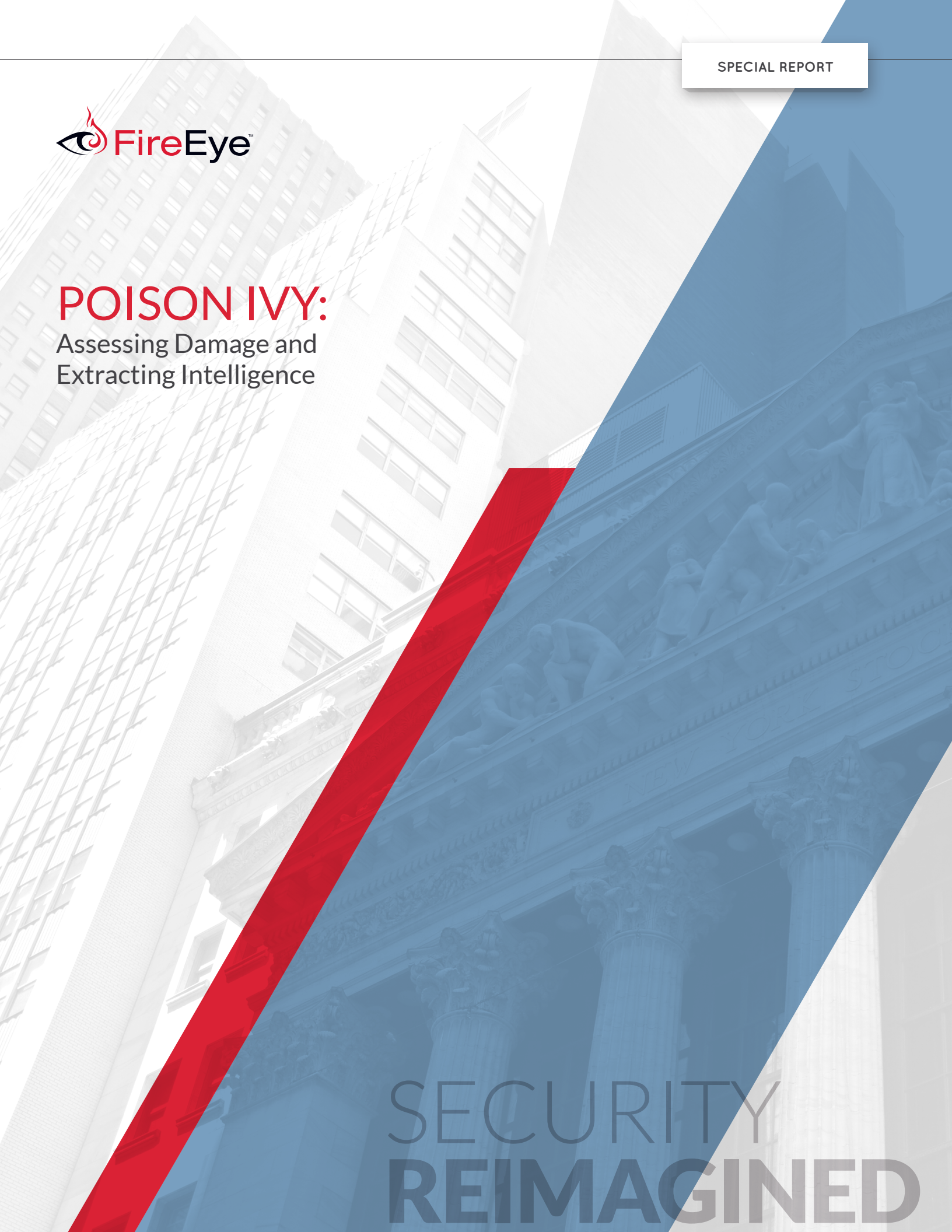
SPECIAL REPORT

FireEye

POISON IVY:

Assessing Damage and Extracting Intelligence



SECURITY
REIMAGINED

CONTENTS

Executive Summary

Remote access tools (RATs) may be the hacker's equivalent of training wheels, as they are often regarded in IT security circles. But dismissing this common breed of malware could be a costly mistake. Despite their reputation as a software toy for novice "script kiddies," RATs remain a linchpin of many sophisticated cyber attacks.

Requiring little technical savvy to use, RATs offer unfettered access to compromised machines. They are deceptively simple—attackers can point and click their way through the target's network to steal data and intellectual property. But they are often delivered as key component of coordinated attacks that use previously unknown (zero-day) software flaws and clever social engineering.

Even as security professionals shrug off the threat, the presence of a RAT may in itself indicate a targeted attack known as an advanced persistent threat (APT). Unlike malware focused on opportunistic cybercrime (typically conducted by botnets of comprised machines), RATs require a live person on the other side of the attack.

This report spotlights Poison Ivy (PIVY), a RAT that remains popular and effective a full eight years after its release, despite its age and familiarity in IT security circles. In conjunction with the study, FireEye® is releasing Calamine, a set of free tools to help organizations detect and examine Poison Ivy infections on their systems.

Poison Ivy has been used in several high-profile malware campaigns, most notoriously, the 2011 compromise of RSA SecurID data. The same year, Poison Ivy powered a coordinated attack dubbed Nitro against chemical makers, government agencies, defense firms and human-rights groups.

Several ongoing cyber attack campaigns use Poison Ivy, including these:

- admin@338—Active since 2008, this campaign mostly targets the financial services industry, though we have also seen activity in the telecom, government, and defense sectors.
- th3bug—First detected in 2009, this campaign targets a number of industries, primarily higher education and healthcare.
- menuPass—Also launched in 2009, this campaign appears to originate from China, targeting U.S. and overseas defense contractors.

Understanding why Poison Ivy remains one of the most widely used RATs is easy. Controlled through a familiar Windows interface, it offers a bevy of handy features: key logging, screen capturing, video capturing, file transfers, password theft, system administration, traffic relaying, and more.

And Poison Ivy is so widely used that security professionals have a harder time tracing attacks that use the RAT to any particular attacker.

We hope to eliminate some of that anonymity with the FireEye Calamine package. The package, which enables organizations to easily monitor Poison Ivy's behavior and communications, includes these components:

- PIVY callback-decoding tool (ChopShop module)
- IVY memory-decoding tool (Immunity Debugger PyCommand script)

ChopShop[1] is a new framework developed by the MITRE Corporation for network-based protocol decoders that enable security professionals to understand actual commands issued by human operators controlling endpoints. The FireEye PIVY module for ChopShop decrypts Poison Ivy network traffic.

PyCommands, meanwhile, are Python scripts that automate tasks for Immunity Debugger, a popular tool for reverse-engineering malware binaries.[2] The FireEye PyCommand script dumps configuration information from a running PIVY process on an infected endpoint, which can provide additional telemetry about the threat actor behind the attack.

FireEye is sharing the Calamine tools with the security community at large under the BSD 2-Clause License[3] for both commercial and non-commercial use worldwide. The tools are available for download at the following locations:

- https://github.com/fireeye/pycommands
- https://github.com/fireeye/chopshop

By tracking the PIVY server activity, security professionals can find these telltale indicators:

- The domains and IPs used for Command and Control (CnC)
- The attacker's PIVY process mutex
- The attacker's PIVY password
- The launcher code used in the malware droppers
- A timeline of malware activity

This report explains how Calamine can connect these and other facets of the attack. This evidence is especially useful when it is correlated with multiple attacks that display the same identifying features.

Combining these nuts-and-bolts details with big-picture intelligence can help profile threat attackers and enhance IT defenses.

Calamine may not stop determined attackers that use Poison Ivy. But it can make their criminal endeavors that much more difficult.

Introduction

Poison Ivy is a remote access tool that is freely available for download from its official web site at www.poisonivy-rat.com. First released in 2005, the tool has gone unchanged since 2008 with v ersion 2.3.2. Poison Ivy includes features common to most Windows-based RATs, including key logging, screen capturing, video capturing, file transfers, system administration, password theft, and traffic relaying.

Poison Ivy's wide availability and easy-to-use features make it a popular choice for all kinds of

[1] ChopShop is available for download at https://github.com/MITRECND/chopshop.

[2] Immunity Debugger is available at http://debugger.immunityinc.com/.

[3] For more information about the BSD 2-Clause License, see the Open Source Initiative's template at http://opensource.org/licenses/BSD-2-Clause.

criminals. But it is probably most notable for its role in many high profile, targeted APT attacks.

These APTs pursue specific targets, using RATs to maintain a persistent presence within the target's network. They move laterally and escalate system privileges to extract sensitive information—whenever the attacker wants to do so.[4,5] Because some RATs used in targeted attacks are widely available, determining whether an attack is part of a broader APT campaign can be difficult. Equally challenging is identifying malicious traffic to determine the attacker's post-compromise activities and assess overall damage—these RATs often encrypt their network communications after the initial exploit.

In 2011, three years after the most recent release of PIVY, attackers used the RAT to compromise security firm RSA and steal data about its SecureID authentication system. That data was subsequently used in other attacks.[6] The RSA attack was linked to Chinese threat actors and described at the time as extremely sophisticated. Exploiting a zero-day vulnerability, the attack delivered PIVY as the payload.[7,8] It was not an isolated incident. The campaign appears to have s tarted in 2010, with many other companies compromised.[9]

PIVY also played a key role in the 2011 campaign known as Nitro that targeted chemical makers, government agencies, defense contractors, and human rights groups.10,11 Still active a year later, the Nitro attackers used a zero-day vulnerability in Java to deploy PIVY in 2012.12 Just recently, PIVY was the payload of a zero-day exploit in Internet Explorer used in what is known as a "strategic web compromise" attack against visitors to a U.S. government website and a variety of others.[13]

RATs require live, direct, real-time human interaction by the APT attacker. This characteristic is distinctly different from crimeware (malware focused on cybercrime), where the criminal can issue commands to their botnet of compromised endpoints whenever they please and set them to work on a common goal such as a spam relay. In contrast, RATs are much more personal and may indicate that you are dealing with a dedicated threat actor that is interested in your organization specifically.

