2022
PRAGUE

28 - 30 September, 2022 / Prague, Czech Republic

PRILEX: THE PRICEY PRICKLE CREDIT CARD COMPLEX

Kaspersky researchers

Kaspersky

www.virusbulletin.com

ABSTRACT

Prilex is a Brazilian-originated threat actor that has evolved from ATM-focused malware into modular point-of-sale (PoS) malware. The actor was behind one of the largest attacks on ATMs in the country, infecting and jackpotting more than 1,000 ATMs while also cloning over 28,000 credit cards that had been used in these ATMs before the big heist. But the greed of the criminals had no limit – they wanted far more, and they achieved it.

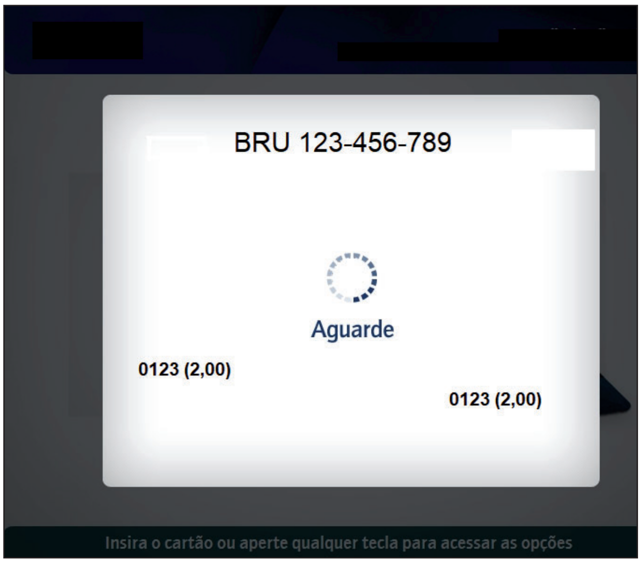
Active since 2014, the group decided in 2016 to give up ATM malware and focus all its attacks on PoS systems, targeting the core of the payments industry. These are criminals with extensive knowledge of the payment market and EFT software and protocols. They quickly adopted malware-as-a-service operations and expanded their reach abroad, creating a modular toolset including backdoors, uploaders and stealers. Since then, we've been tracking their every move, witnessing the damage and big financial losses brought to the payments industry.

Prilex evolved from a simple memory scraper into very advanced and complex malware: dealing directly with the PIN pad hardware protocol instead of using higher level APIs; doing real-time patching of targeted software; hooking operating system libraries; messing with replies, communications and ports; switching from a replay-based attack to be able to generate cryptograms for its GHOST transactions, even from credit cards protected with CHIP and PIN technology.

IT ALL STARTED WITH ATMs IN A CARNIVAL PARTY

During the carnival in 2016, a Brazilian bank realized that its ATMs had suffered an attack in which all funds present in those machines had been stolen. According to reports from law enforcement agencies, the criminals behind the attack were able to infect more than 1,000 ATMs from the same bank in that incident, allowing them to clone 28,000 unique credit cards in Brazil.

During the incident, the attackers did not have physical access to the machines, but they were able to access the bank's network by using a handmade device containing a 4G router and a *Raspberry Pi*. By opening a backdoor to the attacker, they could hijack the institution's wireless connection and target ATMs at will. After obtaining initial network access, the attacker would run a network recognition process to find the IP address of each of the ATMs. With that information to hand, a lateral movement phase would begin, using default *Windows* credentials and then installing custom crafted malware in the desired systems. The backdoor would allow the attacker to empty the ATM socket by launching the malware interface and typing the correct code supplied by the mastermind of the operation: this code was specific to each ATM being defrauded.



[redacted] [redacted]

BRU 123-456-789

Aguarde

0123 (2,00)

0123 (2,00)

Insira o cartão ou aperte qualquer tecla para acessar as opções

Figure 1: ATM infected with Prilex ready to dispense money.

The malware used in the attack was named Prilex and had been developed from scratch using privileged information and advanced knowledge of the ATM network. To control the ATM, Prilex patched legitimate software for jackpotting purposes. In addition to its ability to perform a jackpot, the malware was also able to capture information from the magnetic strips present in the credit and debit cards inserted into the infected ATMs. Afterwards, this valuable information could be used to clone cards and, in addition, steal funds from the bank's clients.

EVOLVING INTO PoS MALWARE

Prilex has evolved from ATM-focused malware into modular point-of-sale malware targeting payment systems developed by Brazilian vendors, the so-called EFT/TEF software [1]. As we described in 2018 [2], there are many similarities between the ATM and PoS versions of the malware. The first Prilex PoS malware was spotted in the wild in October 2016, aside from the two samples that are shown in Figure 2 with a 2010/2011 compilation date, which we believe were the result of an incorrect system time. We also note that in the 2022 branch they've started using Subversion as the malware's version control system.

