Examining a VBA-Initiated Infostealer Campaign

By Vicky Ray, Rob Downs

Published: 2014-10-29 · Archived: 2026-04-05 18:28:26 UTC

While Microsoft documents that leverage malicious, embedded Visual Basic for Applications (VBA) macros are not a new thing, their use has noticeably increased this year, thanks in part to their simplicity and effectiveness.

Some threat actors commonly use this class of malware to drop a second stage payload on victim systems. Even though Microsoft attempts to mitigate this threat by disabling macros by default, the percentage of users who explicitly bypass this protection and enable macros remains high.

Exploiting the human factor, the most effective attacker strategy is the tried and true spear phishing attack, ideally made to look authentic by appearing to originate from a legitimate organization/individual and containing role-relevant or topic-of-interest content to entice its intended target. This post examines an information stealer campaign that leveraged a VBA macro script, focusing on its progression, from delivery to Command and Control (C2), and its attribution to a malicious actor for context on objectives and motivation.

Delivery and Exploitation

The recent campaign started with an email sent to an employee responsible for processing financial statements at a global financial organization (Figure 1). The sender's email address was spoofed as originating from an energy company. Subsequent analysis would show that this façade was very thin; yet, it is often all that is required to encourage a user to open an attachment or click on a link that then executes malicious code.

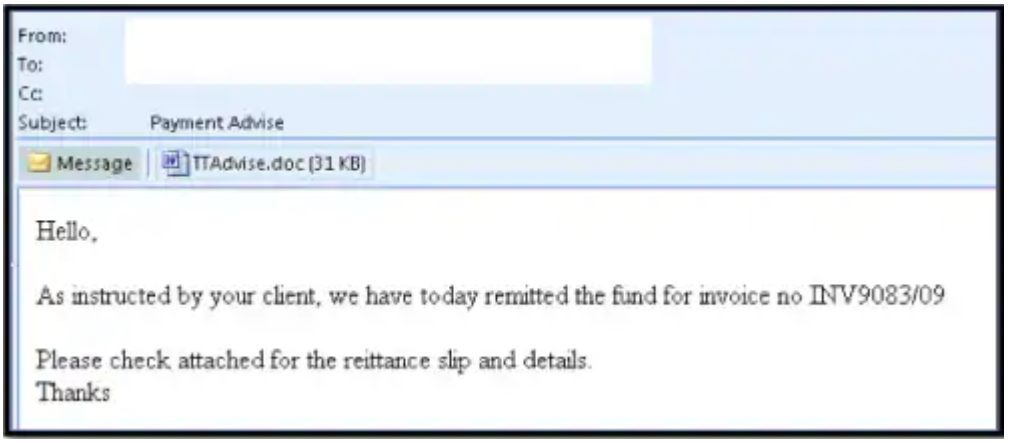

Figure 1: Delivery of a phishing message containing malicious DOC file

The above e-mail employs common pressure tactics for phishing messages. Specifically, it touches on two areas of potential concern for a target: financial responsibility and the introduction of a state of uncertainty and confusion. In this case, the role of the target as a processor of financial statements might mean that the target is accustomed to receiving similarly structured legitimate e-mails; accordingly, they may open a malicious attachment without a second thought.

The second factor is much broader and relates to how humans deal with uncertainty. Without specific awareness and training, some users may be inclined to open the attachment, wondering why the e-mail was sent to them. In psychology, this is referred to as the "Need for Closure" personality trap.

The next layer of this attack is found within the malicious DOC file once a victim opens it. With a system properly configured to protect against automatic execution of VBA macros, no malicious code has been run at this point. Figure 2 presents a screenshot of the malicious attachment's displayed contents.

Figure 2: Displayed contents of malicious DOC file, TTAdvise.doc

This content further compounds the two points of concern for the target, and now presents a convenient option of clicking on "Enable Content" to obtain closure on the matter. Despite a security warning (Figure 3), a number of users still choose to enable respective content, allowing for malicious VBA macros to run on their system.