Technical Analysis

Build and implantation

The Poison Ivy builder kit allows attackers to customize and build their own PIVY server, which is delivered as mobile code to a target that has been compromised, typically using social engineering. Once the server executes on a target's endpoint, it connects to a PIVY client installed on the attacker's machine, giving the attacker control of the target system.

The PIVY server code can executed on the target endpoint in a number of ways, depending on how the attacker configured it. In the most common configuration, the PIVY server divides its code into two parts:

[4] Joe Stewart. "The Sin Digoo Affair." February 2012.
[5] Nart Villeneuve. "Trends in Targeted Attacks." October 2011.
[6] eWeek. "Northrop Grumman, L-3 Communications Hacked via Cloned RSA SecurID Tokens." June 2011.
[7] RSA FraudAction Research Labs. "Anatomy of an Attack." April 2011.
[8] CNET. "Attack on RSA used zero-day Flash exploit in Excel." April 2011.
[9] Brian Krebs. "Who Else Was Hit by the RSA Attackers?" October 2011.
[10] Eric Chien and Gavin O'Gorman. "The Nitro Attacks: Stealing Secrets from the Chemical Industry." October 2011.
[11] GovCERTUK Computer Emergency Response Team. "Targeted Email Attack Alert." October 2011.
[12] Symantec. "Java Zero-Day Used in Targeted Attack Campaign." August 2012.
[13] Yichong Lin. "IE Zero Day is Used in DoL Watering Hole Attack." May 2013.

- Initialization and maintenance code
- Networking code

The initialization and maintenance code is injected into the already-running explorer.exe process.

Depending on how the attacker configures it, the networking code launches a hidden Web browser process (the system's default browser) and injects itself into that process. The networking code then remotely downloads (from the attacker's PIVY client as shellcode) the rest of the code and data it needs for its features and functionality. The new code executes on the target's endpoint within the

context of the target process. All of PIVY's global variables, configuration details, and function pointers are stored in a C-style struct (data structure), which is also injected into the target processes in both the PIVY networking code and initialization and maintenance code.

This distinct characteristic has the side effect of having every CALL instruction and global variable address being referenced as an offset to a register when looking at the code's disassembly. The code injected into explorer.exe is peculiar in that, unlike most malware-injected code, this code is injected function by function—each with its own memory region, filling in the proper function pointers in its struct. If the "persistence" PIVY option is enabled, a watchdog thread is also injected into explorer.exe, which automatically restarts the PIVY server process if it is unexpectedly terminated by the target's operating system. PIVY's keylogging function, if enabled, is also injected into explorer.exe.

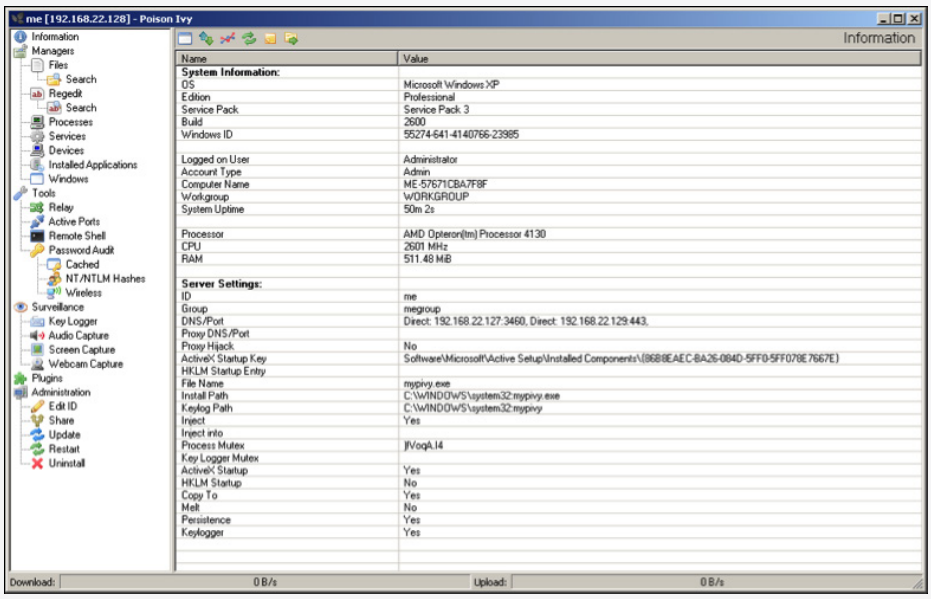


Figure 1: PIVY server configuration details being reported to the PIVY client

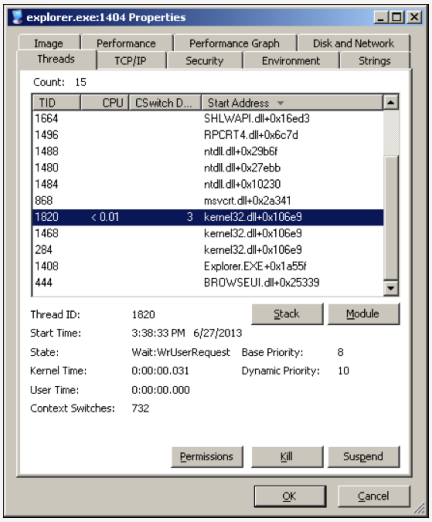
Figure 2: Data and functions referenced as offsets to the struct pointed to by the ESI register

```
debug151:02070009 mov     esi, [ebp+8]
debug151:0207000C lea     eax, [esi+3FBh]
debug151:02070012 push    eax
debug151:02070013 push    0
debug151:02070015 push    0
debug151:02070017 call    dword ptr [esi+85h]
debug151:0207001D mov     [esi+8C5h], eax
debug151:02070023 call    dword ptr [esi+89h]
debug151:02070029 cmp     eax, 0B7h
debug151:0207002E jnz     short loc_2070034
debug151:02070030 leave
debug151:02070031 retn    4
debug151:02070034 ; ---------------------------------------------------------------------------
debug151:02070034
debug151:02070034 loc_2070034:                            ; CODE XREF: debug151:0207002E↑j
debug151:02070034 push    esi
debug151:02070035 lea     eax, [esi+96Bh]
debug151:0207003B push    eax
debug151:0207003C lea     eax, [esi+145h]
debug151:02070042 push    eax
debug151:02070043 call    dword ptr [esi+0FDh]
debug151:02070049 call    loc_2070055
debug151:02070049 ; ---------------------------------------------------------------------------
debug151:0207004E aWs2_32 db 'ws2_32',0
debug151:02070055 ; ---------------------------------------------------------------------------
debug151:02070055
debug151:02070055 loc_2070055:                            ; CODE XREF: debug151:02070049↑p
debug151:02070055 pop     eax
debug151:02070056 push    eax
debug151:02070057 call    dword ptr [esi+9Dh]
debug151:0207005D mov     [esi+0AC3h], eax
```

Figure 3: Injected functions in separate memory regions in explorer.exe

debug151	02070000	02071000	R	W	X	D	.	byte	0000	public	CODE
debug152	02080000	02081000	R	W	X	D	.	byte	0000	public	CODE
debug153	02090000	02091000	R	W	X	D	.	byte	0000	public	CODE
debug154	020A0000	020A1000	R	W	X	D	.	byte	0000	public	CODE
debug155	020B0000	020B1000	R	W	X	D	.	byte	0000	public	CODE
debug156	020C0000	020C1000	R	W	X	D	.	byte	0000	public	CODE
debug157	020D0000	020D1000	R	W	X	D	.	byte	0000	public	CODE
debug158	020E0000	020E1000	R	W	X	D	.	byte	0000	public	CODE
debug159	020F0000	020F1000	R	W	X	D	.	byte	0000	public	CODE
debug160	02100000	02101000	R	W	X	D	.	byte	0000	public	CODE
debug161	02110000	02111000	R	W	X	D	.	byte	0000	public	CODE
debug162	02120000	02121000	R	W	X	D	.	byte	0000	public	CODE

Figure 4: The persistence thread in explorer.exe can easily be killed from Process Explorer



FireEye

Poison Ivy features a complex, custom network protocol over TCP. Most of this communication is encrypted using the Camellia cipher with a 256-bit key.14 The key is derived from a password provided by the attacker when building the PIVY server. The password, "admin" by default, can be provided in plain text or as hex-ASCII. The password is zero-padded to 32 bytes (256 bits). The key is validated at the beginning of the TCP session with a challenge-response algorithm. The PIVY server sends 256 bytes of randomly generated data to the PIVY client which, in turn, encrypts the data using the key and sends it back to the PIVY server for validation. Much of the data sent throughout PIVY's communications is also compressed before encryption using Microsoft's LZNT1 compression algorithm,15 which PIVY utilizes through the Windows RtlCompressBuffer API. The protocol operates by sending encrypted data in chunks that are prepended with the following encrypted 32-byte header:

```
struct PI_chunk_header {
        int command_id;
        int stream_id;
        int padded_chunk_size;
        int chunk_size;
        int decompressed_chunk_size;
        long total_stream_size;
        int padding;
};
```

The PI_chunk_header structure is arranged as follows:
command_id—This member identifies which feature of PIVY the chunked data is related to.
stream_id—This member identifies which stream this flow corresponds to. PIVY's protocol supports sending multiple streams of data simultaneously.
padded_chunk_size—Because Camellia is a 16-byte block cipher, padding is utilized in the headers and in the data chunks.

chunk_size—Chunks are assembled into a stream of data that could be anything, such as a transferred file, shellcode to execute, a screen capture bitmap file, or raw data.
decompressed_chunk_size—If this size is different from the chunk_size, the chunk is compressed using LZNT1.
total_stream_size—This member specifies the total size of the data being sent for the related command_id.
padding—This member specifies the zero padding (up to 32 bytes).