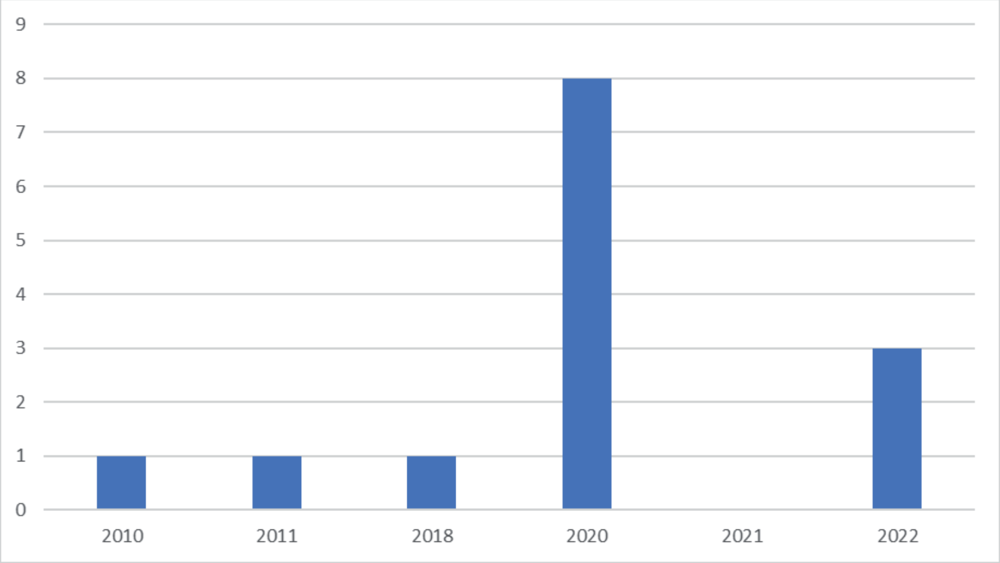


Figure 2: Versions of the Prilex PoS malware: three new versions in 2022.

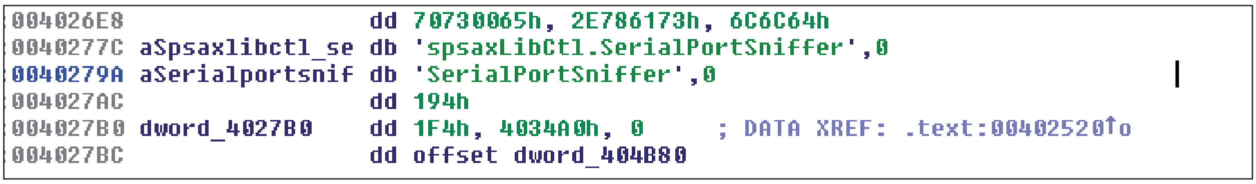
As we see, Prilex was very active in 2020, but suddenly disappeared in 2021, restarting its operation in 2022, releasing three new modifications of the malware.

The PoS version of Prilex is coded in Visual Basic, but the stealer module (described in this article) is in P-Code [3]. In a nutshell, this is an intermediate step between the high-level instructions in your Visual Basic program and the low-level native code executed by your computer's processor. At run time, Visual Basic translates each p-code statement to native code.

A link with the past

Prilex is not the only PoS malware originating in Brazil. We saw a weak link with the old Trojan-Spy.Win32.SPSniffer that we described in 2010 [4]. Both families are able to intercept communications from PIN pads [5], but using different approaches. PIN pads are equipped with hardware and security features to ensure that security keys are erased if someone tries to tamper with the device. In fact, the PIN is encrypted in the device immediately on entry using a variety of encryption schemes and symmetric keys. Most often, this is a triple DES encoder, making it hard to crack the PIN.

But there's a problem: these devices are always connected to a computer via a USB or serial port which communicates with the EFT software. Older and outdated PIN pads devices use old and weak cryptography schemes, making it easy for malware to install a USB or Serial port sniffer, capturing and decrypting the traffic between the PIN pad and the infected system. Sometimes this traffic isn't even encrypted.



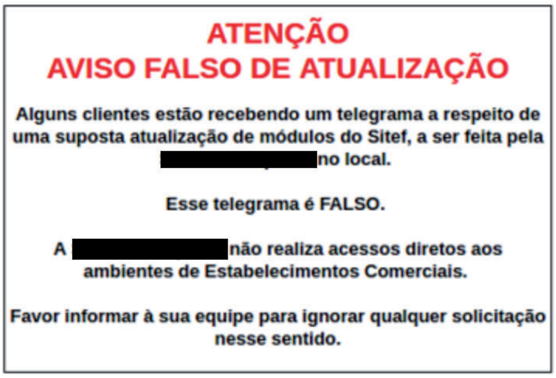
```
004026E8                    dd 70730065h, 2E786173h, 6C6C64h
0040277C aSpsaxlibctl_se db 'spsaxLibCtl.SerialPortSniffer',0
0040279A aSerialportsnif db 'SerialPortSniffer',0
004027AC                    dd 194h
004027B0 dword_4027B0     dd 1F4h, 4034A0h, 0      ; DATA XREF: .text:00402520↑o
004027BC                    dd offset dword_404B80
```

Figure 3: SPSniffer: serial port sniffer allowing capture of non-encrypted traffic.

Besides this trick, the main approach used by Prilex to capture credit card data is a patch in the PoS system libraries, allowing it to collect the data transmitted by the software. The code will look for the location of a particular set of executables and libraries in order to apply the patch, thus overwriting the original code. With the patch in place, the malware collects the data from TRACK2, such as the account number, expiration date and other cardholder information needed to perform fraudulent transactions.

Initial infection vector

Prilex is not widespread malware as it's not distributed through email spam campaigns. It's highly targeted malware and is usually delivered through social engineering, where the business target receives a call from someone telling them a technician needs to update their PoS software. This happens either with the fake technician going personally to the targeted company, or the victim is asked to install *AnyDesk* and provide the fake technician with remote access in order to install the malware.



ATENÇÃO
AVISO FALSO DE ATUALIZAÇÃO

Alguns clientes estão recebendo um telegrama a respeito de uma suposta atualização de módulos do Sitef, a ser feita pela [redacted] no local.

Esse telegrama é FALSO.

A [redacted] não realiza acessos diretos aos ambientes de Estabelecimentos Comerciais.