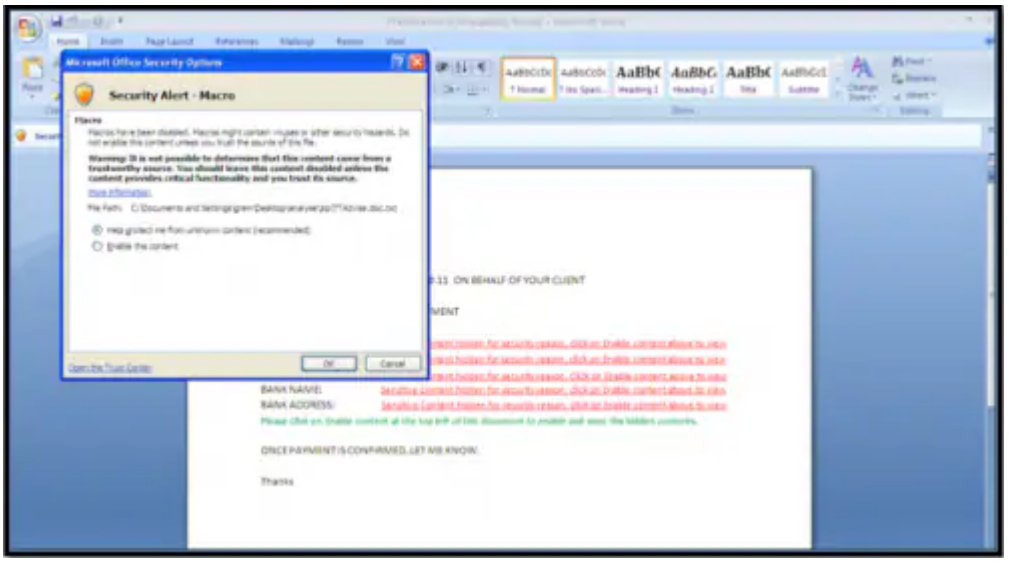

Figure 3: Often ignored Microsoft security warning against enabling macro content

After enabling macros, none of the promised data is shown to the victim; however, the malicious VBA macro script executes in the background without the user's knowledge.

VBA Macro Script

The embedded VBA macro script is shown in Figure 4.

