Fidelis Threat Advisory #1011

"New CDTO: A Sneakernet Trojan Solution"

January 15, 2014

Document Status: FINAL
Last Revised: 2014-01-15

Executive Summary:

The General Dynamics Fidelis Cybersecurity forensics team analyzed several related malware samples that together provide a sophisticated mechanism to gather data from individual computer systems. The malware appears to be part of a system that may be optimized for use by an insider agent and/or for collecting data from disparate networks or air-gapped systems. The malware includes features to clean up after itself by deleting key indicators that it was present.

The malware system apparently includes additional components that have not been identified. These components would potentially perform additional command and control functions and potentially exfiltration from the central host. The sophistication of the malware and the effort involved in its development would indicate that it was developed for a high value target. However, the specific targeting of this malware is not clear at this time. We are concerned that while the malware system was probably developed for a specific target or family of targets, it could be employed with little adaptation against virtually any target.

This threat advisory describes the functionality of the three malware files to include command inputs and the resulting behavior of the malware.

The Fidelis XPS™ advanced threat defense system has been updated with rules to detect various components of this malware system. However, the fact there are still unanswered questions about the components of the malware system and its intended targeting, emphasizes the importance of employing that best practices such as denying use of removable media on sensitive systems and disabling autorun! This is particularly true for systems that are not protected by Fidelis XPS.

Additional reverse engineering and analysis is on-going at this time.

Forensic Analysis Findings:

On 8 Jan 2014, The Fidelis Network Defense and Forensics team received three files: netsat.exe, netui3.dll and winmgt.dll. All three files were 32 bit executable files. Preliminary analysis disclosed netsat.exe would terminate when run if the system date was on or after 21 Jun 2013. The submitted files appeared to represent two parts of a suspected data collection scheme. Essentially, netsat.exe appeared to operate as a master program that infected removable media connected to the system whereon it was running and collected data from infected drives when the drives returned. netsat.exe received commands from an encrypted file stored on the local system. The infection was in the form of a renamed netui3.dll or winmgt.dll file along with an Autorun.inf file set to run the renamed netui3.dll/winmgt.dll when the infected drive was connected to a target host. There could be many iterations of netsat.exe running on enterprise or targeted entity systems. Based on available analysis results, netsat.exe could collect surreptitiously gathered data from any infected drive connected to the system whereon netsat.exe was running, e.g., the infected drive would not have to be processed by the same system whereon it became infected. Data, in the form of files, destined for exfiltration may be obfuscated via a custom XOR operation. The gathered data would ostensibly be exfiltrated via other means.

Some components of the malware's behavior are possibly remarkable. Quickly considering these results in cursory questions reflected as follows:

- Command and Control (C2) appears to be accomplished via providing commands in an encrypted file stored on the local 'master' system (re: netsat.exe). This C2 scheme would seem to dictate:
    - Intruder remote access to the 'master' system
    - Intruder local access to the 'master' system
    - a C2 delivery/retrieval component, such as another piece of code that downloads a C2 file
- Available information precludes determination of the means of exfiltration. netsat.exe's data collection functionality suggested data destined for exfiltration might be collected by the 'master' system. This possibility suggests:
    - Intruder remote access to the 'master' system
    - Intruder local access to the 'master' system
    - An exfiltration mechanism in the form of another piece of code

Detection

Scanning with several select third party malware detection applications resulted in zero detections.

Cursory online research disclosed a file named netui3.dll was possibly submitted to VirSCAN.org on or before 2 Dec 2013. The name netui3.dll appears to have been used for malware in the past and was likely associated with a backdoor. The name may be a play on the name netui2.dll, a legitimate Windows file name.