Favor informar à sua equipe para ignorar qualquer solicitação nesse sentido.

Figure 4: Warning from a PoS vendor about Prilex social engineering attacks.

MESSING WITH EMV STANDARD

Brazil started the migration to EMV cards in 1999 and nowadays almost all cards issued in the country are chip-enabled. A small Java-based application sits inside this chip and can easily be manipulated in order to create a 'golden ticket' card that will be valid in most (if not all) point of sale systems. Having this knowledge has enabled the criminals to update their activities, allowing them to create their own cards featuring this new technology and keeping them 'in the business'.

The first versions of Prilex were able to perform a 'replay attack', where the criminals didn't break the EMV protocol, but took advantage of bad implementations of it. Due to the fact that most payment operators do not perform all validations as required by the EMV standard, the criminals were able to exploit this vulnerability within the process to the advantage of their operation.

In this kind of attack, fraudsters pushed regular magnetic stripe transactions through the card network as EMV purchases, as they were in control of a payment terminal and had the ability to manipulate data fields for transactions put through that terminal. After capturing traffic from a real EMV-based chip card transaction, the thieves could insert stolen card data into the transaction stream, while modifying the merchant and acquirer bank account on the fly.

Local cybercriminals have been performing replay attacks since at least 2014. As pointed out by Brian Krebs [6], a small financial institution in New England battled some $120,000 in fraudulent charges from Brazilian stores in less than two days. The bank managed to block $80,000 of those fraudulent charges, but the bank's processor, which approves incoming transactions when the bank's core systems are offline, let through the other $40,000. All the transactions were debit charges, and all came into *MasterCard*'s network looking to *MasterCard* like chip transactions without a PIN.

It is also worth mentioning the attack against a German bank in 2019 that registered losses of EUR 1.5 million using the same technique [7]. The Prilex gang claimed responsibility for this attack, using the software they are selling in the black market.

To automate such attacks using a PoS terminal, Prilex criminals used tools like this, found by our telemetry in 2020:

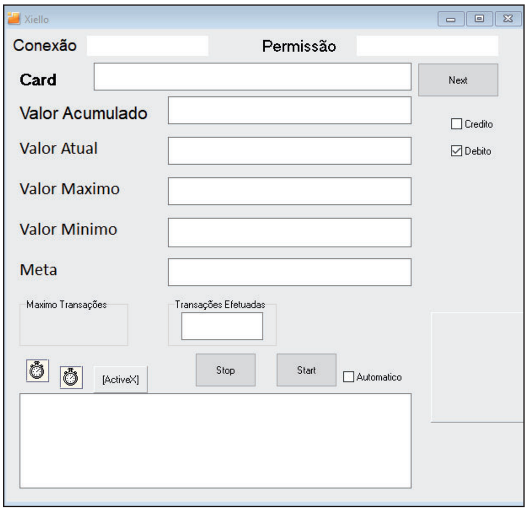


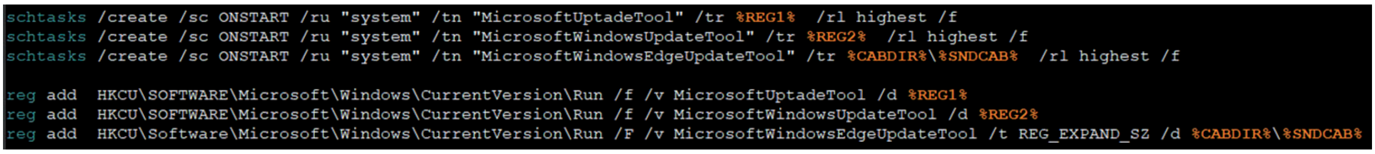
Figure 5: Tool used by Prilex to automate transactions.

As the payment industry and credit card issuers were fixing EMV implementation errors, the replay attack became obsolete and ineffective, pushing the Prilex gang to innovate and adopt other ways to perform credit card fraud.

FROM ‘REPLAY’ TO ‘GHOST’

The newest versions of Prilex show some differences to the previous one regarding the way the attack occurs. They have switched from a replay attack to fraudulent transactions using cryptograms, which are previously generated by the victim card during the store payment process, referred to as GHOST transactions by the authors.

In these attacks, the Prilex samples are installed in the system as RAR SFX executables that extract all required files to the malware directory and execute the installation scripts (VBS files). From the installed files, we can highlight three modules used in the campaign: a backdoor, which is unchanged in this version other than the C2 servers used to communicate; a stealer module; and an uploader module.

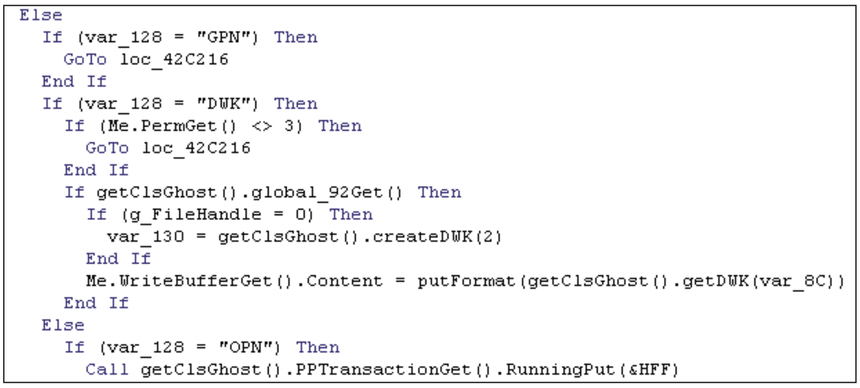


```
schtasks /create /sc ONSTART /ru "system" /tn "MicrosoftUptadeTool" /tr %REG1%  /rl highest /f
schtasks /create /sc ONSTART /ru "system" /tn "MicrosoftWindowsUpdateTool" /tr %REG2%  /rl highest /f
schtasks /create /sc ONSTART /ru "system" /tn "MicrosoftWindowsEdgeUpdateTool" /tr %CABDIR%\%SNDCAB%  /rl highest /f

reg add  HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /f /v MicrosoftUptadeTool /d %REG1%
reg add  HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /f /v MicrosoftWindowsUpdateTool /d %REG2%
reg add  HKCU\Software\Microsoft\Windows\CurrentVersion\Run /F /v MicrosoftWindowsEdgeUpdateTool /t REG_EXPAND_SZ /d %CABDIR%\%SNDCAB%
```