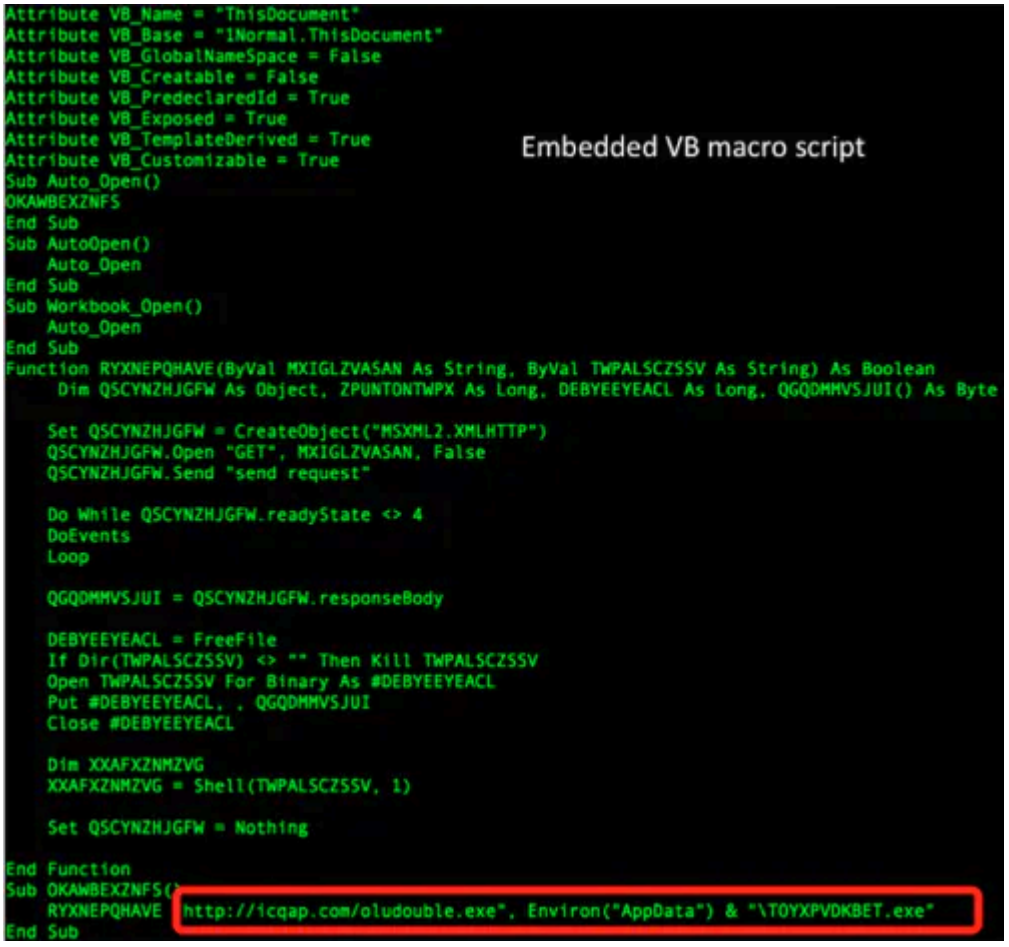

Figure 4: Embedded VBA macro script

This script operates as a downloader, pulling a second stage payload from the following URL (Note: at the time of this post, the referenced domain was no longer active):

hxxp://icqap.com/oludouble.exe

Installation and Persistence

Static analysis of the "oludouble.exe" binary is summarized in Figure 5.

```
################################################################################
Record 0
################################################################################

Meta-data
================================================================================
File:     oludouble.exe
Size:     1084416 bytes
Type:     PE32 executable for MS Windows (GUI) Intel 80386 32-bit Mono/.Net assembly
MD5:      c401d078bb6087bdacf833bcd8b0c68e
SHA1:     aad6288d45ef98a1297d15183a93c5460be15326
ssdeep:   24576:bTgrI2xXOR6nwORBc4hEI8Bl/t9iB8lIs:AJxfTB9hEDBNt91x
Date:     0x5327DFAB [Tue Mar 18 05:54:51 2014 UTC]
EP:       0x50a2be .text 0/3
CRC:      Claimed: 0x0, Actual: 0x110e24 [SUSPICIOUS]

Resource entries
================================================================================
Name               RVA      Size     Lang         Sublang                Type
--------------------------------------------------------------------------------
RT_VERSION         0x10c0a0 0x13c    LANG_POLISH  SUBLANG_DEFAULT        data
RT_MANIFEST        0x10c1dc 0x1ea    LANG_NEUTRAL SUBLANG_NEUTRAL        XML document text

Sections
================================================================================
Name       VirtAddr     VirtSize     RawSize      Entropy
--------------------------------------------------------------------------------
.text      0x2000       0x1082c4     0x108400     7.983708     [SUSPICIOUS]
.rsrc      0x10c000     0x400        0x400        4.425072
.reloc     0x10e000     0xc          0x200        0.101910     [SUSPICIOUS]

Version info
================================================================================
FileDescription: Windows Accelerato Plugin.
Translation: 0x0415 0x04e4
```

Figure 5: Static analysis of downloaded second stage malware, oludouble.exe

Once executed, “oludouble.exe” drops two executables (Windows XP paths furnished):

- C:\Documents and Settings\Administrator\Desktop\exchangepre.exe
- C:\Documents and Settings\Administrator\Application Data\Windows Update.exe

Both binaries are exact copies (Figure 6).

```
b6275be58a539ea9548d02ab6229c768  exchangepre.exe
b6275be58a539ea9548d02ab6229c768  Windows Update.exe
```