File System Artifacts

```
File Name:  netsat.exe
File Size:  43520 bytes
MD5:       eb8399483b55f416e48a320d68597d72
SHA1:      8a7183b7ceb30517460b6ab614551ca38e915309
PE Time:    0x5154F7F2 [Fri Mar 29 02:09:54 2013 UTC]
Sections (4):
 Name     Entropy  MD5
 .text    6.37    df1790813aca1265bc475f3891957512
 .rdata   5.19    a598dca4a8fe8ee17941fa60be746d31
```

```
  .data    0.29    b3d1c1a0b1054a082c841ebd1354755f
  .rsrc    3.34    d4b9539426ff130b80e11efec7465acd

File Name:  netui3.dll
File Size:  39424 bytes
MD5:        68aed7b1f171b928913780d5b21f7617
SHA1:       44e711e95311b81d597a7800d96482b873cb8235
PE Time:    0x5152AE99 [Wed Mar 27 08:32:25 2013 UTC]
Sections (3):
  Name     Entropy  MD5
  .text    6.37    b2a939d2ad678201560285287e7dca1d
  .rdata   5.32    2cd54a2d2ada8650c9bd9eae69aef3ca
  .data    0.58    70a79ca0958afad5b7742641b2cff9ea

File Name:  winmgt.dll
File Size:  37888 bytes
MD5:        54e4a15a68cfbb2314d0aaad455fbfce
SHA1:       49531b098049ae52264ae9b398c2cf59ff5379bc
PE Time:    0x50CAEAE4 [Fri Dec 14 09:01:24 2012 UTC]
Sections (3):
  Name     Entropy  MD5
  .text    6.31    6a0f9499f4ca8e0b2e4f09b9126806e6
  .rdata   5.23    e49920b9ebad63f0d95bad505ea8fdf7
  .data    0.59    a583b2c8490a7f0fcaee2f4776e445d8
```

Date Checking

All three submitted files compared the system date and time to hard coded dates upon execution. If the system date was after the hard coded dates, the malware would delete itself and terminate. The following table illustrates the hard coded dates in relation to the affected files' PE dates:

File Name	PE Timestamp Date	Date Checked Within Executable Image
netsat.exe	29 Mar 2013	21 Jun 2013 (Deletes itself and associated files after this date)
netui3.dll	27 Mar 2013	31 May 2013 (Deletes itself and associated files after this date)
winmgt.dll	14 Dec 2012	30 Dec 2012 (Deletes itself and associated files after this date)

Versioning, etc.

The following version information was recorded in the netsat.exe executable:

```
Child Type:        StringFileInfo
Language/Code Page: 1033/1200
CompanyName:       Microsoft Corporation
FileDescription:    Cdto Netware 2.12 Provider
FileVersion:       5.1.2600.0 (xpclient.010817-1148)
InternalName:      NEWCDTO
LegalCopyright:     Microsoft Corporation. All rights reserved.
OriginalFilename:   cdto.dll
ProductName:        Microsoft Corporation. All rights reserved.
ProductVersion:     5, 1, 2600, 0

Child Type:        VarFileInfo
Translation:        1033/1200
```

The Language/Code Page code 1033 denotes U.S. English. This versioning information appears contrived. However, it looks convincing enough to pass cursory inspection, i.e., the format appears legitimate and the appearance does not engender suspicion. Cursory online searches failed to disclose what, if anything Cdto might be associated with.

Scanning disclosed the file contained a function possibly associated with TEAN encryption. This appeared to indicate TEA (Tiny Encryption Algorithm) involvement. (Note: The encryption is used to encrypt commands stored in a file on the local system.)

The submitted files named netui3.dll and winmgt.dll did not have embedded versioning Information like netsat.exe did.