Figure 6: Prilex methods of maintaining persistence.

The stealer module is responsible for intercepting all the communications between the point-of-sale software and the PIN pad used to read the card during the transaction. Once it identifies a running transaction, the malware will capture and modify the content of the transaction in order to be able to capture the card information, as well as to request new EMV cryptograms to the victim’s card.

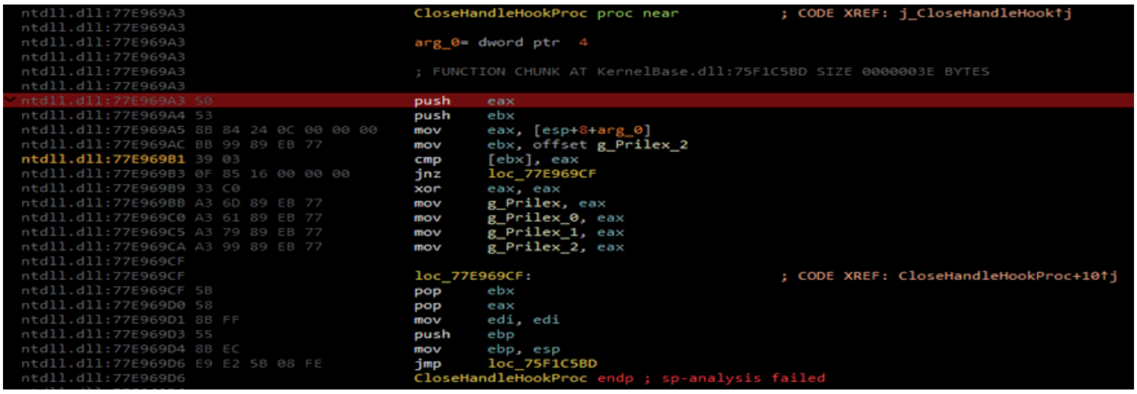


```
Else
  If (var_128 = "GPN") Then
    GoTo loc_42C216
  End If
  If (var_128 = "DWK") Then
    If (Me.PermGet() <> 3) Then
      GoTo loc_42C216
    End If
    If getClsGhost().global_92Get() Then
      If (g_FileHandle = 0) Then
        var_130 = getClsGhost().createDWK(2)
      End If
      Me.WriteBufferGet().Content = putFormat(getClsGhost().getDWK(var_8C))
    End If
  Else
    If (var_128 = "OPN") Then
      Call getClsGhost().PPTransactionGet().RunningPut(&HFF)
```

Figure 7: Method used to parse the PIN pad messages sent/received.

In order to target a specific process the criminal will perform an initial screening of the machine to check if it’s an interesting target, with a minimum number of credit card transactions.

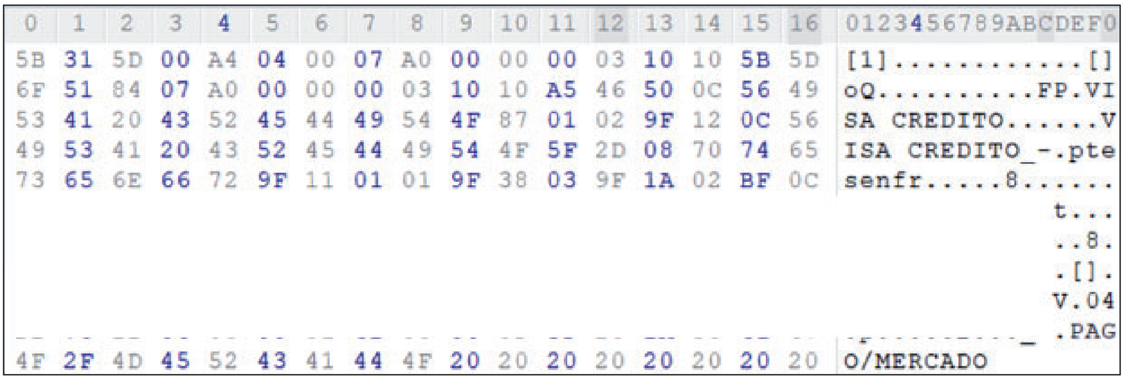
After the process has been identified, the malware will move forward to install the other hooks needed to intercept the transaction information. As the communication between the PoS software and the card reader happens through the COM port, the malware will install a hook to many *Windows* APIs inside the targeted process, aiming to monitor and change data as it needs. But interestingly, instead of allocating memory for the hook procedure, Prilex finds free space within the module’s memory, a technique called code cave [8], making it hard for some security solutions to detect the threat in an infected system.



```
ntdll.dll:77E969A3                         CloseHandleHookProc proc near          ; CODE XREF: j_CloseHandleHook↑j
ntdll.dll:77E969A3
ntdll.dll:77E969A3                         arg_0= dword ptr  4
ntdll.dll:77E969A3
ntdll.dll:77E969A3                         ; FUNCTION CHUNK AT KernelBase.dll:75F1C5BD SIZE 0000003E BYTES
ntdll.dll:77E969A3
ntdll.dll:77E969A3 50                      push    eax
ntdll.dll:77E969A4 53                      push    ebx
ntdll.dll:77E969A5 8B 84 24 0C 00 00 00    mov     eax, [esp+8+arg_0]
ntdll.dll:77E969AC BB 99 89 EB 77          mov     ebx, offset g_Prilex_2
ntdll.dll:77E969B1 39 03                   cmp     [ebx], eax
ntdll.dll:77E969B3 0F 85 16 00 00 00       jnz     loc_77E969CF
ntdll.dll:77E969B9 33 C0                   xor     eax, eax
ntdll.dll:77E969BB A3 6D 89 EB 77          mov     g_Prilex, eax
ntdll.dll:77E969C0 A3 61 89 EB 77          mov     g_Prilex_0, eax
ntdll.dll:77E969C5 A3 79 89 EB 77          mov     g_Prilex_1, eax
ntdll.dll:77E969CA A3 99 89 EB 77          mov     g_Prilex_2, eax
ntdll.dll:77E969CF
ntdll.dll:77E969CF                         loc_77E969CF:                          ; CODE XREF: CloseHandleHookProc+10↑j
ntdll.dll:77E969CF 5B                      pop     ebx
ntdll.dll:77E969D0 58                      pop     eax
ntdll.dll:77E969D1 8B FF                   mov     edi, edi
ntdll.dll:77E969D3 55                      push    ebp
ntdll.dll:77E969D4 8B EC                   mov     ebp, esp
ntdll.dll:77E969D6 E9 E2 5B 08 FE          jmp     loc_75F1C5BD
ntdll.dll:77E969D6                         CloseHandleHookProc endp ; sp-analysis failed
```