Figure 6: Files dropped from second stage malware, oludouble.exe

The second stage malware also copies itself to the following directory (Windows XP) and deletes its original file:

C:\Documents and Settings\Administrator\Application Data\Temp.exe

Persistence (enabling the malware to reload after reboot and restart) is achieved through addition of the following registry key, set to the path for the “Windows Update.exe” binary (Figure 7):

HKCU\Software\Microsoft\Windows\CurrentVersion\Run\Windows Update

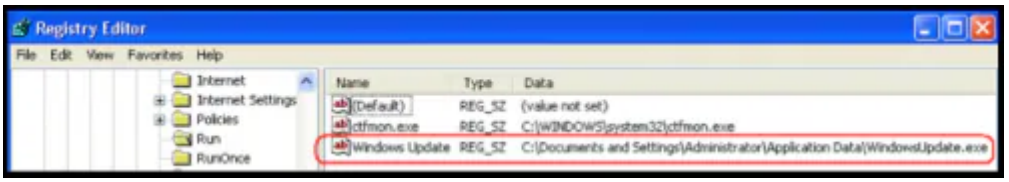

Figure 7: Windows registry modification for persistence

Malware Capabilities

API Calls extracted from “Windows Update.exe” (**b6275be58a539ea9548d02ab6229c768**) hints at associated capabilities (Figure 8).