Figure 5: PIVY initial communication protocol

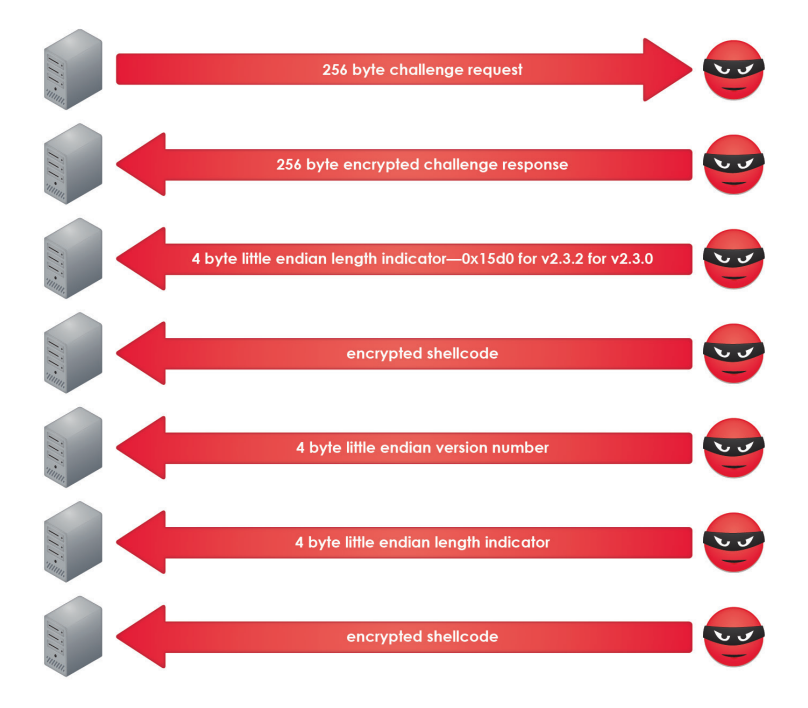
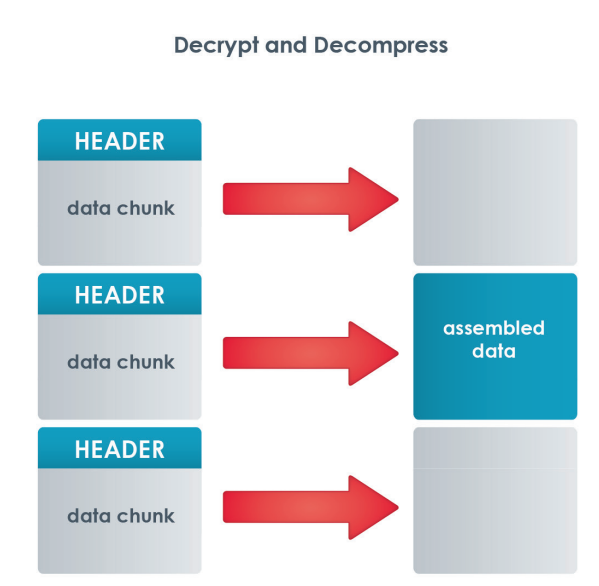


Figure 6: PIVY data chunks with headers



Calamine ChopShop module

The FireEye Poison Ivy decoder checks the beginning of each TCP session for possible PIVY challengeresponse sequences. If found, the module will try to validate the response using one or more passwords supplied as arguments. If no password is supplied, it tries the default "admin" password.

You can supply a single password in either plain-text form or hex-ASCII form. For multiple passwords, you can specify a text file containing line-delimited passwords. If the decoder identifies valid initial PIVY flows based on a supplied password, then the decoder decodes the rest of the relevant flow or flows.

To use the FireEye ChopShop module, you must install CamCrypt, a python wrapper for an open-source implementation of the Camellia encryption library.[16] Most of the features of PIVY are covered to some extent in this module.

Note: If the PIVY flows do not correspond to any supplied password, then decoding fails. Fortunately, you can easily locate the custom PIVY password if you have a compromised endpoint infected with PIVY or a copy of the PIVY server code, as explained in the section "Locating the PIVY Password with the Calamine PyCommand Script."

Calamine ChopShop usage notes

Calamine ChopShop offers the following features and options:

- Files transferred to or from the PIVY server are saved to disk when the -f option is used.
- Webcam, audio, keylog, and single screen captures are saved to disk when the -c option is used.
- The audio captures are saved as raw data that can easily be converted to .wav files u sing a tool such as SoX.[17] The decoder prints the sample rate, channel, and bit data.
- File and registry search details and results are displayed.
- The details of any network relays instantiated are displayed.
- Active port listings are displayed.

This module partially supports decoding listings of Windows files, the registry, services, processes, devices, and installed application listings. During PIVY flow decoding, the module's default output indicates that listing requests have occurred and, when applicable, highlights which key or directory is being listed.

Directory listings are printed, but without file details. When the module is invoked with the -1 option, all returned list data is saved to a file in raw for m, just as it is seen by the PIVY client: a mixtur e of strings and binary data describing those strings. If you are interested in the details of what was listed, running the strings tool on raw file dumps is useful.

If you encounter an unrecognized command or want to extend the functionality of this decoder, the -d option is useful. It prints hex dumps of all the headers and assembled streams in both directions, helping to analyze and build additional parsing functionality.

[16] CamCrypt is available at https://code.google.com/p/camcrypt/.
[17] SoX is available at http://sox.sourceforge.net/.

Locating the PIVY password with the calamine PyCommand script

Many attackers leave the default "admin" password unchanged. In this case, you can start using this decoder immediately. Often, however, the attacker opts for better security by creating a unique password. But if you have access to the PIVY-infected endpoint or to the PIVY server executable, retrieving the password is easy. You can retrieve the password a number of ways, depending on your circumstance and preferences.

If you prefer working with memory dumps, digital forensics expert Andreas Schuster has released a wonderful Volatility plugin for PIVY.[18] Volatility dumps most of PIVY's useful configuration data, including the password, as shown at the Volatility project page (http://code.google.com/p/volatility/source/browse/trunk/contrib/plugins/malware/poisonivy.py?r=2833).

If you have a malware-analysis environment setup, the Calamine PyCommand19 script for Immunity Debugger is quick and simple.[18] The Volatility plugin is available at https://www.volatilesystems.com/default/volatility.[19] Corelan Team. "Starting to write Immunity Debugger PyCommands : my cheatsheet." January 2010.

Follow these steps to use the PyCommand (these steps may vary in some situations):

1. Attach Immunity Debugger to the process PIVY injects into (or to the PIVY process itself if PIVY does not inject).

2. Set breakpoints on the send and connect functions.

3. Continue execution.

4. Wait for the execution to break.

5. Execute until return and step out of the function.

6. Run the PyCommand.

7. Check the logs for the configuration details.

Damage assessment

To effectively assess the damage sustained in an attack, you must reconstruct the attacker's activities. Depending upon the attacker's cleanup efforts, fully reconstructing their activities from host forensics alone may not be possible. But if PIVY network activity is collected, the Calamine ChopShop module can help uncover this information.

In the following example, we set up a test environment and executed commands typically run by attackers after they compromise a system with PIVY and prepare to move laterally. Then

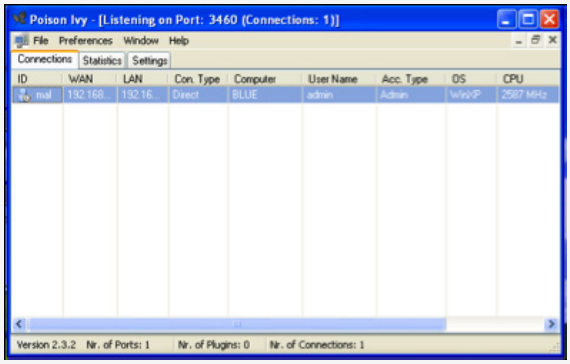
[18] The Volatility plugin is available at https://www.volatilesystems.com/default/volatility.

[19] Corelan Team. "Starting to write Immunity Debugger PyCommands : my cheatsheet." January 2010.