Files and paths

The presence of the following files may indicate netsat.exe, et al, involvement:

CSIDL_WINDOWS\msagent\ netui3.dll
Netui3.dll in any path
CSIDL_WINDOWS\msagent\ netwn.drv
Netwn.drv in any path
CSIDL_MYPICTURES\winsSetup35.exe in any path
Setup23.exe in any path
CSIDL_NETHOOD\Microsoft\Windows\Help\set.fl
CSIDL_LOCAL_APPDATA\Microsoft\Windows\Help\intr
CSIDL_NETHOOD\Microsoft\Windows\Chars\ferf.st
CSIDL_NETHOOD\Microsoft\Windows\Chars\fert.st
CSIDL_LOCAL_APPDATA\Microsoft\Windows\Chars\intr
CSIDL_NETHOOD\Microsoft\Windows\message\
CSIDL_NETHOOD\Microsoft\Windows\Intel\
Act.te in any path

u.t in any path
netwi.drv in any path
~FF325I.tmp or ~FF323D.tmp in the path specified by the TMP. TEMP. or USERPROFILE environment variables or the Windows directory

The presence of the following, specifically on removable media, may indicate netsat.exe, et al, involvement:

Autorun.inf file containing the file name setup35.exe or possibly setup23.exe
RECYCLER\RECYCLED\SYS
RECYCLED\RECYCLED\SYS
RECYCLED\RECYCLED\SYS\desktop.ini (Won't be visible via GUI)
RECYCLER\RECYCLED\SYS\desktop.ini (Won't be visible via GUI)
~disk.ini

Registry

Cursory analysis did not disclose entrenchment data, such as a Registry entry to ensure persistence.

Network Artifacts:

This cursory analysis disclosed no network artifacts specific to the malware's operation. However, evidence of any of the files involved (MD5, strings, file names) traversing the network, e.g., on the move, may be indicative of netsat.exe, et al presence. Given what was revealed during this cursory analysis, finding string and or hash artifacts in SMB traffic seemed the most likely possibility with regard to network detection.

Strings:

netsat.exe

The following interesting strings were noted in the raw netsat.exe file:

```
VMProtect begin
VMProtect end
! Path -- > %s to Added
! Pathlen = %u
AddInit -> ci.DestFile:%s
!ad dri, nD=%d
netui3.dll
netwi.drv
Global\Mtx_Sp_On_PC_1_2_8
```

```
Cdto Netware 2.12 Provider
!Cr Des
!Cr De.i. err=%d
setup35.exe
setup23.exe
act.te
ferf.st
fert.st
netwn.drv
D c p
D c u
SystemPriClass
!Cr ne j
```

Netui3.dll

The following interesting strings were noted in the raw netui3.dll file:

```
set.fl
setup35.exe
setup32.exe
act.te
ferf.st
fert.st
u.t
setup23.exe
%s -wu %s
%s -ws %s
No j n
Mtx_Sp_On_PC_1_2_8
SystemPriClass
```

Winmgt.dll

The following interesting strings were noted in the raw winmgt.dll file:

```
set.fl
setup35.exe
setup32.exe
act.te
ferf.st
fert.st
u.t
```

```
%s -wu %s
No j n
Mtx_Sp_On_PC_1_2_8
\wins.log
SystemPriClass
```

Functionality:

Based on netsat.exe manipulating a file named netui3.dll, the submitted netsat.exe and netui3.dll appeared associated. Versioning is possible. For example, the submitted netui3.dll sample may not match the submitted netsat.exe sample in terms of versioning. However, analysis assumed, for the sake of efficiency, that the submitted netui3.dll and submitted netsat.exe file were associated. The submitted winmgt.dll file appeared very similar to netui3.dll. However, some differences suggested the two files represented disparate versions.

Analysis disclosed date sensitivity built into the submitted files. If the sample was run after a particular date, it would effectively terminate and delete itself:

File Name	**PE Timestamp Date**	**Date Checked Within Executable Image**
netsat.exe	29 Mar 2013	21 Jun 2013 (Deletes itself and associated files after this date)
netui3.dll	27 Mar 2013	31 May 2013 (Deletes itself and associated files after this date)
winmgt.dll	14 Dec 2012	30 Dec 2012 (Deletes itself and associated files after this date)

Analysis disclosed netsat.exe probably serves as the headquarters of malicious activity by:

- Running on a possibly compromised system
- Logging some activity and errors to a file
- Receiving commands via an encrypted file on the local system (possible C2)
- Listening for drive connections
- Infecting connected drives with netui3.dll/winmgt.dll (setup32.exe + Autorun.inf)
- Collecting data gathered by any infected drives, ostensibly upon their return from being connected to other systems