Figure 8: Hook code into CloseHandle.

All the captured information from the transaction is saved to an encrypted file placed in a directory previously set in the malware configuration. Those files will later be sent to the malware C2 server and allow the data to be sent through a fraudulent PoS device.



```
0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16  0123456789ABCDEF0
5B 31 5D 00 A4 04 00 07 A0 00 00 00 03 10 10 5B 5D  [1]............[]
6F 51 84 07 A0 00 00 00 03 10 10 A5 46 50 0C 56 49  oQ..........FP.VI
53 41 20 43 52 45 44 49 54 4F 87 01 02 9F 12 0C 56  SA CREDITO......V
49 53 41 20 43 52 45 44 49 54 4F 5F 2D 08 70 74 65  ISA CREDITO_-.pte
73 65 6E 66 72 9F 11 01 01 9F 38 03 9F 1A 02 BF 0C  senfr.....8......
                                                                 t...
                                                                 ..8.
                                                                 .[].
                                                                 V.04
                                                                 .PAG
4F 2F 4D 45 52 43 41 44 4F 20 20 20 20 20 20 20 20  O/MERCADO
```

Figure 9: Captured credit card data that will later be sent to the operator server.

The previous version monitored the transaction in order to get the cryptogram [9], generated by the card for the original transaction, and then to perform a replay attack using the collected cryptogram. In this case, the cryptogram has the same ATC (Application Transaction Counter), allowing the fraudulent transaction to be identified by the reuse of the ATC as well as the date inside the cryptogram – the latter did not match the date when it was submitted, as the fraudulent transactions were submitted at a later point in time.

These cryptograms are then used in a fraudulent transaction through one of the cybercriminal tools in which the output log can be seen below:

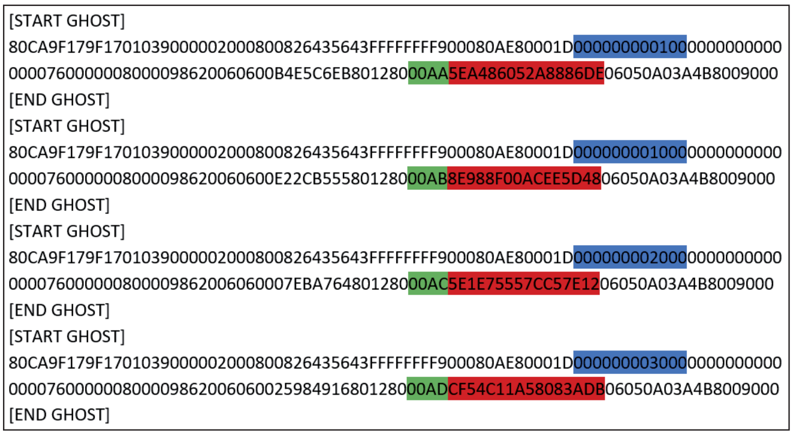
```
[START GHOST]                         _
80CA9F17                               |
9F1701039000                           |
002000800826435643FFFFFFFF             | Check PIN
9000                                  _|

80AE80001D000000000100000000000000007600000080000986200606 00B4E5C6EB -> Generate AC
80128000AA5EA486052A8886DE06050A03A4B8009000 -> Generated ARQC
[END GHOST]
```

The table above shows the data that is collected from the malware. It contains the generated Authorization Request Cryptogram (ARQC) that was generated by the card and should now be approved by the card issuer. After dissecting the response (80128000AA5EA486052A8886DE06050A03A4B8009000) we have the following information:

Data	Field details
80	
12	Size of the response: 18 bytes
80	Cryptogram Information Data: ARQC (Authorization Request Cryptogram) – go and ask the issuer
00AA	ATC – Application Transaction Counter
5EA486052A8886DE	Application Cryptogram
06050A03A4B800	Issuer Application Data
9000	Response OK