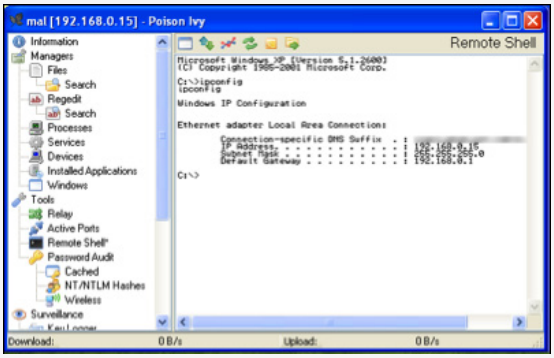
Defender's View

```
Starting ChopShop
Initializing Modules ...
  Initializing module 'poisonivy_23x'
Transferred files will be saved..
Screen/Cam/Audio/Key captures will be saved..
Running Modules ...
[2013-07-03 06:46:29 PDT] Poison Ivy
Version:2.32
[2013-07-03 06:46:30 PDT]*** Host Information***
PI profile ID: mal
IP address: 192.168.0.12
Hostname: BLUE
Windows User: admin
Windows Version: Windows XP
Windows Build: 2600
Service Pack: Service Pack 3
```

Attacker's View




```
[2013-07-03 06:46:36 PDT] *** Shell Session ***
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\>
[2013-07-03 06:46:42 PDT] *** Shell Session ***
ipconfig
[2013-07-03 06:46:43 PDT] *** Shell Session ***
Windows IP Configuration
Connection-specific DNS Suffix . :
  IP Address. . . . . . . . . . . . : 192.168.0.15
  Subnet Mask . . . . . . . . . . . : 255.255.255.0
  Default Gateway . . . . . . . . . : 192.168.0.1
C:\>
```

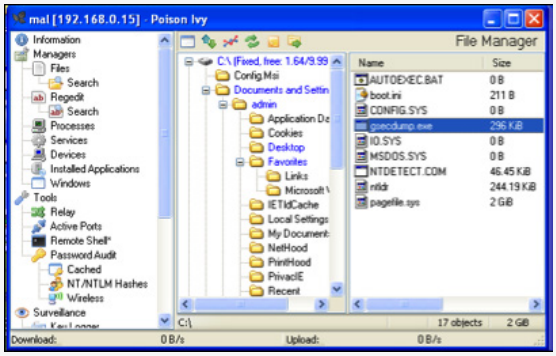


FireEye

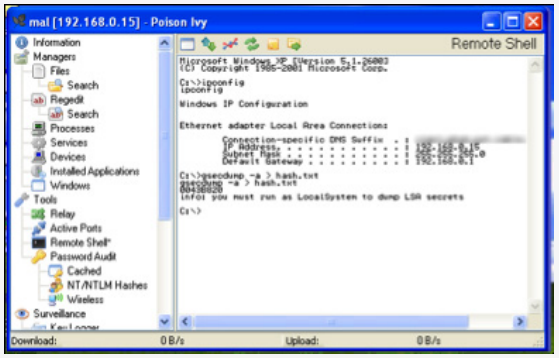
Defender's View

Attacker's View

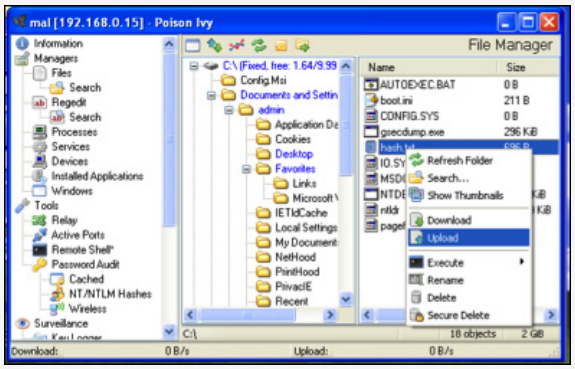
```
[2013-07-03 06:47:23 PDT] inbound file
C:\gsecdump.exe
[2013-07-03 06:47:46 PDT] saved PI-
extractedinbound-
file-1-gsecdump.exe..
```




```
[2013-07-03 06:47:46 PDT] *** Shell Session ***
gsecdump.exe -a > hash.txt
0043B820
info: you must run as LocalSystem to dump LSA
secrets
[2013-07-03 06:47:46 PDT] *** Shell Session ***
C:\>
```




```
[2013-07-03 06:47:54 PDT] *** Directory Listing
Sent ***
AUTOEXEC.BAT
boot.ini
CONFIG.SYS
gsecdump.exe
hash.txt
IO.SYS
MSDOS.SYS
NTDETECT.COM
ntldr
pagefile.sys
[2013-07-03 06:48:02 PDT] outbound file
C:\hash.txt
[2013-07-03 06:48:02 PDT] saved PI-extractedoutbound-
file-2-hash.txt..
```



FireEye

Defender's View

```
[2013-07-03 06:48:57 PDT] *** Screen Capture Sent
***
PI-extracted-file-3-screenshot.bmp saved..
[2013-07-03 06:49:03 PDT] *** Remote Desktop
Session ***
[2013-07-03 06:49:03 PDT] *** Remote Desktop
Session ***
Shutting Down Modules ...
Shutting Down poisonivy_23x
Module Shutdown Complete ...
ChopShop Complete
```

Attacker's View

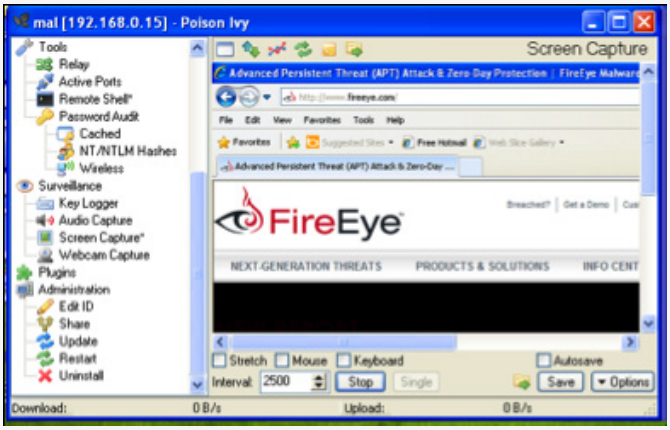


Figure 7: Example PIVY commands an views (defender, left; attacker, right)

using our Calamine ChopShop module, we reconstructed what operations occurred.

After the initial compromise, the "attackers" see that they have a new target endpoint and do the following:

- Execute some basic commands such as ipconfig to collect the network information of the endpoint
- Upload the password-dumping tool gsecdump (available at http://www.truesec.se/sakerhet/verktyg/saakerhet/gsecdump_v2.0b5)
- Dump the password hashes on the endpoint to a file
- Download a file containing the password hashes off the endpoint (to crack the target's passwords offline)
- Take a screenshot of the target's desktop

Extracting Intelligence

APT activity is best described as a campaign—a series of attacks over time. Each individual attack within a campaign can be divided into the following phases:[20,21,22]

- Reconnaissance
- Exploitation
- CnC
- Lateral movement
- Exfiltration (or other malicious actions on the target)

Each of these phases provides opportunities to derive threat intelligence about the adversary. Over time, security professionals can acquire and analyze evidence to determine whether the attacks constitute malware-based espionage.

Such an assessment requires understanding these components of an attack:

- Timing and targeting preferences
- Exploits and malware
- Network infrastructure
- Scope of attackers' activities within a compromised network (including stolen data)
- Characteristics of the target population

This assessment demands more than just malware analysis. It requires analyzing both the technical and contextual aspects of a breach and exploring competing hypotheses.[23,24] These steps are important because investigators will always face visibility gaps—limitations in knowing the geographic and industry reach of attacks or details of malware activity in some phases of the attack.

Poison Ivy Sample Analysis

For this analysis, we collected 194 Poison Ivy samples used in targeted attacks between 2008 and 2013. We extracted 22 different passwords and 148 mutexes. We also mapped out the CnC infrastructure, which comprised 147 domains and 165 IP addresses.

We analyzed these samples to better understand the tools, tactics, and procedures (TTPs) of the attackers, explore campaign connections among them, and enable defenders to better secure their networks. In addition to clustering the samples based on technical indicators, such as the passwords and CnC information extracted from the samples, we also analyzed contextual indicators where possible, such as attackers' targeting preferences and lures used in social engineering.

[20] SANS Computer Forensics. "Security Intelligence: Defining APT Campaigns." June 2010.

[21] SANS Computer Forensics. "Security Intelligence: Attacking the Cyber Kill Chain." October 2009.

[22] Richard Bejtlich. "Incident Phases of Compromise." June 2009.

[23] Richard Bejtlich. "Attribution Is Not Just Malware Analysis." January 2010.

[24] Jeffrey Carr. "Mandiant APT1 Report Has Critical Analytic Flaws." February 2013.

Each PIVY server (the malware that the attacker sends to the target) can be configured to connect to multiple CnC servers using any TCP port. So seeing a PIVY sample that attempts to connect to multiple CnC servers on different TCP ports is not unusual. But the most common ports used in targeted attacks are those associated with Web traffic—especially 443, the TCP port used for SSL-encrypted Web traffic.

Port 443 is a significant choice for two reasons. First, perimeter defenses must allow outbound traffic through this port so that users can access legitimate SSL-encrypted websites. Second, because the traffic on port 443 is encrypted, PIVY's encrypted traffic may bl end in with normal network activity. (Many protocol-aware perimeter defenses, however, can detect and flag PIVY traffic).

Table 1: Common TCP ports used by PIVY variants in APT attacks

TCP Port Used	PIVY Sample Count
443	157
80	104
8080	22
8000	12
1863	7

Table 2: Common process mutex seen in PIVY variants attributed to APT attacks

PIVY Process Mutex	PIVY Sample Count
)!VoqA.I4	14
K^DJA^#FE	4
KEIVH^#$S	3
%1Sjfhtd8	3
2SF#@R@#!	3

The attacker can set the PIVY process mutex name at build time.25 While some attacks use the default mutex of)!VoqA.I4, most create a custom mutex for each attack. Of the 147 mutexes in our sample set, 56 were designed for one-time use.

If attackers create a unique password at build time rather than using the PIVY default "admin", that custom password is the most unique indicator. While threat actors may change passwords used over time, we have found that they often use the same one for significant periods. When combined with CnC data, the passwords used by the actors provide unique indicators that can be used to cluster related activity.

