SPEARTIP®

CYBER COUNTERINTELLIGENCE

Outmaneuver Your Adversary™

POINT OF SALE BREACHES

Protecting Your Customer's Credit Card Data

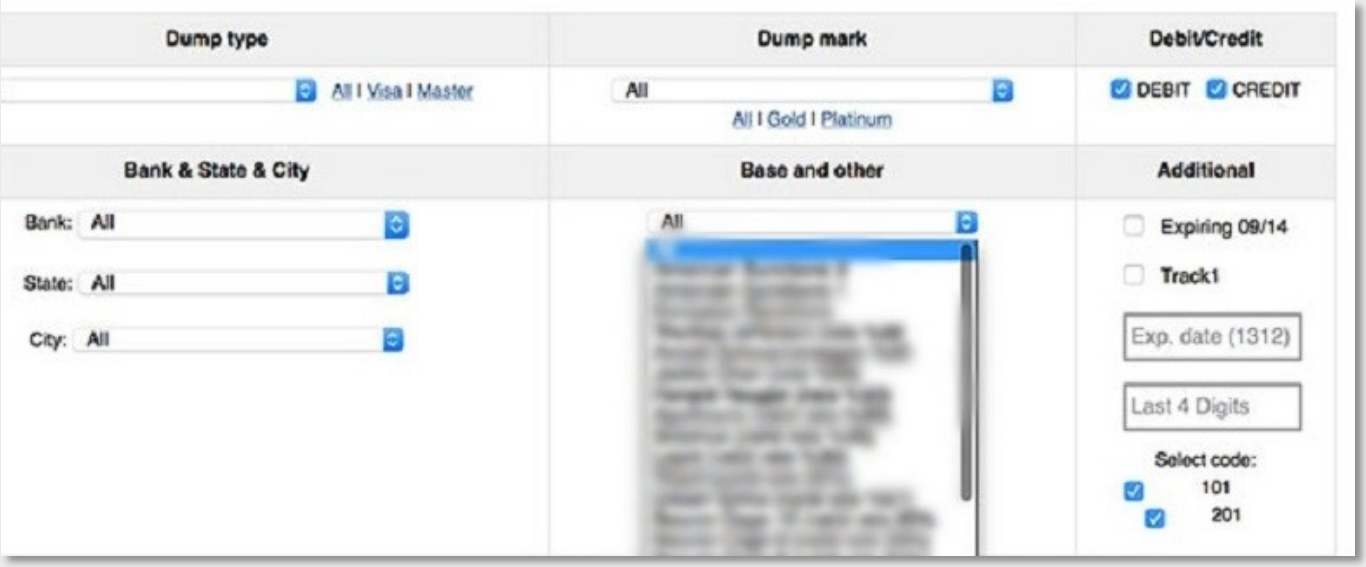
Attackers continue to increase the sophistication of Point of Sale breaches in order to circumvent traditional security measures.

Kris Bleich, CISSP, EnCE, C|EH

While many large organizations struggle with responding to Point of Sale (POS) breaches, smaller companies are increasingly becoming the target of these malware attacks. The retail industry in general continues to be targeted using these methods and attackers continue to increase the sophistication of these attacks in order to circumvent traditional security measures. As a result, the technical expertise required to detect, identify and mitigate these attacks also increases. In addition, the complications introduced by legal exposure from these breaches also weigh heavily on executives and shareholders during the course of response and mitigation when these incidents occur.

In the ongoing cyberwar, POS malware act as "Cyber Thieves" designed to steal credit card and debit card information which can then be used to fund the ongoing cyberwar. POS systems offer cybercriminals the most tempting target because of the sheer number of credit and debit cards processed by these systems. In addition, on POS systems utilizing software encryption, POS malware known as "scrapers" can obtain valuable credit card data directly from active memory of the compromised system before the data is encrypted and stored or transmitted.

Once this data is compromised, it is sold in several different formats on different websites.



Rescator

This data may be stored as "Track 2" data, which would allow the buyer to use the stolen data for online purchases. In some cases, buyers may want to create their own cards and use the stolen information on these cards. In this circumstance, "Track 1" data may be sold. Credit and debit cards contain this information on the magnetic strips on the back of the card.