Multiple application cryptograms are applied to the card, where the amount of the transaction (blue), ATC (green) and the generated cryptogram (red) changes for each transaction.



```
[START GHOST]
80CA9F179F17010390000020008008264356 43FFFFFFFF900080AE80001D0000000001000000000000
00007600000080000986200606 00B4E5C6EB80128000AA5EA486052A8886DE06050A03A4B8009000
[END GHOST]
[START GHOST]
80CA9F179F170103900000200080082643564 3FFFFFFFF900080AE80001D0000000010000000000000
000076000000800009862006060 0E22CB55580128000AB8E988F00ACEE5D4806050A03A4B8009000
[END GHOST]
[START GHOST]
80CA9F179F170103900000200080082643564 3FFFFFFFF900080AE80001D0000000020000000000000
0000760000008000098620060600 07EBA76480128000AC5E1E75557CC57E1206050A03A4B8009000
[END GHOST]
[START GHOST]
80CA9F179F170103900000200080082643564 3FFFFFFFF900080AE80001D0000000030000000000000
00007600000080000986200606002598491680128000ADCF54C11A58083ADB06050A03A4B8009000
[END GHOST]
```

Figure 10 shows the entire Prilex scheme in a nutshell.

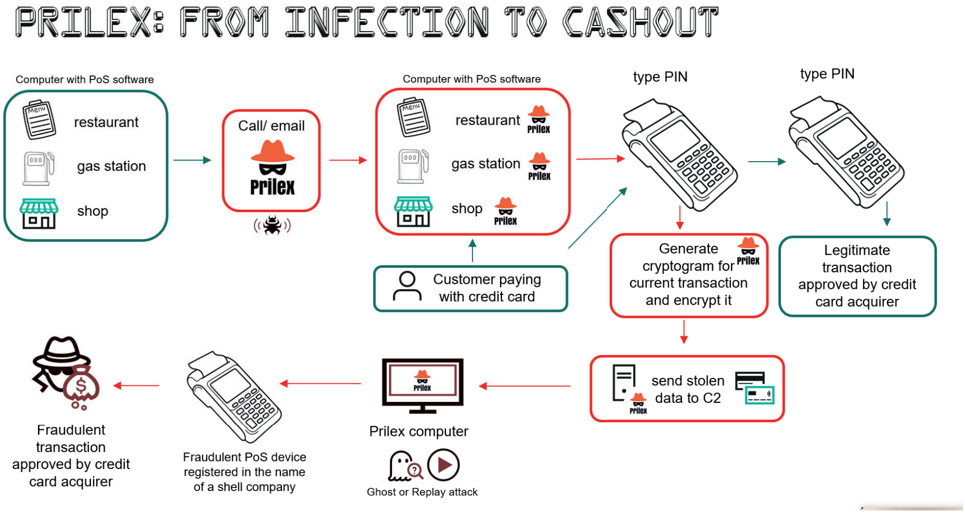
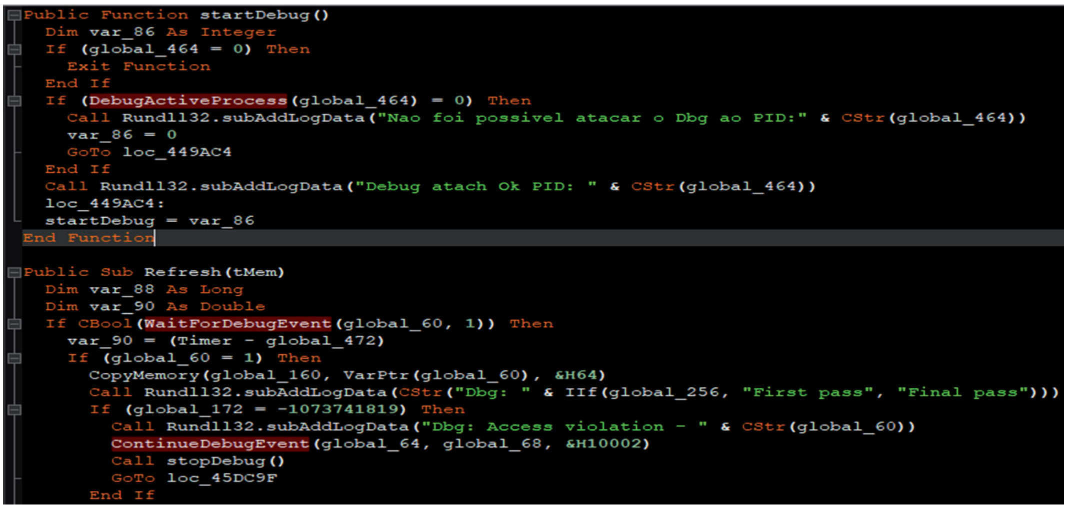

PRILEX: FROM INFECTION TO CASHOUT
Computer with PoS software
restaurant
gas station
shop
Call/ email
Prilex
Computer with PoS software
restaurant
Prilex
gas station
Prilex
shop
Prilex
Customer paying with credit card
type PIN
type PIN
Generate cryptogram for current transaction and encrypt it
Prilex
Legitimate transaction approved by credit card acquirer
send stolen data to C2
Prilex
Prilex
Prilex computer
Ghost or Replay attack
Fraudulent PoS device registered in the name of a shell company
Fraudulent transaction approved by credit card acquirer

Figure 10: Prilex: from the infection to the cashout.