Clustering

To cluster the activity of specific APT campaigns across our PIVY sample set, we first grouped the PIVY samples by common CnC infrastructure. Using passive DNS, we clustered the domain names used by the common IP address to which they resolved. Because attackers may park their domains by pointing to IP addresses that they do not necessarily control (and to account for other possible anomalies in passive DNS data), we layered additional indicators extracted from the samples, such as PIVY passwords, mutexes, campaign "marks/codes,"26 and launcher information.27 From our data set, we focused on three separate APT campaigns and corresponding threat actors identified by the PIVY password used in subsequent attacks:

- admin@338
- th3bug
- menuPass

Each of these campaigns is detailed in the corresponding sections.

Table 3: Common PIVY passwords seen in PIVY variants attributed to APT attacks

PIVY Password	PIVY Sample Count
admin	38
keaidestone	35
menuPass	24
suzuki	14
happyyongzi	13

[25] A mutex is a Windows object used for inter-process synchronization. They are often used by malware to ensure only one instance of the malware is running on an infected system at a given time.

[26] A campaign mark/code is typically a string designated by a threat actor that is often included as part of the malware communication and/or embedded within the malware binaries. It is used to identify targeted attack campaigns against a set number of targets (usually by industry), so the threat actor can keep attacks organized.

[27] Launchers are malware built specifically to load other malware (payload), often by decrypting the payload and injecting it into a host process on the target's endpoint.

To triangulate the timing (when the sample was likely used), we used the portable executable (PE) compile time extracted from the PIVY samples and the date each sample appeared first appeared in a malwareanalysis services such as VirusTotal. Each of these APT campaigns has been active from 2008 through 2013.

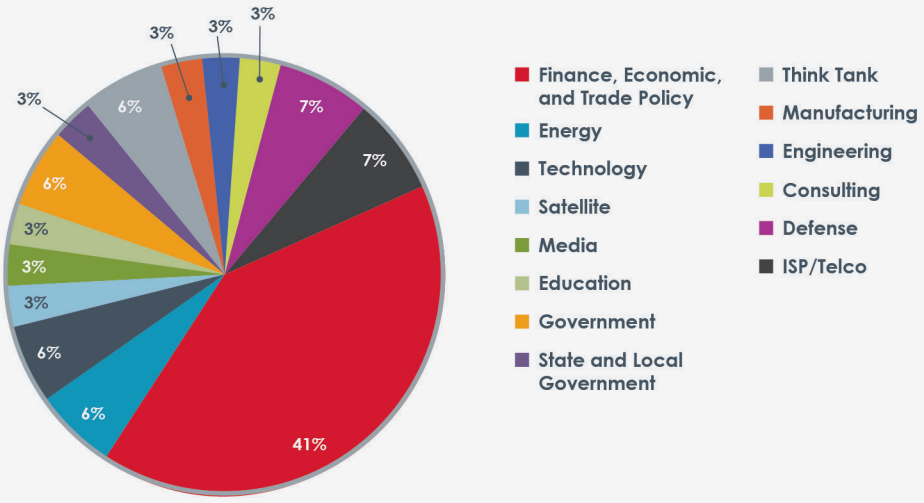
Campaign 1: admin@338

Our data set for the admin@338 threat actor contains the following:

- 21 Poison Ivy samples
- 3 passwords
- 43 CnC servers

The earliest admin@338 PIVY sample we have dates to December 27, 2009. But we believe that this campaign was active as early as January 7, 2008, using other PIVY passwords (key@123 and gwx@123). This ongoing campaign tends to target the finance, economic, and trade policy but we see significant activity in the ISP/telco, government, and defense sectors as well.

Figure 9: Percent of admin@338 APT group attacks by industry



Attack vector

The preferred attack vector used by this campaign is spear-phishing emails. Using content that is relevant to the target, these emails are designed to entice the target to open an attachment that contains the malicious PIVY server code.

The content of the spear-phishing emails and the decoy documents opened after exploitation tend to be in English—although the character set of the email message body in Figure 10 is actually Chinese (character set GB2312).[28]

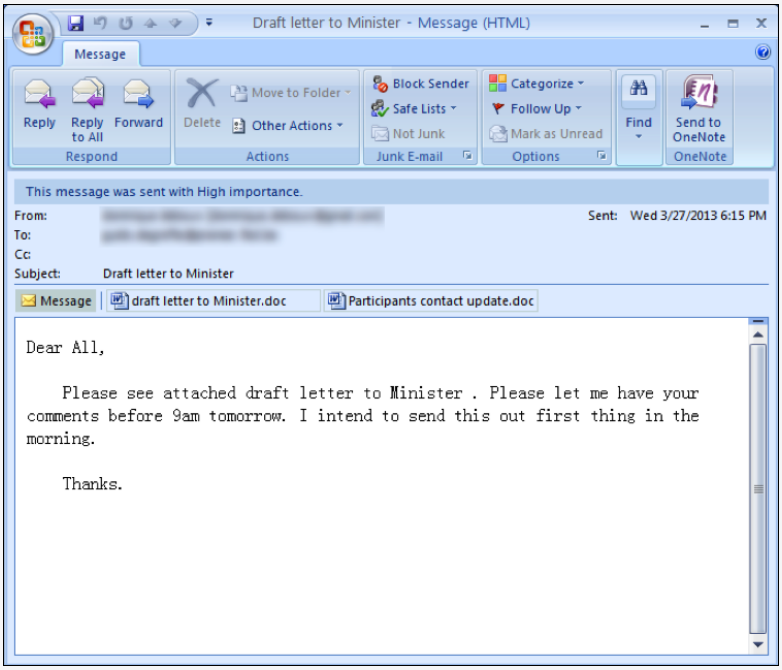


Figure 10: Example spear-phishing email launched by the admin@338 APT group

[28] Wikipedia. "GB 2312." February 2013.

Weaponization

This campaign has used weaponized Microsoft Word documents (CVE-2012-0158),[29] Adobe Acrobat PDFs (CVE-2009-4324)[30] and Microsoft Help Files (.HLP) to drop PIVY on their targets.

The decoy documents that are opened in exploitation typically contain content that is contextually relevant to both the text of the spear-phishing email and to the interests of the intended target. The documents are legitimate—but weaponized—documents in English.

Clustering

In addition to the PIVY password admin@338, we clustered individual attacks by using passive DNS data to look at the IP addresses the CnC servers have resolved to over time. We found that common IP addresses among PIVY samples for admin@338, key@123 and gwx@123.

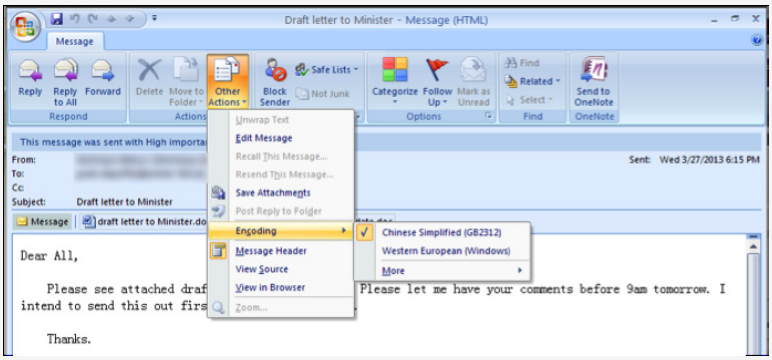


Figure 11: GB2312 encoding in spear-phishing email launched by the admin@338 APT group

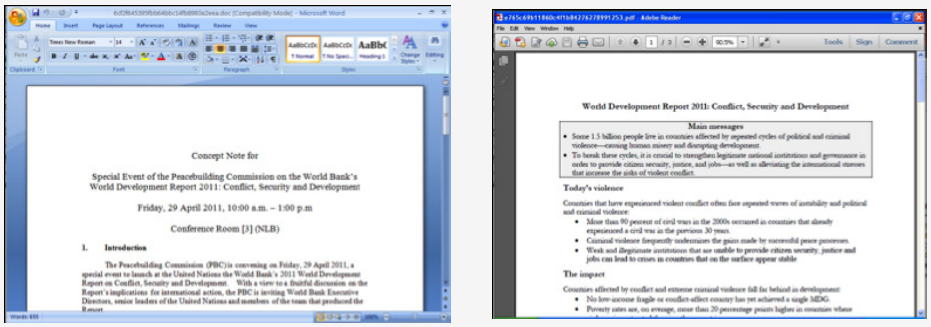


Figure 12: Contents of decoy attachments used by the admin@338 APT group

[29] National Institute of Standards and Technology. "Vulnerability Summary for CVE-2012-0158." April 2012.

[30] National Institute of Standards and Technology. "Vulnerability Summary for CVE-2009-4324." December 2009.

We can link PIVY passwords key@123 with admin@338 by observing the following connections:

- The key@123 sample, 808e21d6efa2884811fbd0adf67fda78, connects directly to 219.76.208.163.

- Two CnC domain names from the admin@338 sample 8010cae3e8431bb11ed6dc9acabb93b7, www.webserver.dynssl.com and www.webserver.freetcp.com, resolved to that same IP address (219.76.208.163).

We can link PIVY passwords gwx@123 with admin@338 by observing the following connections:

• The gwx@123 sample 0323de551aa10ca6221368c4a73732e6 connects to the CnC domain names microsofta.byinter.net, microsoftb.byinter.net, microsoftc.byinter. net, and microsofte.byinter.net. These domain names resolved to 113.10.246.30 219.90.112.203, 202.65.220.64, 75.126.95.138, 219.90.112.197, 202.65.222.45, and 98.126.148.114.

