inbank.it","unicredit.it","intesasanpaolo","ubibanca","finecobank","creval","mps.it","bnl.it","poste.it");
```

Figure 9: sLoad contains a hardcoded array of banking keywords and hostnames (in this instance, for Italian banks). It compares the infected machine's DNS cache to this list, and reports any matches to the C&C in the "bu" parameter.

```
$oB="";
$b = @("nwolb.com","bankline","bankofscotland.co.uk","bankofscotland.co.uk","secure.lloydsbank.co.uk",
"secure.halifax-online.co.uk","hsbc.co.uk","rbsdigital.com","barclays.co.uk","onlinebusiness.lloydsbank","tsb.co.uk",
"retail.santander.co.uk","business.santander.co.uk","onlinebanking.nationwide.co.uk");
```

Figure 10: sLoad contains a hardcoded array of banking keywords and hostnames (in this instance for UK banks). It compares the infected machines DNS cache to this list, and reports any matches to the C&C in the "bu" parameter.

```
$cit=Get-ChildItem -Path c:\users -Filter *.ICA -Recurse -ErrorAction SilentlyContinue -
Force
if ($cit){ $ci=1; }
$ci | Out-File $path"\f.ini"
```

Figure 11: sLoad searching for files with .ICA extension, starting in "C:\users" folder. We assume these are most likely Citrix-related due to this format used for Citrix application servers as a configuration file and the "$cit" variable.

sLoad Versions

Since May 2018 we have observed multiple versions of sLoad, which introduced incremental changes.