Credit and debit cards routinely use Track 1 and Track 2 data to process payment data. A third track (Track 3) is not often used. Some cards may not be capable of storing Track 3 data. The formats between Track 1 and 2 differ.

Track 1 data includes the following information:

Format Code- This indicates card type. "B" indicates credit or debit cards
Primary Account Number- Credit or Debit card number
Account Holders Name- First and last name of the account holder
Expiration Date- Formatted in MMYY
Discretionary Data- This may include PINs or other codes used by the card issuer

Track 2 data includes the following information:

Primary Account Number- Credit or Debit card number
Expiration Date- Formatted in MMYY
Discretionary Data- This may include PINs or other codes used by the card issuer

Track 1 data will not exceed 79 characters while Track 2 data will not exceed 40 characters. Both types of data contain enough information to process a payment at a POS system.

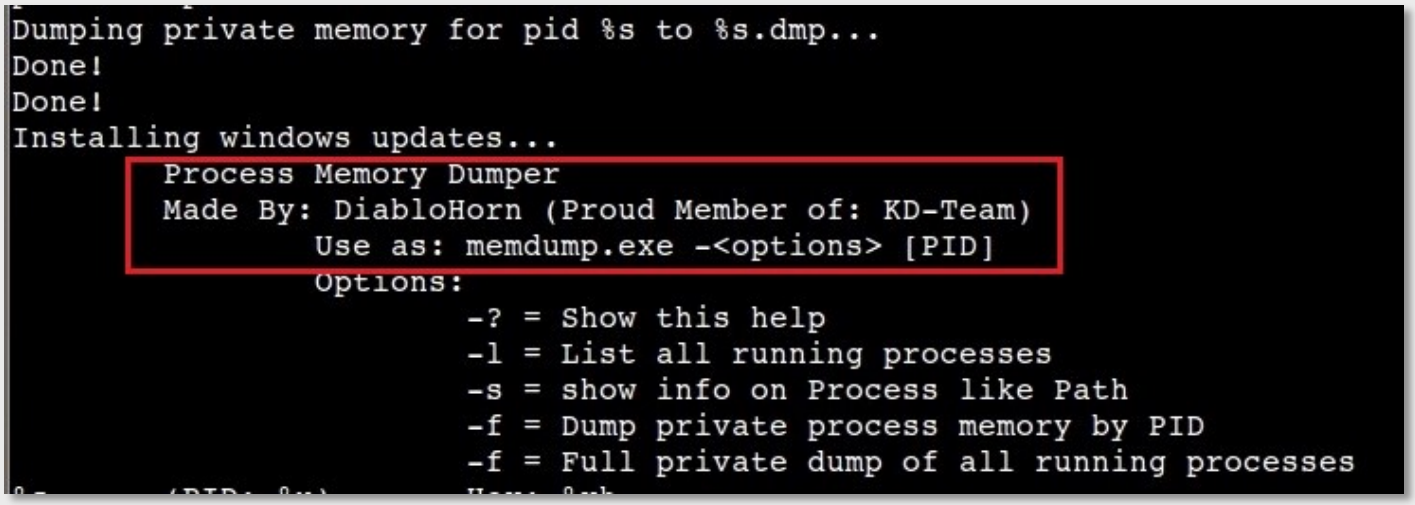
Memory scraping malware identifies patterns which match Track 1 or 2 data within active memory prior to this data being encrypted on the system. Once this data is located, it is typically saved to a file on the system and then exfiltrated at a later time.

An analysis conducted by SpearTip during a recent response identified POS malware operating on multiple systems within an organization's network environment. This malware was operating within active memory on multiple systems despite the deployment and regular use of host-based antivirus software. After responding, SpearTip obtained active memory images of POS systems within the network environment as well as images of the hard drives of several of these systems.

The analysis of active memory located malware running on two systems. The process names suggested that the POS malware involved may be "Backoff."