- The admin@338 sample 8010cae3e8431bb11ed6dc9acabb93b7 connects to the CnC domains www.webserver.fartit.com, www.webserver.freetcp.com, and www.webserver.dynssl.com.

- www.webserver.fartit.com resolved to 113.10.246.30, 219.90.112.203, 202.65.220.64, and 75.126.95.138, which overlap with the gwx@123 IP addresses.

- www.webserver.freetcp.com resolved to 113.10.246.30, 219.90.112.203, 202.65.220.64, 75.126.95.138, 219.90.112.197, and 202.65.222.45, which overlap with the gwx@123 IP addresses.

- www.webserver.dynssl.com resolved to 113.10.246.30, 219.90.112.203, 219.90.112.203, 75.126.95.138, 219.90.112.197, and 202.65.222.45, which overlap with the gwx@123 IP addresses.

This data indicates a relationship among the threat actors behind these attacks—in most cases, they at least share a common CnC infrastructure.

In addition to historic DNS resolutions, PIVY process mutexes suggest a relationship between PIVY passwords gwx@123 and admin@338.

Although the mutexes of gwx@123, wwwst@Admin, and admin@338 samples were different, the choice of characters in the mutex revealed a similar pattern.

Campaign 2: th3bug

Our data set for th3bug threat actor comprises the following:

- 14 Poison Ivy samples
- 2 passwords
- 9 CnC servers

The earliest th3bug PIVY sample we have is dated October 26, 2009. This ongoing campaign targets a number of industries but appears to prefer targets in higher education and the healthcare sectors.

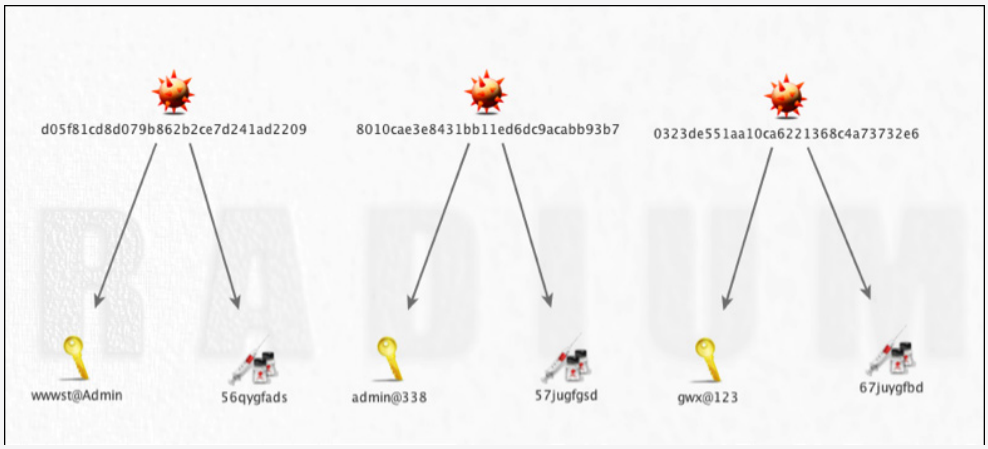


Figure 14: Linkage of admin@338 PIVY samples by password and mutex

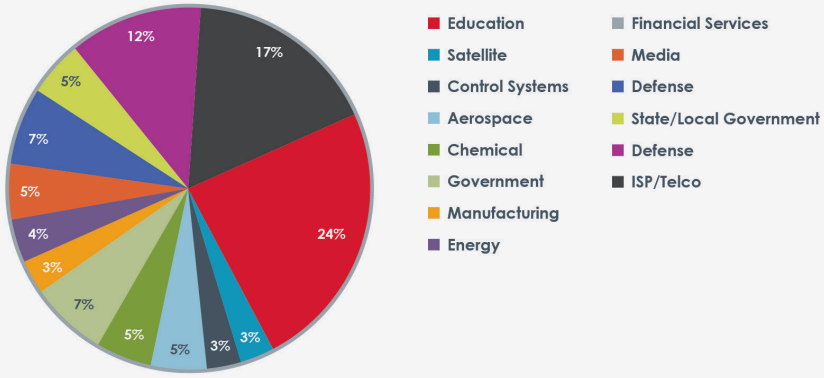


Figure 15: Percent of th3bug APT group attacks by industry

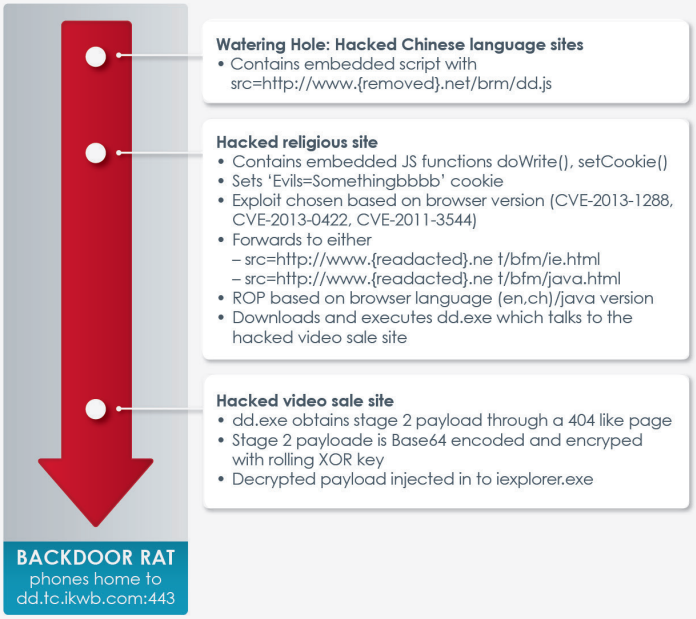
Attack vector

Unlike the other two campaigns described in this report (admin@338 and menuPass), th3bug does not appear to rely on spear phishing to distribute PIVY. Instead, attacks attributed to th3bug use a strategic Web compromise to infect targets. This approach is more indiscriminate, which probably accounts for the more disparate range of targets.

In the FireEye blog, we documented a recent th3bug strategic Web compromise.[31]

In the following example, the actor or actors behind the th3bug campaign compromised multiple websites that catered to the intended targets. The attacker used injected JavaScript on the compromised websites to redirect targets to an Internet Explorer exploit that dropped Stage 1 launcher/downloader mobile code. This downloader then retrieved and installed a PIVY RAT variant.

Figure 16: Example of initial infection vector by th3bug APT group



[31] Thoufique Haq and Yasir Khalid. "Internet Explorer 8 Exploit Found in Watering Hole Campaign Targeting Chinese Dissidents." March 2013.

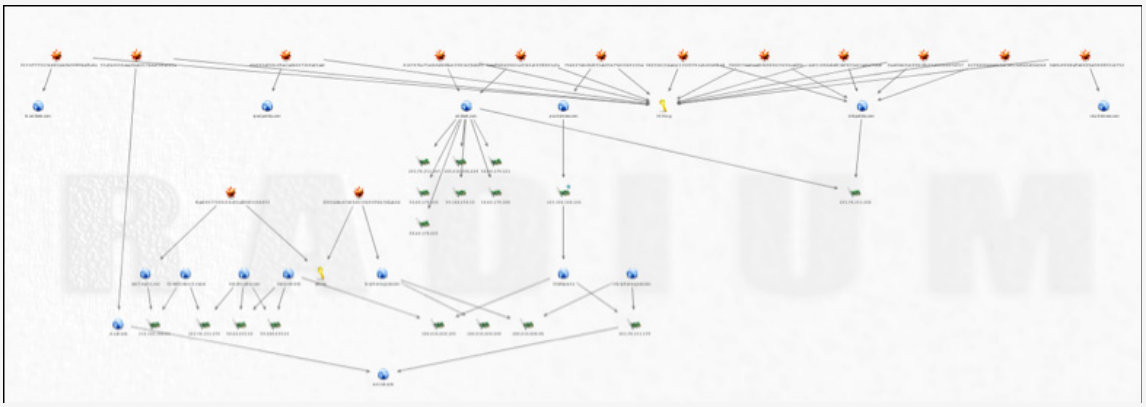
Weaponization

In related campaigns, th3bug has used a number of different Java and Internet Explorer exploits, including (CVE-2013-0422),[32] (CVE-2013-1347),[33] and (CVE-2011-3544).[34]

Figure 17: Example of ROP exploit code used by th3bug APT group

```
if (is_cn())
{
        ropcode = ""
        + int_to_hex(0x77c241da)
        + int_to_hex(0x77c241da)
        + int_to_hex(0x77c17653)
        + int_to_hex(0x00000001)
        + int_to_hex(0x77c1e18c)
        + int_to_hex(0x00001000)
        + int_to_hex(0x77c1017f)
        + int_to_hex(0x00000040)
        + int_to_hex(0x77bfe93d)
        + int_to_hex(0x77c09f92)
        + int_to_hex(0x77c16efb)
        + int_to_hex(0x77bfaacc)
        + int_to_hex(0x77c1ded4)
        + int_to_hex(0x77be110c)
        + int_to_hex(0x77d23ad9)    //////
        + int_to_hex(0x77c21025)
        + unescape("%u3eeb%u9090%u9f92%u77c0%u9f92%u77c0%u9f92%u77c0%u9f
92%u77c0%u9f92%u77c0%u9f92%u77c0%u9f92%u77c0%u9f92%u77c0%u9f92%u77c0%u9f92%u77c0
%u64d5%u77c0%u5ed5%u77be%u1025%u77c2%u9f92%u77c0%u0c8c%u0c0c");
}
else if(is_en())
```

Figure 18: Cluster intel of the th3bug APT group



[32] National Institute of Standards and Technology. "Vulnerability Summary for CVE-2013-0422." February 2013.