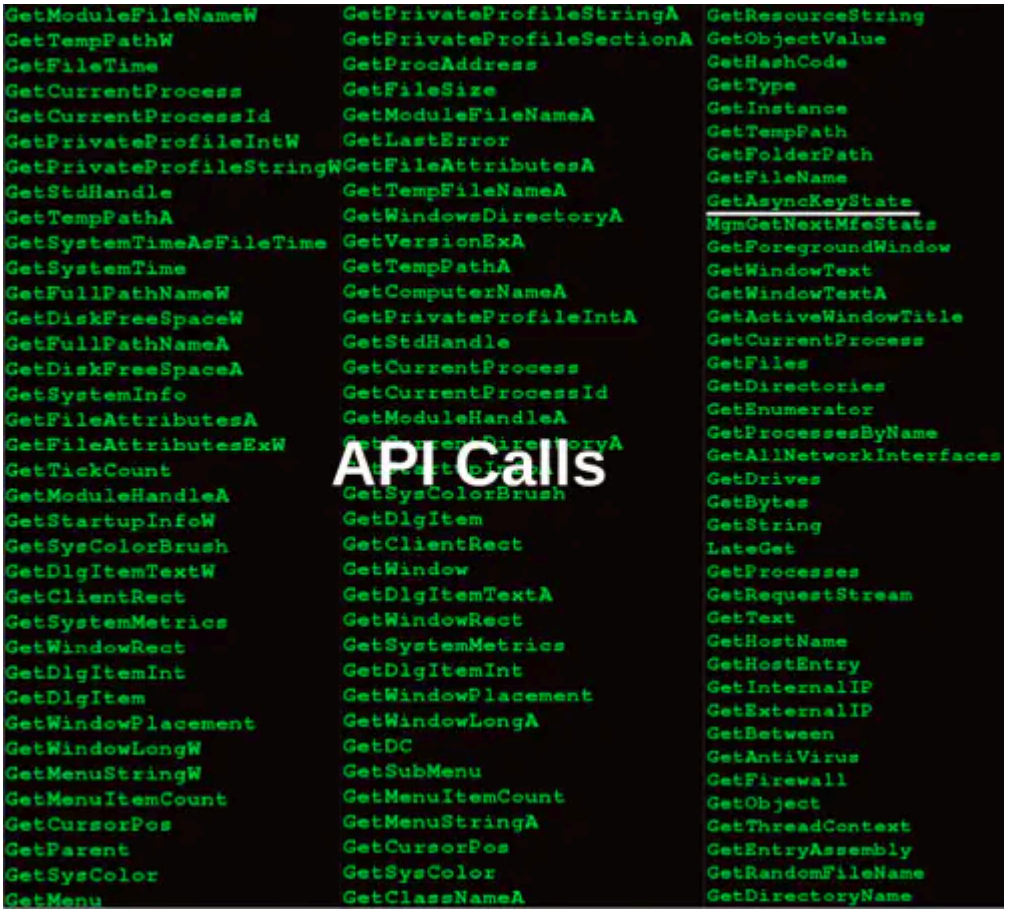

Figure 8: API calls found in “Windows Update.exe” binary

Based on these API calls, the malware appears to support enumeration of a variety of system information. Additionally, the use of “GetAsyncKeyState”, which obtains key press status, could be indicative of keylogging capabilities.

Further investigation and research revealed that this malware leverages the Predator Pain keylogger, a favorite tool of this threat actor. Overall, this malware functions as an information stealer (Infostealer), including capture and exfiltration of the following types of information:

- Website credentials
- Financial information
- Chat session contents
- Email contents

Command and Control (C2)

Once installed, this malware determines its Internet-facing IP address and then establishes a connection with the following domains:

- whatismyipaddress.com
- www.myip.ru
- mail[.]rivardxteriaspte.co[.]uk
- ftp[.]rivardxteriaspte.co[.]uk

The first two domains are legitimate public IP verification services. The latter two are C2 servers run by the malicious actor, which use SMTP and FTP communications, respectively.

Attribution

E-mail headers are a valuable source of intelligence when investigating these types of attacks (Figure 9).

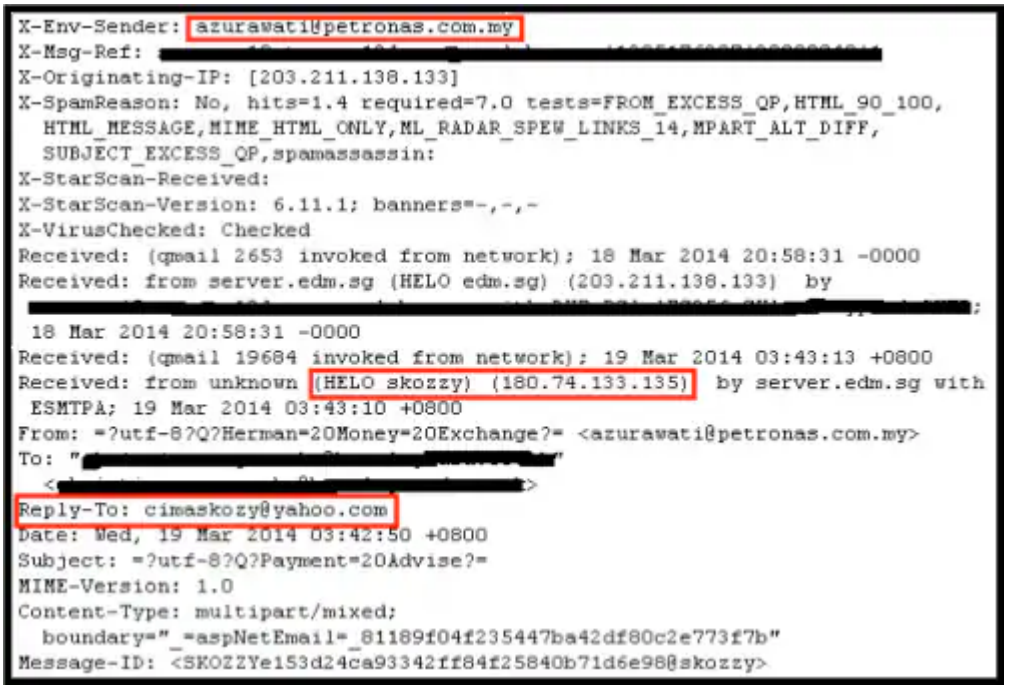

Figure 9: E-mail headers for phishing message

In this example, when the victim opened the phishing message, it appeared to originate from a legitimate organization. However, closer inspection revealed that the sender address was spoofed through the 'X-Env-Sender' header. In an attempt to slide past cursory examination, the malicious actor used an open mail relay, server[.]edm.sg. Another important e-mail header field for this message is 'Reply-To', which contains a valid e-mail for this malicious actor:

cimaskozy(at)yahoo.com

Setting the 'Reply-To' email header field to a valid address is another common threat actor tactic. It supports elicitation activities by that actor should a target respond to the message (i.e., further social engineering). Yet, this technique should also present a red flag to a user, as the initial façade of the originating e-mail address is removed at that point.

Research on the above email address reveals that this actor has been active in the cybercrime underground since at least 2010. Specifically, this actor goes by the handle “Skozzy” and is a known carder, seller of compromised credit card information, and facilitator of related services. Accordingly, we categorize “Skozzy” as primarily a cybercrime actor motivated by financial gain, although roles across nation state, cybercrime, hacktivist and ankle-biter/script kiddies are not mutually exclusive and – in fact – continue to become fuzzier over time.

Figure 10 is a screenshot of a YouTube post by “Skozzy” (skozzy11) from 2010.

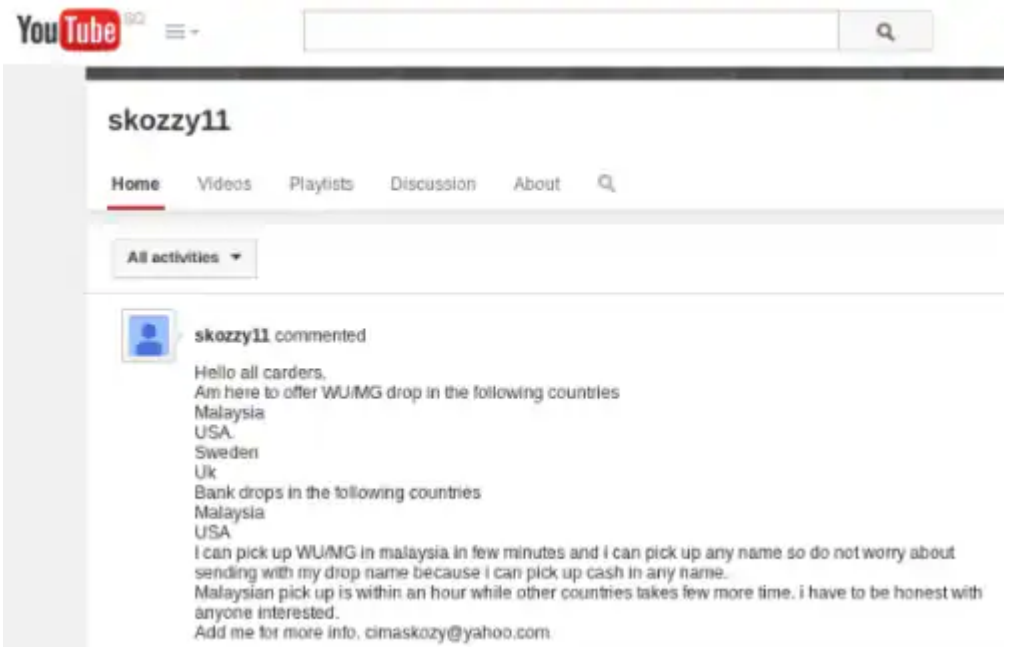

Figure 10: YouTube post from “Skozzy”, 2010

Figure 11 is a screenshot from a Pastebin post, also from 2010.