Backdoor module

The backdoor has many commands. Aside from the memory scanning ones that are common to memory scrappers, Prilex also features a command to debug a process and peek on its memory. It's highly likely that this was used to understand target software behaviour and perform adjustments to the malware or environment to perform fraudulent transactions. Older versions of Prilex used to perform patching on specific software libraries, while newer samples do not rely on specific software anymore and instead hook *Windows* APIs to perform their job.



```
Public Function startDebug()
  Dim var_86 As Integer
  If (global_464 = 0) Then
    Exit Function
  End If
  If (DebugActiveProcess(global_464) = 0) Then
    Call Rundll32.subAddLogData("Nao foi possivel atacar o Dbg ao PID:" & CStr(global_464))
    var_86 = 0
    GoTo loc_449AC4
  End If
  Call Rundll32.subAddLogData("Debug atach Ok PID: " & CStr(global_464))
  loc_449AC4:
  startDebug = var_86
End Function

Public Sub Refresh(tMem)
  Dim var_88 As Long
  Dim var_90 As Double
  If CBool(WaitForDebugEvent(global_60, 1)) Then
    var_90 = (Timer - global_472)
    If (global_60 = 1) Then
      CopyMemory(global_160, VarPtr(global_60), &H64)
      Call Rundll32.subAddLogData(CStr("Dbg: " & IIf(global_256, "First pass", "Final pass")))
      If (global_172 = -1073741819) Then
        Call Rundll32.subAddLogData("Dbg: Access violation - " & CStr(global_60))
        ContinueDebugEvent(global_64, global_68, &H10002)
        Call stopDebug()
        GoTo loc_45DC9F
      End If
```

Figure 11: The Prilex debugger.

List of commands:

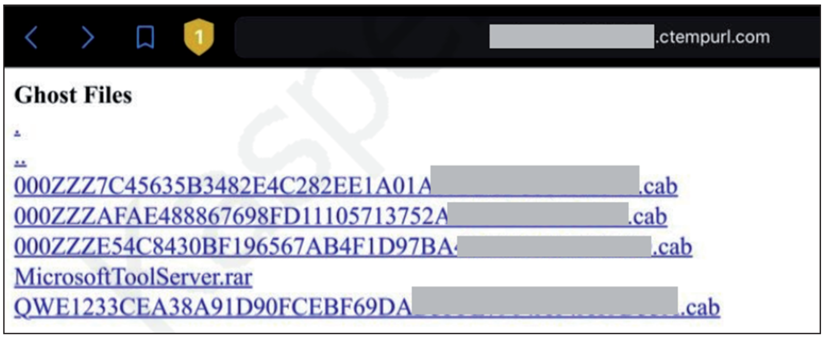
Reboot, SendKeys, ShowForm, Inject, UnInject, HideForm, Recursos, GetZip, SetStartup, PausaProcesso, LiberaProcesso, Debug, SendSnapShot, GetStartup, CapRegion, CapFerro, KillProcess, Shell, Process, GetModules, GetConfig, StartSendScreen, StopSendScreen, ReLogin, StartScan, GetKey, SetConfig, RefreshScreen, Download, TakeRegions, Enviar Arquivo, ScanProcessStart, ScanProcessStop, StartRegiao, StopRegiao, StartDownload, StopDownload.

Uploader module

This module is responsible for checking the directory specified in the CABPATH parameter in the configuration file and sending all CAB files generated from the stolen transactions to the server; these files are sent through an HTTP POST request. The endpoint used by the module is also mentioned in the uploader configuration file.

```
[SNDCAB]
CABHOST=C2
CABPORT=80
CABPAGE=/upload.php
CABPATH=c:\cab
```

The usage of such a module indicates a change in the structure of the group's operation, since in the previous version the collected information was sent to a server whose address is hard coded into the stealer code, and used the same protocol as the backdoor. This uploader allows the operator to set the endpoint for the collected information based on the configuration file; based on the samples analysed, it is possible to see a different infrastructure involved in the process.



[redacted].ctempurl.com

Ghost Files

.

..

000ZZZ7C45635B3482E4C282EE1A01A[redacted].cab

000ZZZAFAE488867698FD11105713752A[redacted].cab

000ZZZE54C8430BF196567AB4F1D97BA[redacted].cab

MicrosoftToolServer.rar

QWE1233CEA38A91D90FCEBF69DA[redacted].cab

Figure 12: Captured data stored in the uploader C2.

MALWARE-AS-A-SERVICE

In 2019, a website posing as the Prilex group appeared, offering its malware packages for sale. But we have low confidence about the ownership of this website: it could be a copycat, trying to impersonate the group to get some money using the fame the group achieved over the years.

The website is still up and running today.

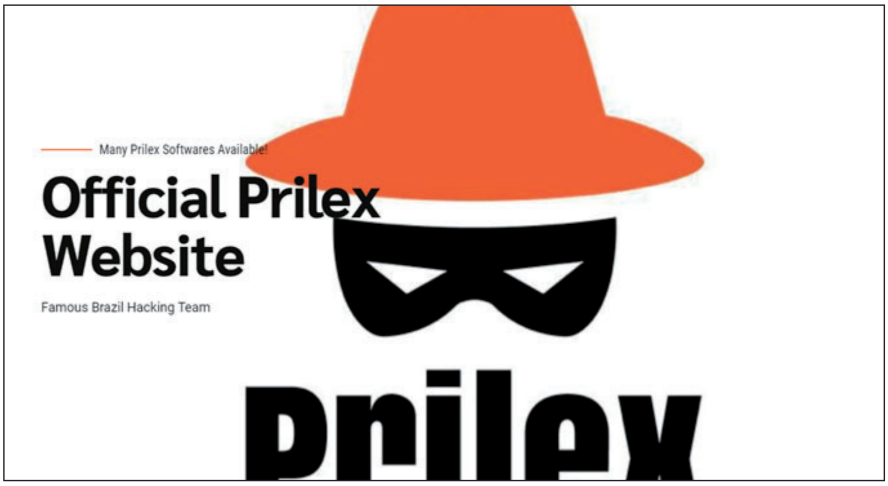
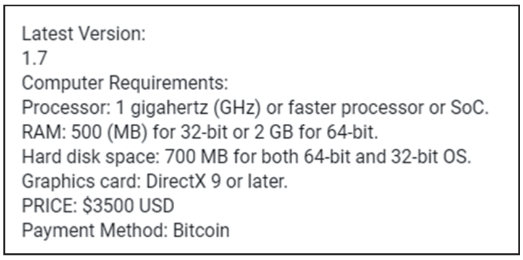