Analysis disclosed netui3.dll/winmgt.dll probably serve as the field units of malicious activity by:

- Collecting information about systems it comes into contact with through connection to the targeted systems with the drive whereon the malware resides
    - *IP*
    - *Platform*
    - *Name*
    - *Version*
    - *Type*
    - *Primary or Backup Domain Controller (PDC or BDC)*
    - *Determines network join status (none, workgroup, domain) via NetGetJoinInformation API*
    - *Detailed OS version*
    - *Running Time*
    - *Computer Name*
    - *User Name*
    - *System Directory*
    - *Current Date and Time*
    - *Locale Information (Country and Language)*
    - *Drive info (total/free size, type, etc.)*
    - *Network Adapter description*
    - *IP Address*
    - *IP Mask*
    - *Gateway IP*
    - *Recursive directory listings*
    - *Enumerates normal user account names*
- Collecting file listings from local and share connected drives
- Discovering and connecting to shared drives visible to the local targeted system
- Copying and writing files to/from drives visible to the local targeted system

Commands

The following commands and their descriptions, listed by executable file, illustrate the submitted malware's functionality:

netsat.exe	
Command	**Description**
Cpd	copies directories and contents
Cpr	copies files with size checking
Der	deletes files and records activity in log
Dir	obtains a directory listing

netsat.exe	
Command	**Description**
Ferry	writes malicious files to a hidden RECYCLED or RECYCLER directory Files: setup35.exe (renamed netui3.dll), Autorun.inf, ~disk.ini (renamed netwi.drv), act.te
Getres	iteratively copies files from RECYCLED/RECYCLER directory on target drive, deletes from source after copy - source is assumed to be drive used to collect data from one or more systems

netui3.dll (setup35.exe)	
Command	**Description**
Cp	copies files from one location to another
Cpu	copies files from one location to another setting copied files as hidden
Cptur	creates a directory and copies file to that directory
Ddr	silently deletes directory (performs an FO_DELETE shell file operation on a directory with the FOF_NOERRORUI, FOF_NOCONFIRMATION, and FOF_SILENT flags set)
Del	deletes a file
Delu	deletes a file after setting attributes to normal
Gd	recursively writes and reads encoded data to/from a directory
Gdir	prints directory listings to ~FF323D.tmp; data gets encoded; original ~FF323D.tmp file is deleted
Gf	writes and reads encoded data to/from a file
Gfover	determines if it has access to a file; may be a temp file creation/rename involved
Gi	collects system related and possibly network related information such as, domains, system information
Ndr	creates a directory
Newend	closes a file that was opened for writing
Newstar	sets normal attributes on a targeted file, deletes the file, opens the same file name as a binary file
Wr	writes a string to a new file opened by the newstar command.
Runb	try to run a targeted executable and then checks for the existence of that file every second for the next 15 minutes as long as it exists

netui3.dll (setup35.exe)	
Command	**Description**
Rune	try to run a targeted executable one time
Slf	generates a targeted file listing, e.g., dir, then copies the files in the list one by one
Srf	copies files in a list one by one
Srmf	uses NetUseAdd to connect to ipc$ share of a target host, creates a listing of files in the c$ - z$ shares of the target host, copies the files to a new location, deletes the share connection added using NetuseAdd
Note: rows highlighted in grey denote a best guess on functionality; more analysis was pending at the time of this report	

Conclusions:

This advisory is based on preliminary information. It is important to note that reverse engineering and analysis of the malware system is still underway. We expect to provide additional data, and some of this information may change as a result of continued research.

However, due to the unique functionality of malware system and its potential for employment against targets beyond the initially intended victim, the network security community should be concerned and track this malware closely.

While we have updated the Fidelis XPS system to detect known components of the malware family, this package reemphasizes the importance of employing good basic network security practices such as denying use of removable media on sensitive systems and disabling autorun!