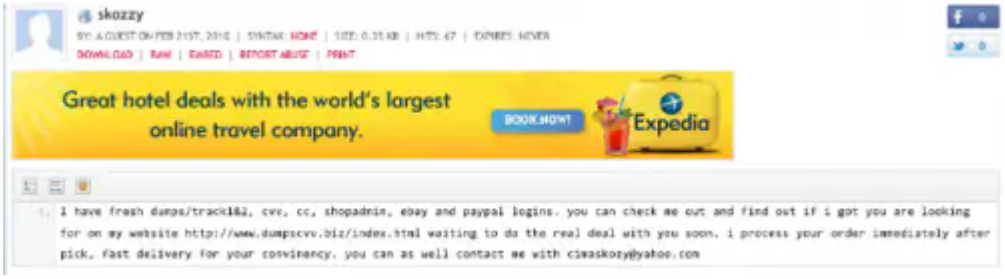

Figure 11: Pastebin post from “Skozzy”, 2010

“Skozzy” is also active on HackForums[.]net and has shared thoughts and experiences related to keylogging tools like Limitless Logger and Predator Pain (Figure 12). Of particular note, the infostealer/keylogger tools that “Skozzy” prefers are able to steal much more than what has been observed so far for this actor.

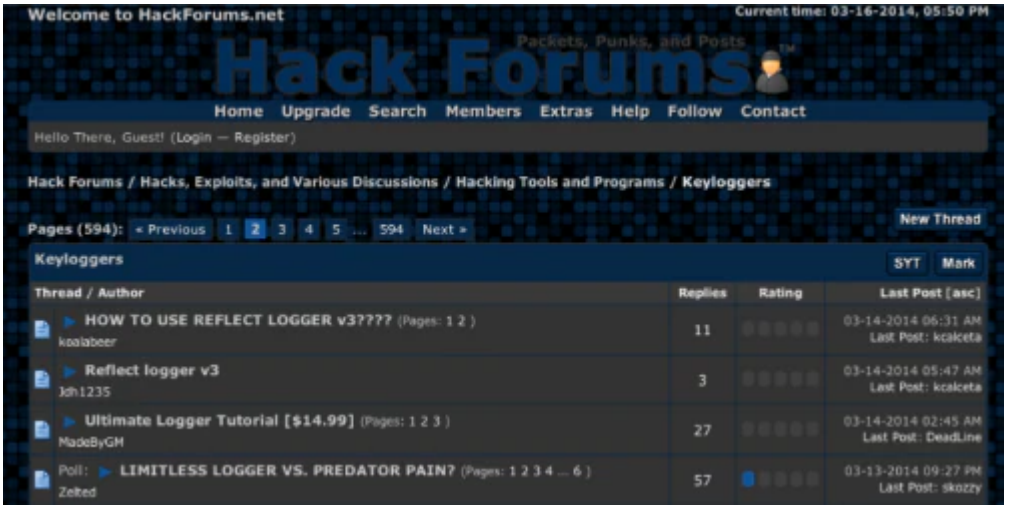

Figure 12: Posts on HackForums[.]net regarding keyloggers

“Skozzy” also shares that Predator Pain is a preferred tool, as it offers great support (Figure 13).

Figure 13: “Skozzy” prefers the Predator Pain keylogger

Deeper analysis and correlation across domains and samples that we believe related to this threat actor will be covered in subsequent blog content.

Conclusion

This case epitomizes how easy it has become these days to steal sensitive information from victims who fall prey to such campaigns. Associated tools can be bought online for less than $100, which often also includes support packages that rival those of mainstream commercial software.

Stolen information can be used for more than standard credit card fraud. The crossover between malicious actor objectives may include opportunistic aspects of cyber espionage, extortion, identity theft, intellectual capital theft, and much more. It is also important to note that none of the major anti-virus (AV) vendors detected this threat at the time it was delivered. The natural gap between creation of these threats and a corresponding signature for their detection by traditional AV remains a sweet spot for successful malicious campaigns. Therefore, it is increasingly important to properly architect and deploy network and endpoint protections to ensure thorough and effective defense of computing and information assets.

The Palo Alto Networks Enterprise Security Platform is a prime example of technology meant to address and minimize the risk associated with emerging threats. Learn more about the platform here.

Source: https://unit42.paloaltonetworks.com/examining-vba-initiated-infostealer-campaign/