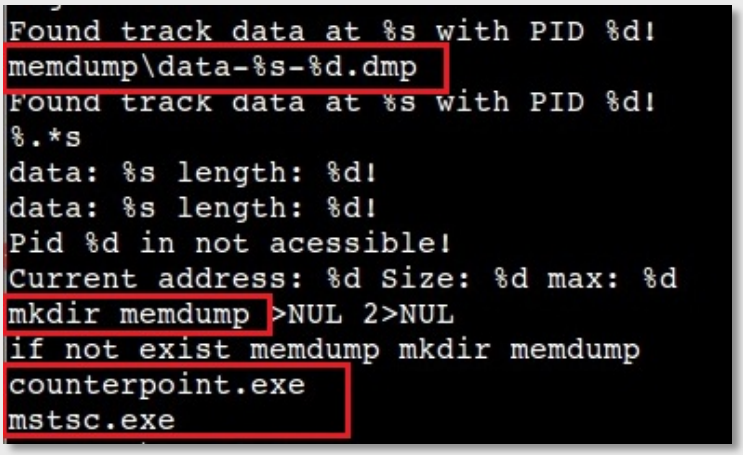
In 2014, the U.S. Secret Service issued an advisory warning businesses of the threat posed by Backoff. In the advisory, the U.S. Secret Service confirmed that numerous organizations had been affected by this particular family of POS malware.

Further analysis of this process located information identifying the functionality of the malware.



The screenshot above indicates the malware may have the capability to dump private memory to a file with the file extension ".dmp". In addition, the string "DiabloHorn (Proud Member of: KD-Team)" is located within strings, indicating the likely author of this particular module.

More information is located within the process in active memory. The command "mkdir memdump" is located indicating that the folder "memdump" is created on the compromised system to store stolen data. The file format for files used to store stolen information is also identified (data-%s-%d.dmp).



The format strings used in this case would result in a filename similar to the following:

Data-stringofdata-1234.dmp

As shown above, the malware also contains strings indicating that it specifically targeted data within the following two processes:

Counterpoint.exe
Mstsc.exe

An analysis of the active memory image for Master File Table entries using “FILE” or “BAAD” signatures located numerous entries for the “memdump” folder identified above. An entry for a file contained in this folder was also located.

```
2014-01-22 20:40:26 UTC+0000 2014-01-22 20:41:07 UTC+0000   2014-01-22 20:41:07 UTC+0000   2014-01-22 20:40
 2014-01-22 20:41:07 UTC+0000   2014-01-22 20:40:29 UTC+0000   memdump\data-counterpoint.exe-7740.dmp.prc
2014-12-18 09:20:50 UTC+0000 -
```

The file name of the file located (data-counterpoint.exe-7740.dmp.prc) is consistent with the format of files constructed by the malware for the storing of stolen data. The file extension .prc is also added, likely to obfuscate and hide the existence of the file. The file appears to have been created on January 22, 2014 at approximately 8:40PM, UTC.

Further analysis of active memory of the affected systems confirmed injected code into the *explorer.exe* process.

```
>>> db(0x03b99ed8, 2048)
0x03b99ed8  46 6f 75 6e 64 20 74 72 61 63 6b 20 64 61 74 61   Found.track.data
0x03b99ee8  20 61 74 20 25 73 20 77 69 74 68 20 50 49 44 20   .at.%s.with.PID.
0x03b99ef8  25 64 21 0a 00 25 2e 2a 73 00 64 61 74 61 3a 20   %d!..%.*s.data:.
0x03b99f08  25 73 20 6c 65 6e 67 74 68 3a 20 25 64 21 0a 00   %s.length:.%d!..
0x03b99f18  64 61 74 61 3a 20 25 73 20 6c 65 6e 67 74 68 3a   data:.%s.length:
0x03b99f28  20 25 64 21 0a 00 50 69 64 20 25 64 20 69 6e 20   .%d!..Pid.%d.in.
0x03b99f38  6e 6f 74 20 61 63 65 73 73 69 62 6c 65 21 0a 00   not.acessible!..
0x03b99f48  43 75 72 72 65 6e 74 20 61 64 64 72 65 73 73 3a   Current.address:
0x03b99f58  20 25 64 20 53 69 7a 65 3a 20 25 64 20 6d 61 78   .%d.Size:.%d.max
0x03b99f68  3a 20 25 64 0a 00 6d 6b 64 69 72 20 6d 65 6d 64   :.%d..mkdir.memd
0x03b99f78  75 6d 70 20 3e 4e 55 4c 20 32 3e 4e 55 4c 00 69   ump.>NUL.2>NUL.i
```

This injected code included the format in which captured data can be identified as well as the command “mkdir memdump”, which is executed by the malware for the storing of captured data (highlighted above). This is consistent with identical strings located within malicious processes (rptsvc32.exe) on the compromised system indicating that the malware targets and compromises the *explorer.exe* process on an affected system.