Figure 13: Website purporting to be the official Prilex website.

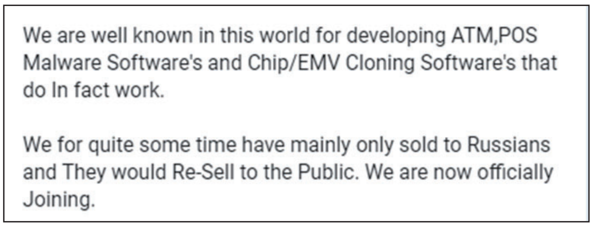
Where the site sells the supposed Prilex PoS kit, the cost is USD3,500, as shown in Figure 14.



Latest Version:
1.7
Computer Requirements:
Processor: 1 gigahertz (GHz) or faster processor or SoC.
RAM: 500 (MB) for 32-bit or 2 GB for 64-bit.
Hard disk space: 700 MB for both 64-bit and 32-bit OS.
Graphics card: DirectX 9 or later.
PRICE: $3500 USD
Payment Method: Bitcoin

Figure 14: The Prilex PoS kit is offered at USD 3,500.

It is also stated on the website that the group has worked with Russian cybercriminals in the past, which is something else we are unable to confirm.



We are well known in this world for developing ATM,POS Malware Software's and Chip/EMV Cloning Software's that do In fact work.

We for quite some time have mainly only sold to Russians and They would Re-Sell to the Public. We are now officially Joining.

Figure 15: The website claims the group has worked with Russian cybercriminals.

Anyway, the fact that the Prilex group is controlling its new versions using Subversion is a clear sign it is working with partners, using a malware-as-a-service model.

CONCLUSIONS

The Prilex group has demonstrated a high level of knowledge related to credit and debit card transactions, as well as an understanding of the way the software used to process payments works. This allows the attackers to keep updating their tools in order to find a way to circumvent the authorization policies, allowing them to perform attacks.

During its years of activity, the group has changed its attack method a lot. However, it has always abused processes related to PoS software in order to be able to intercept and modify the communication with the PIN pad. Given that, we strongly suggest that PoS software developers implement self-protection techniques [10] in their modules, such as the protection available in our SDK [11], aiming to prevent malicious code from tampering with the transactions managed by those modules. For credit card acquiring and issuers we recommend avoiding ‘security by obscurity’ – don’t belittle the fraudster. All EMV validations must be implemented!

Prilex’s success is the biggest stimulant for the emergence of new families, in a fast-evolving and more complex malware environment with a major impact on the payment chain.

The Prilex family is detected by all *Kaspersky* products as HEUR:Trojan.Win32.Prilex and HEUR:Trojan.Win64.Prilex. More details about the threat, and a full analysis, is available to customers of our Threat Intelligence Reports [12].

REFERENCES

[1] Wikipedia. Transferência eletrônica de fundos. https://pt.wikipedia.org/wiki/Transfer%C3%AAncia_eletr%C3%B4nica_de_fundos.

[2] GREAT. Goodfellas, the Brazilian carding scene is after you. SecureList. March 2018. https://securelist.com/goodfellas-the-brazilian-carding-scene-is-after-you/84263/.

[3] BrainBell. A Little About P-Code. https://www.brainbell.com/tutors/Visual_Basic/A_Little_About_P_Code.htm.

[4] Assolini, F. The ‘Chupa Cabra’ malware: attacks on payment devices. SecureList. February 2012. https://securelist.com/the-chupa-cabra-malware-attacks-on-payment-devices/32248/.

[5] Wikipedia. PIN pad. https://en.wikipedia.org/wiki/PIN_pad.

[6] Krebs, B. ‘Replay’ Attacks Spoof Chip Card Charges. Krebs on Security. October 2014. https://krebsonsecurity.com/2014/10/replay-attacks-spoof-chip-card-charges/.

[7] Cimpanu, C. German bank loses €1.5 million in mysterious cashout of EMV cards. ZDNET. September 2019. https://www.zdnet.com/article/german-bank-loses-eur1-5-million-in-mysterious-cashout-of-emv-cards/.

[8] Wikipedia. Code cave. https://en.wikipedia.org/wiki/Code_cave.

[9] Wikipedia. Cryptogram. https://en.wikipedia.org/wiki/Cryptogram.

[10] Kaspersky Online Help. Enabling and disabling Self-Defense. https://support.kaspersky.com/KESWin/11/en-us/133797.htm.

[11] Kaspersky. Kaspersky Anti-Virus Software Development Kit. https://www.kaspersky.com/antivirus-sdk.

[12] Kaspersky. Kaspersky Threat Intelligence. https://www.kaspersky.com/enterprise-security/threat-intelligence.