[33] National Institute of Standards and Technology. "Vulnerability Summary for CVE-2013-0422." January 2013.

[34] National Institute of Standards and Technology. "Vulnerability Summary for CVE-2011-3544." October 2011.

The default PIVY password of admin has been used by multiple, distinct threat actors—so clearly, we cannot link all PIVY samples with the admin password to th3bug. But evidence suggests that the attackers originally used the default password before settling on th3bug. We can link at least one PIVY sample that uses the admin password to the th3bug campaign based on the following connections:

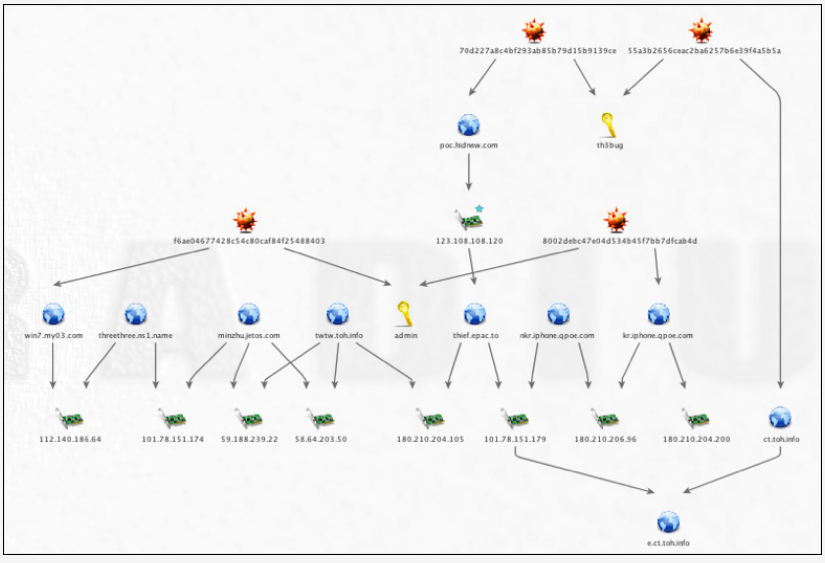
- The sample 8002debc47e04d534b45f7bb7dfcab4d connected to kr.iphone.qpoe.com with the PIVY password admin.
- The domain kr.iphone.qpoe.com resolved to 180.210.206.96 on January 12, 2012.
- The domain nkr.iphone.qpoe.com also resolved to 180.210.206.96 on January 3, 2012.
- The domain nkr.iphone.qpoe.com also resolved to 101.78.151.179 on December 23, 2011.
- The domain e.ct.toh.info resolved to 101.78.151.179 on June 12, 2012.
- The sample 55a3b2656ceac2ba6257b6e39f4a5b5a connected to ct.toh.info domain with the PIVY password th3bug.

We found a smaller number of distinct PIVY samples linked to th3bug than we did for the admin@338 and menuPass campaigns. This paucity is likely a result of two factors.

First, th3bug does not appear to stage a high volume of attacks. Instead, it appears to run only a handful of strategic Web compromise attacks each year. Second, th3bug stages its delivery of PIVY.

So to acquire the second-stage PIVY server payload, an attack must be observed in real time.

Figure 19: Partial cluster intel of the th3bug APT group (zoomed in)



FireEye

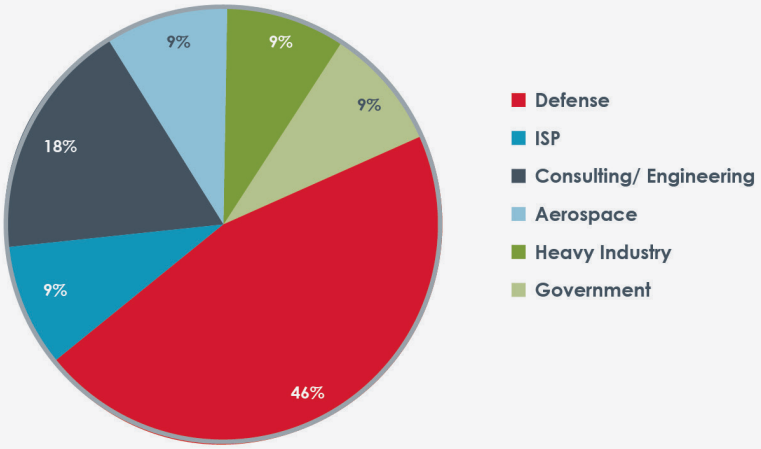
Campaign 3: menuPass

Our data set for the menuPass threat actor comprises the following:

- 118 Poison Ivy samples
- 8 passwords
- 61 domains
- 74 IP addresses

The earliest menuPass PIVY sample we have is dated December 14, 2009. This sample (b08694e14a9b966d8033b42b58ab727d) connected to a control server at js001.3322.org with a password xiaoxiaohuli (Chinese translation: "little little fox"). Based on what we have found, it appears that the threat actor behind menuPass prefers to target U.S. and foreign defense contractors.

Figure 20: Percent of menuPass APT group attacks by industry



Attack vector

The menuPass campaign appears to favor spear phishing to deliver payloads to the intended targets. The email shown in Figure 21 shows a typical menuPass spear-phishing attempt.

While the attackers behind menuPass have used other RATs in their campaign, it appears that they use PIVY as their primary persistence mechanism.

Weaponization

The menuPass campaign has used weaponized Microsoft Word documents (CVE-2010-3333)35 and ZIP files containing executable files to drop PIVY directly onto its targets. Figure 22 outlines several executables delivered in ZIP files attached to menuPass spear-phishing emails.

Figure 21: Example of spear-phishing email launched by the menuPass APT group

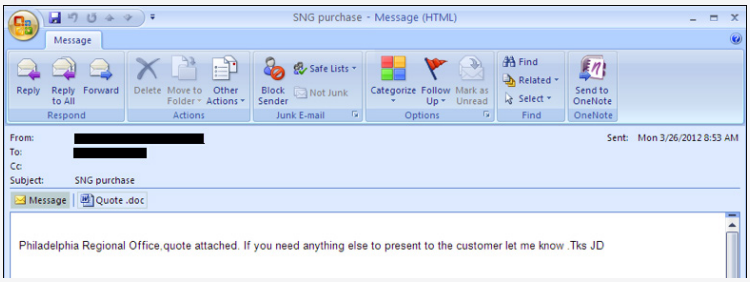


Figure 22: Example of weaponized, nested EXEs, used by menuPass APT group

File Name	Compile Time	MD5
Strategy_Meeting.exe	2012-06-11 04:41:31	8d6b6e023b4221bae8ed-37bb18407516
Background Consent Form.exe	2012-05-13 22:13:07	8d769c63427a8ce407d17946 702c7626
Doha_Climate_Change_Conference-November_2012.exe	2012-11-13 07:19:03	001b8f696b6576798517168cd 0a0fb44

Clustering

The menuPass attackers favor using a launcher that masquerades as a Microsoft Foundation Class Library application36 using the document/view architecture. This launcher includes a packed copy of the PIVY server that is subsequently unpacked and executed in memory shortly after a useless call to the FindFirstFile API. Out of the 155 samples we collected f or menuPass, 81 of them are MFC apps with a document class. Out of these 81 MFC launchers, 64 use the CBricksDoc class name. We also found these names:

- CMy20130401Doc
- CShellCodeDoc
- CMy20130401Doc
- CPiShellPutDo
- CCrocodileDoc
- CMy20130401Doc
- CStatePattern_GameDoc
- CPiShellPutDoc
- CPIVCDoc
- CMy1124Doc
- CLightGameDoc
- CPiShellPutDoc

Some samples were packed into projects taken from the Web and repurposed to serve as launchers.

The most popular PIVY password used by the menuPass campaign is keaidestone (used in 35 samples) followed by menuPass (24 samples). The threat actor also used these PIVY passwords in the same campaign:

- suzuki
- happyyongzi
- admin
- smallfish
- XGstone
- xiaoxiaohuli
- fishplay

A number of IPs in the 60.10.1.0/24 Classless Inter-Domain Routing (CIDR) block have hosted domains used in the menuPass campaign. We can see the connection between the keaidestone password and the XGstone password by observing the following connections in this same /24 CDIR block:

- The IP 60.10.1.120 hosted the domain apple.cmdnetview.com.
- The sample d8c00fed6625e5f8d0b8188a5caac115 connected to apple.cmdnetview.com with the password XGstone.
- The IP 60.10.1.115 hosted the domain autuo.xicp.net.
- The sample b1deff736b6d12b8d98b485e20d318ea connected to autuo.xicp.net with the password keaidestone.
- The samples b1deff736b6d12b8d98b485e20d318ea and d8c00fed6625e5f8d0b8188a5caac115 also shared the use of the CBricksDoc launcher.
- 08709f35581e0958d1ca4e50b7d86dba has a compile time of July 20. 2012 and connected to tw.2012yearleft.com with the password keaidestone. This sample also used the CBricksDoc launcher.
- 2012yearleft.com was registered on February 13, 2012 by zhengyanbin8@gmail.com.
- The domain cmdnetview.com was also registered on February 13, 2012 by zhengyanbin8@gmail.com.