In addition, further analysis of other processes located a *svchost.exe* process which also contained injected malicious code. The injected code contained the string “DiabloHorn (Proud Member of: KD-Team)” identifying the source of the injected code.

```
Owner: Process svchost.exe Pid 2896
0x03b99bf7  44 75 6d 70 69 6e 67 20 70 72 69 76 61 74 65 20   Dumping.private.
0x03b99c07  6d 65 6d 6f 72 79 20 66 6f 72 20 70 69 64 20 25   memory.for.pid.%
0x03b99c17  73 20 74 6f 20 25 73 2e 64 6d 70 2e 2e 2e 0a 00   s.to.%s.dmp.....
0x03b99c27  44 6f 6e 65 21 0a 00 44 6f 6e 65 21 0a 00 49 6e   Done!..Done!..In
0x03b99c37  73 74 61 6c 6c 69 6e 67 20 77 69 6e 64 6f 77 73   stalling.windows
0x03b99c47  20 75 70 64 61 74 65 73 2e 2e 2e 0a 00 09 50 72    .updates......Pr
0x03b99c57  6f 63 65 73 73 20 4d 65 6d 6f 72 79 20 44 75 6d   ocess.Memory.Dum
0x03b99c67  70 65 72 0a 00 09 4d 61 64 65 20 42 79 3a 20 44   per...Made.By:.D
0x03b99c77  69 61 62 6c 6f 48 6f 72 6e 20 28 50 72 6f 75 64   iabloHorn.(Proud
0x03b99c87  20 4d 65 6d 62 65 72 20 6f 66 3a 20 4b 44 2d 54    .Member.of:.KD-T
0x03b99c97  65 61 6d 29 0a 00 09 09 55 73 65 20 61 73 3a 20   eam)....Use.as:.
0x03b99ca7  6d 65 6d 64 75 6d 70 2e 65 78 65 20 2d 3c 6f 70   memdump.exe.-<op
0x03b99cb7  74 69 6f 6e 73 3e 20 5b 50 49 44 5d 0a 00 09 09   tions>.[PID]....
0x03b99cc7  4f 70 74 69 6f 6e 73 3a 0a 00 09 09 09 2d 3f 20   Options:.....-?.
0x03b99cd7  3d 20 53 68 6f 77 20 74 68 69 73 20 68 65 6c 70   =.Show.this.help
0x03b99ce7  0a 00 09 09 09 2d 6c 20 3d 20 4c 69 73 74 20 61   .....-l.=.List.a
```

The affected *svchost.exe* process (PID 2896) is a child process to *services.exe (PID 496)* and does not show any outward signs of malicious behavior.

```
.. 0xfffffa80036f56f0:svchost.exe                2896    496
.. 0xfffffa8003195b30:svchost.exe                 100    496
```

The analysis of the affected systems located the following URLs within active memory.

https://internetbanking.caixa.gov.br
http://santasalete.sp.gov.br
http://hahadomau.info
http://tributoaofuturo.org.br
https://bankline.itau.com.br
https://www.xiuzhe.com

An analysis of hard drive images of the affected systems was conducted to locate further evidence of the compromise. The analysis located the "memdump" folder created by the malware within the system32 directory (shown below with creation date).

The malicious file *rptsvc32.exe* was also located within the system32 directory. The compile date of the malicious executable is identical to the creation date of the located "memdump" folder.