Table 3: sLoad versions observed

Version	**Date Observed**
0.01b	2018-05-01
2.01b	2018-05-09
2.11b	2018-05-12
2.37b	2018-06-06
3.47b	2018-06-26
4.07b	2018-08-23
5.07b	2018-09-20
5.08b	2018-10-03

We were also able to observe control panels for a number of these versions (Figures 12-15).

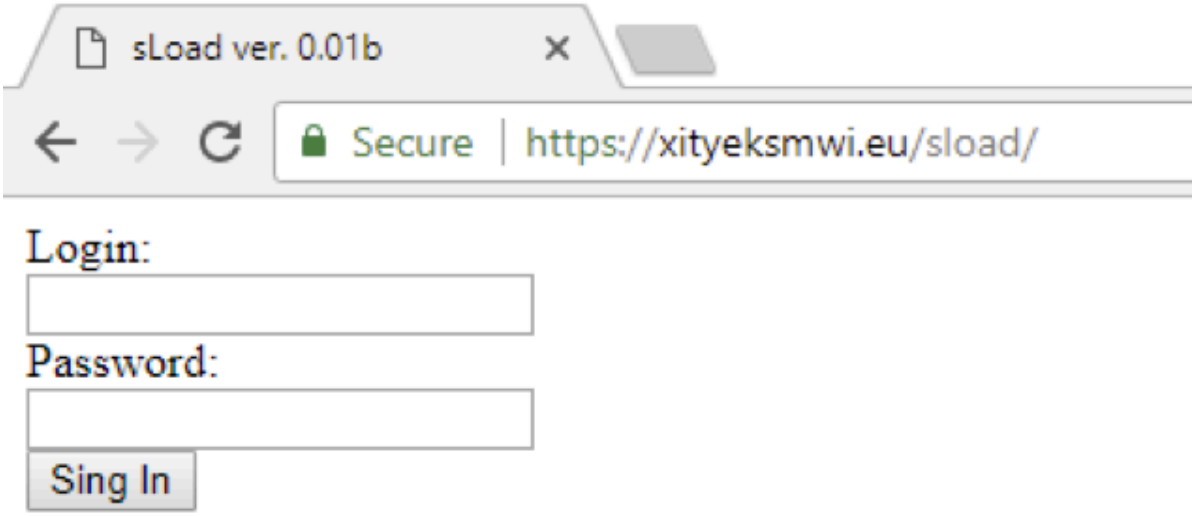

Figure 12: Screenshot of the C&C panel, version 0.01b

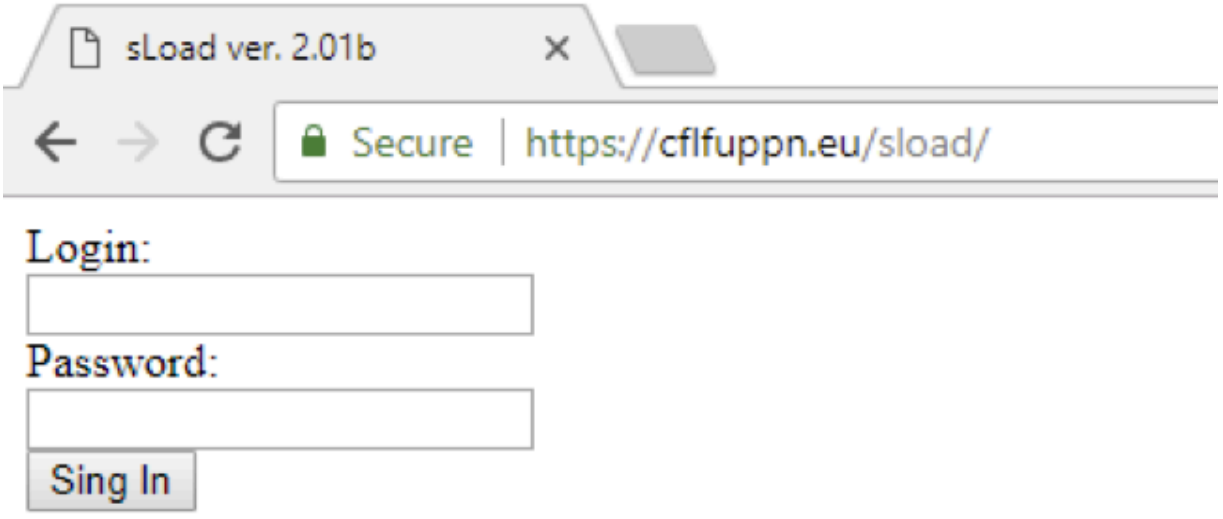

Figure 13: Screenshot of the C&C panel, version 2.01b

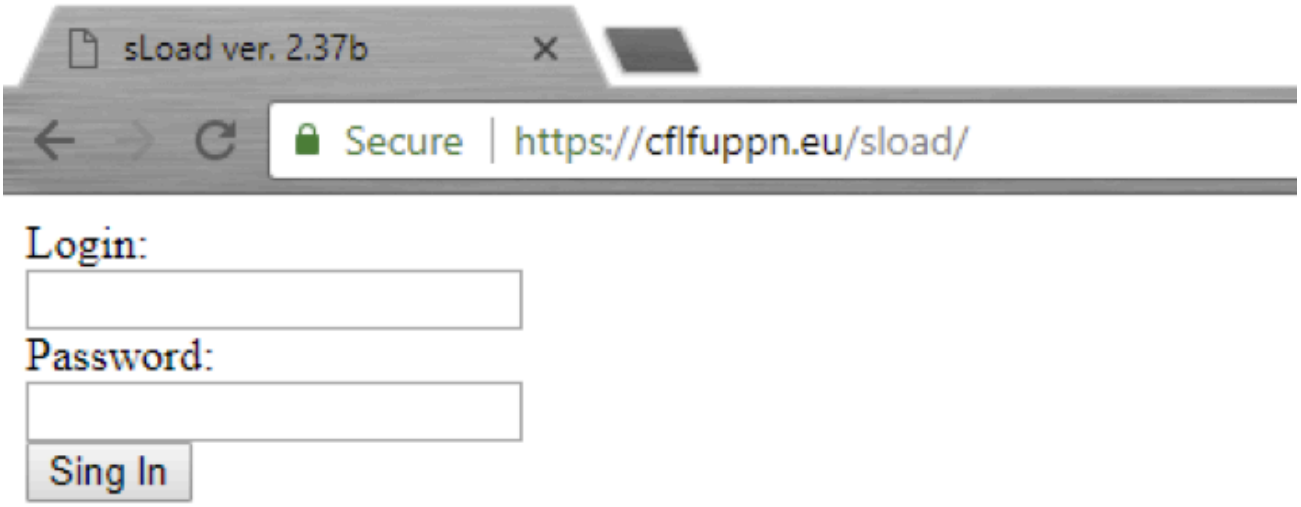

Figure 14: Screenshot of the C&C panel, version 2.37b

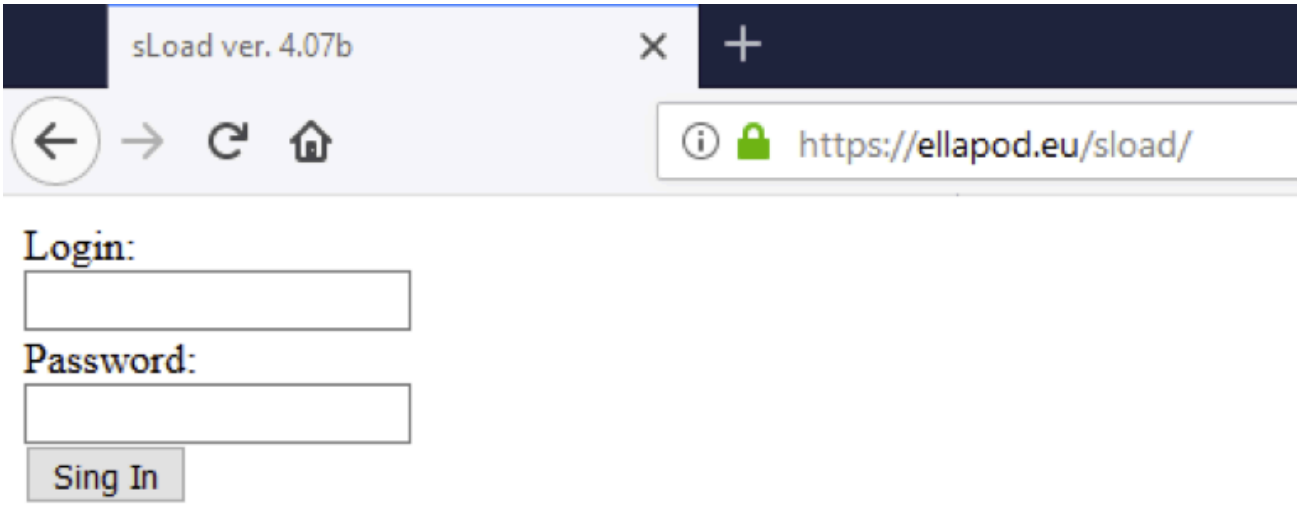

Figure 15: Screenshot of the C&C panel, version 4.07b

Updated October 23, 2018 - New TTP

On October 22, 2018, the actor added a victim facing landing at the zipped-lnk download step [8] (Figure 16). In this case, the .LNK was downloading sLoad directly without the additional intermediate PowerShell.

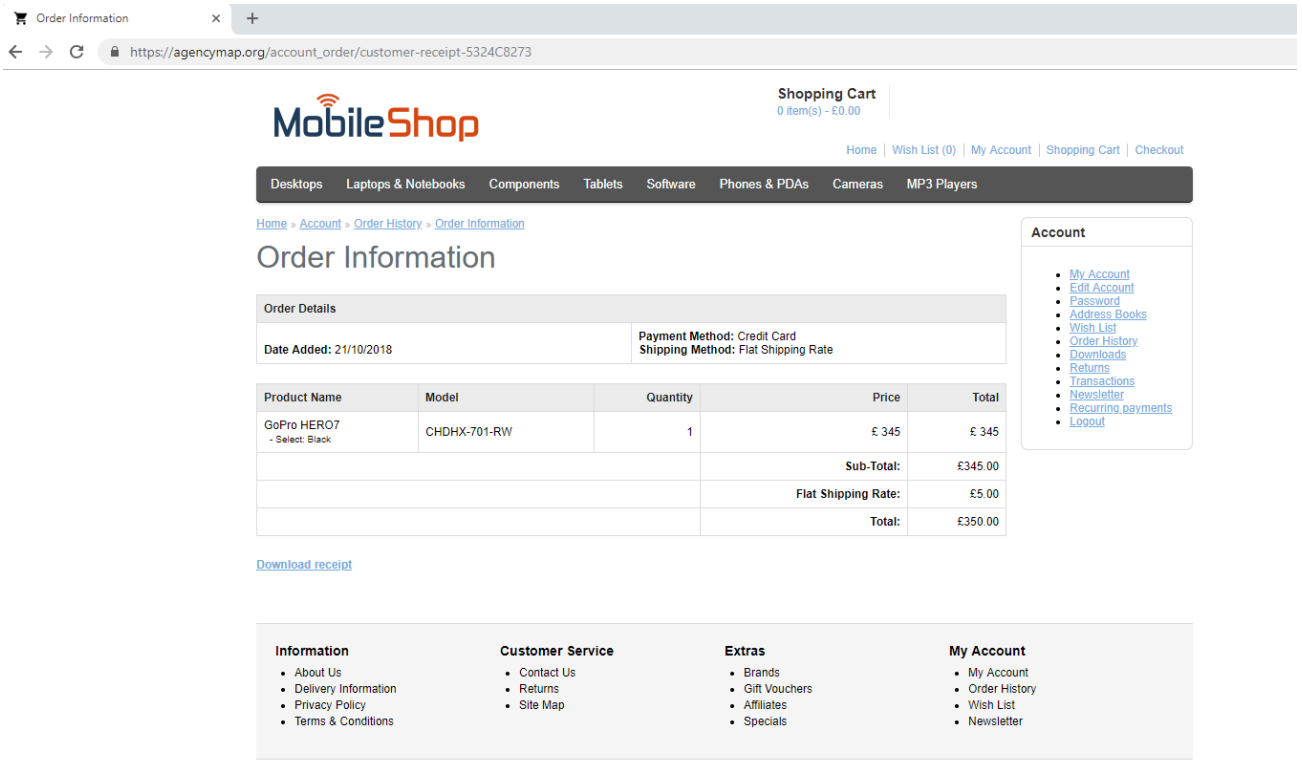

Figure 16: New victim-facing landing page

Conclusion

Proofpoint researchers identified yet another stealthy downloader, this time paired with personalized email lures and sophisticated geofencing. sLoad, like other downloaders we have profiled recently, fingerprints infected systems, allowing threat actors to better choose targets of interest