Figure 21: Example of spear-phishing email launched by the menuPass APT group

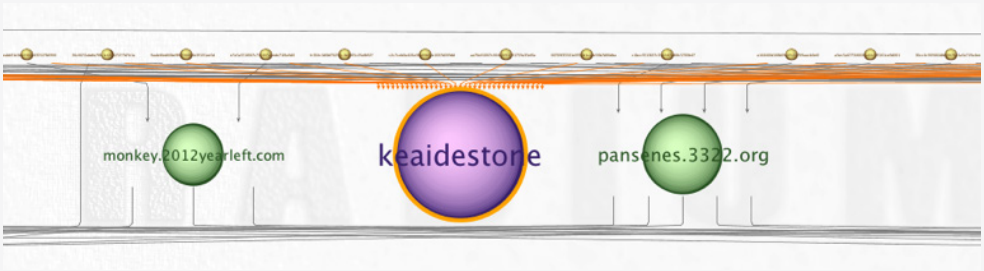
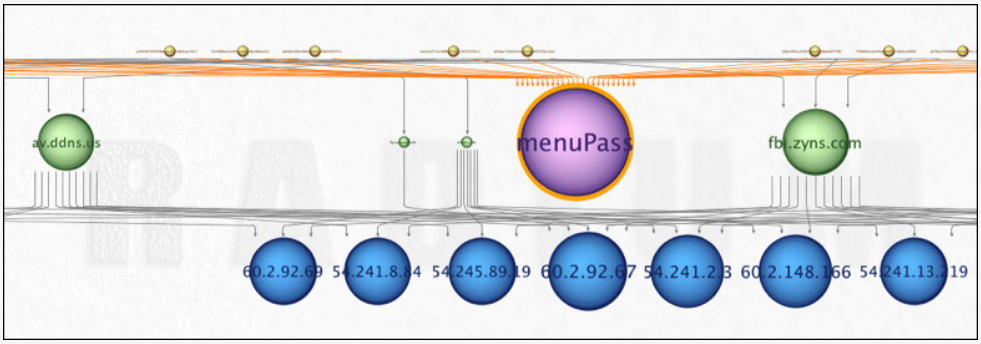


Figure 24: Partial cluster intel of the menuPass APT group (zoomed in on menuPass)



We can also see the connection between the keaidestone password and the smallfish password by observing the connections in the 60.10.1.0/24 CDIR block:

- The domain dedydns.ns01.us resolved to 60.10.1.121.
- The sample e84853c0484b02b7518dd683787d04fc connected to dedydns.ns01.us with the password smallfish and used the CBricksDoc launcher.

We can see the connection between the keaidestone password and the happyyongzi password by observing the connections in the 60.10.1.0/24 CDIR block:

- The domain maofajapa.3322.org resolved to 60.10.1.121.
- The sample cf8094c07c15aa394dddd4eca4aa8c8b connected to maofajapa.3322.org with the password happyyongzi.

The password suzuki can be linked to keaidestone by observing the following relationships:

- The sample 410eeaa18dbec01a27c5b41753b3c7ed connected to send.have8000.com with the password of suzuki.
- The domain have8000.com was registered on 2012-02-13 via the email zhengyanbin8@gmail.com.
- The same email of zhengyanbin8@gmail.com also registered cmdnetview.com on the same date of 2012-02-13.
- As stated above, the sample b2dc98caa647e64a2a8105c298218462 connected to apple.cmdnetview.com with the password XGstone.

We can link the password of menuPass to keaidestone by observing the following connections:

- 08709f35581e0958d1ca4e50b7d86dba has a compile time of July 20, 2012 and connected to tw.2012yearleft.com with the password keaidestone. This sample also used the CBricksDoc launcher.
- tw.2012yearleft.com resolved to 60.10.1.114 on June 6, 2012 and to 60.1.1.114 on March 11, 2013.
- The domain fbi.zyns.com resolved to 60.10.1.118 on August 21, 2012.
- 68fec995a13762184a2616bda86757f8 had a compile time of March 25, 2012 and connected to fbi.zyns.com with the password menuPass. This sample also used the CBricksDoc launcher.
- • The sample 39a59411e7b12236c0b4351168fb47ce had a compile time of April 2, 2010 and connected to weile3322b.3322.org with the password keaidestone. This sample used a launcher of CPiShellPutDoc.
- The sample f5315fb4a654087d30c69c768d80f826 had a compile time of May 21, 2010 and connected to ngcc.8800.org with the password menuPass. This sample also used a launcher of CPiShellPutDoc.

We can see the connection between the happyyongzi password and menuPass by observing the following connections:

- The sample e6ca06e9b000933567a8604300094a85 connected to the domain sh.chromeenter.com with the password happyyongzi.
- The domain sh.chromeenter.com previously resolved to the IP 60.2.148.167.
- The domain jj.mysecondarydns.com also resolved to 60.2.148.167.

Similar to other threat actors, this threat actor has also used PIVY samples using the default admin password. Again, not all PIVY samples with the password admin can be linked to menuPass. But we can see the connection between the menuPass and at least a couple of instances of PIVY using the admin password via the following connections:

- • The sample 56cff0d0e0ce486aa0b9e4bc0bf2a141 was compiled on 2011-08-31 and connected to mf.ddns.info with the password menuPass.
- The domain mf.ddns.info resolved to 54.241.8.84 on November 22, 2012. This same IP also hosted the domain av.ddns.us on the same date.
- The sample 60963553335fa5877bd5f9be9d8b23a6 was compiled on June 9, 2012 and connected to av.ddns.us with the password of admin.
- A number of menuPass and admin samples also shared the same CBricksDoc launcher including but not limited to 6d989302166ba1709d66f90066c2fd59 and4bc6cab128f623f34bb97194da21d7b6.
- The sample 4e84b1448cf96fabe88c623b222057c4 connected to jj.mysecondarydns.com with the password menuPass.

The password of fishplay can be linked to menuPass by observing the following relationships:

- The sample 494e65cf21ad559fccf3dacdd69acc94 connected to mongoles.3322.org with the password fishplay.
- The mongoles.3322.org domain resolved to 123.183.210.28.
- The domain a.wubangtu.info also resolved to 123.183.210.28.
- The sample a5965b750997dbecec61358d41ac93c7 connected to 3q.wubangtu.info with the password menuPass.
- The sample 494e65cf21ad559fccf3dacdd69acc94 and a5965b750997dbecec61358d41ac93c7 also share the same CBricksDoc launcher.

We can link the password of xiaoxiaohuli to menuPass through the shared CPiShellPutDoc launcher:

- f5315fb4a654087d30c69c768d80f826 had a compile time of May 21, 2010 and connected to ngcc.8800.org with the password of menuPass.
- e6ca06e9b000933567a8604300094a85 had a compile time of June 29, 2010 and connected to sh.chromeenter.com with the password happyyongzi.

- Both f5315fb4a654087d30c69c768d80f826 and e6ca06e9b000933567a8604300094a85 use the same CPiShellPutDoc launcher.

Finally, we can link the password of happyyongzi to xiaoxiaohuli by observing the following relationships:

- e6ca06e9b000933567a8604300094a85 has a compile time of 2010-06-29 and connects to sh.chromeenter.com with the password happyyongzi.
- e62584c9cd15c3fa2b6ed0f3a34688ab has a compile time of 2009-12-28 and connects to the domain js001.3322.org with the password xiaoxiaohuli.
- Both e6ca06e9b000933567a8604300094a85 and e62584c9cd15c3fa2b6ed0f3a34688ab use the same CPiShellPutDoc launcher.

Figure 21: Example of spear-phishing email launched by the menuPass APT group

Conclusion

We cannot say with certainty why the actors responsible for the admin@338, menuPass, and th3bug campaigns rely on Poison Ivy. But possible explanations include PIVY's easy-to-use features and the relative anonymity that an off-the-shelf RAT provides for attackers.

Compared to other RATs, PIVY is very easy to operate. Its graphical user interface (GUI) makes building new servers and controlling infected targets simple. Attackers can point and click their way through a compromised network and exfiltrate data.

Commodity RATs also complicate efforts by security professionals to correlate a threat actor's activity over time—attackers can hide in the sea of malicious activity that also uses Poison Ivy-based malware.

By exposing the role of PIVY and other commodity RATs in APT campaigns we hope to complicate attackers' ability to hide behind these off-the-shelf tools—and perhaps force them away from using these RATs.

In this report, we have provided several techniques that network defenders can use to not only identify a PIVY infection, but also classify and correlate detected infections to previously observed APT activity.

In the process of building their PIVY servers, attackers leave a number of potentially useful clues, such as:

- The domains and IPs used for CnC
- The chosen PIVY process mutex
- The chosen PIVY password
- Launcher code used in the droppers
- Timeline of activity
- Targets of attack

Together, all of these data points can help effectively identify and correlate APT activity that uses the Poison Ivy RAT.

About FireEye

FireEye has invented a purpose-built, virtual machine-based security platform that provides real-time threat protection to enterprises and governments worldwide against the next generation of cyber attacks. These highly sophisticated cyber attacks easily circumvent traditional signature-based defenses, such as next-generation firewalls, IPS, anti-virus, and gateways. The FireEye platform provides real-time, dynamic threat protection without the use of signatures to protect an organization across the primary threat vectors, including Web, email, and files and across the different stages of an attack life cycle.

The core of the FireEye platform is a virtual execution engine, complemented by dynamic threat intelligence, to identify and block cyber attacks in real time. FireEye has over 1,000 customers across more than 40 countries, including over one-third of the Fortune 100.

FireEye, Inc. | 1440 McCarthy Blvd. Milpitas, CA 95035 | 408.321.6300 | 877.FIREEYE (347.3393) | info@fireeye.com | **www.fireeye.com**

FireEye