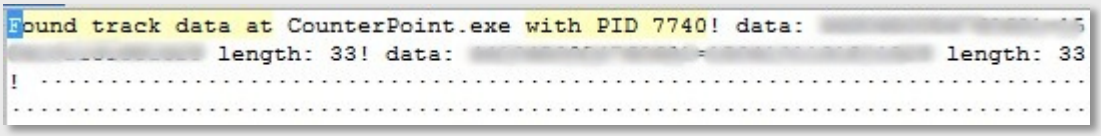
File Name	rptsvc32.exe
File Size	197632 byte
Compile Time	2013-11-27 15:20:24
DLL	No
Sections	8
Hash MD5	56d420a5528b802ee3e07292ee78bb49
Hash SHA1	05b164badf2a521265890ff8643c3d0336eec41c
Packer	Yes
Anti Debug	Yes
Anti VM	No
Directory	Import, Export, Resource, TLS, Relocation

The analysis of the malicious executable located evidence that the malware targeted *counterpoint.exe* processes on the compromised systems. The compile date of *rptsvc32.exe* was identical to the creation date of the "memdump" folder identified on several of the compromised systems, indicating that the executable was likely altered to target *counterpoint.exe* processes prior to deployment. In addition, these changes would likely help the malware evade signature based security mechanisms such as host-based antivirus solutions.

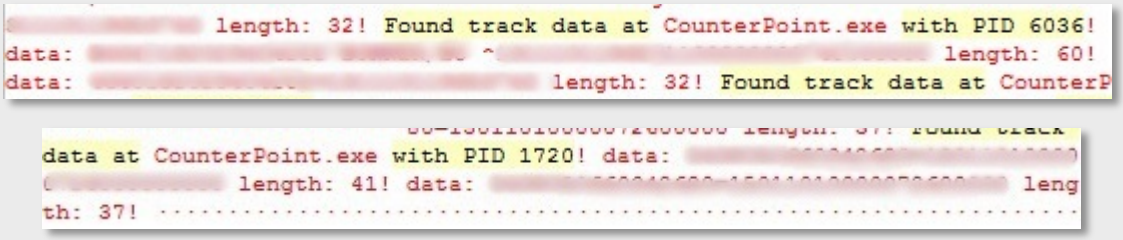
An analysis of the "memdump" folder identified a deleted file "data-counterpoint.exe-7740.dmp.prc" within the directory.

Name	File Created
data-counterpoint.exe-7740.dmp.prc	01/22/14 02:40:26PM

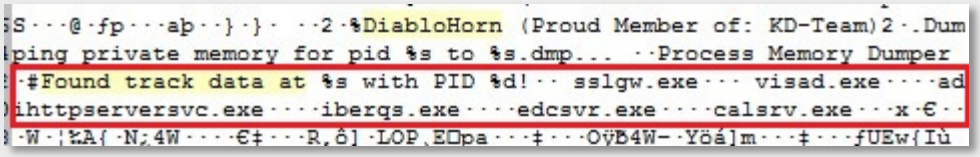
Track 2 data was located in clear text within this file.



The format of the captured Track 2 data is consistent with the format identified within the compromised *explorer.exe* process. Further analysis of the hard drive image located numerous instances of deleted clear text Track 1 and Track 2 data captured by the malware.



In addition, the analysis of the hard drive image of the compromised system identified the "DiabloHorn" text string within unallocated space.



Located text strings (highlighed above) indicate the targeting of the following processes, in addition to *counterpoint.exe* and *mstsc.exe*.

Sslgw.exe
Visad.exe
Adihttpserversvc.exe
Iberqs.exe
Edcsvr.exe
Calsrv.exe

The following Yara rule was developed based on the findings of this analysis and can be used to detect the malicious process as well as malicious code running within legitimate processes.

```
rule RawPOS
{
        meta:
        Description = "RawPOS- rptsvc32 process- appears to be Backoff"
        Author = "Kristopher Bleich, kbleich@speartip.com"

        strings:
        $s1 = "Made By: DiabloHorn (Proud Member of: KD-Team)"
        $s2 = "pid-%s.dmp"
        $s3 = "Dumping private memory for pid %s to %s.dmp..."
        $s4 = "memdump\\data-%s-%d.dmp"
        $s5 = "Found track data at %s with PID %d!"
        $s6 = "mkdir memdump"

        condition:
        all of them
}
```