for the payloads of their choice. In this case, that final payload is generally a banking Trojan via which the actors can not only steal additional data but perform man-in-the-browser attacks on infected individuals. Downloaders, though, like sLoad, Marap, and others, provide high degrees of flexibility to threat actors, whether avoiding vendor sandboxes, delivering ransomware to a system that appears mission critical, or delivering a banking Trojan to systems with the most likely return.

References

[1] https://asert.arbornetworks.com/snatchloader-reloaded/

[2] https://isc.sans.edu/forums/diary/Malicious+Powershell+Targeting+UK+Bank+Customers/23675/

[3] https://myonlinesecurity.co.uk/your-order-no-8194788-has-been-processed-malspam-delivers-malware/

[4] http://blog.dynamoo.com/2017/02/highly-personalised-malspam-making.html

[5] https://msdn.microsoft.com/en-us/library/dd871305.aspx

[6] https://www.uperesia.com/booby-trapped-shortcut-generator

[7] https://lifeinhex.com/analyzing-malicious-lnk-file/

[8] https://twitter.com/ps66uk/status/1054706165878321152

Indicators of Compromise (IOCs)

IOC	IOC Type	Description
hxxps://invasivespecies[.]us/htmlTicket-access/ticket-T559658356711702	URL	URL in email - 2018-10-17
hxxps://davidharvill[.]org/htmlTicket-access/ticket-V081650502356	URL	URL in email - 2018-10-17
hxxps://schwerdt[.]org/htmlTicket-access/ticket-823624156690858	URL	URL in email - 2018-10-17
5ea968cdefd2faabb3b4380a3ff7cb9ad21e03277bcd327d85eb87aaeecda282	SHA256	ticket-T559658356711702.zip - 2018-10-17
hxxps://hotkine[.]com/otki2/kine	URL	Zipped LNK gets PowerShell - 2018-10-17
a446afb6df85ad7819b90026849a72de495f2beed1da7dcd55c09cd33669d416	SHA256	kine - ps1 - 2018-10-17
hxxps://lookper[.]eu/userfiles/p2.txt	URL	PowerShell gets sLoad - 2018-10-17
hxxps://lookper[.]eu/userfiles/h2.txt	URL	PowerShell gets sLoad hosts file - 2018-10-17
79233b83115161065e51c6630634213644f97008c4da28673e7159d1b4f50dc2	SHA256	p2.txt sLoad - GBR - 2018-10-17
245c12a6d3d43420883a688f7e68e7164b3dda16d6b7979b1794cafd58a34d6d	SHA256	h2.txt sLoad hosts - GBR - 2018-10-17

hxxps://maleass[.]eu/images//img.php?ch=1	URL	sLoad C&C - 2018-10-17
hxxps://informanetwork[.]com/update/thrthh.txt	URL	sLoad payload (Ramnit) - 2018-10-17
b1032db65464a1c5a18714ce3541fca3c82d0a47fb2e01c31d7d4c3d5ed60040	SHA256	Ramnit - 2018-10-17
xohrikvjhiu[.]eu|185.197.75.35	DOMAIN|IP	Ramnit C&C - 2018-10-17

ET and ETPRO Suricata/Snort Signatures

2830633 || ETPRO TROJAN sLoad CnC Checkin M2

2830632 || ETPRO TROJAN sLoad CnC Checkin

2018856 || ET TROJAN Windows executable base64 encoded

Source: https://www.proofpoint.com/us/threat-insight/post/sload-and-ramnit-pairing-sustained-campaigns-against-uk-and-italy