

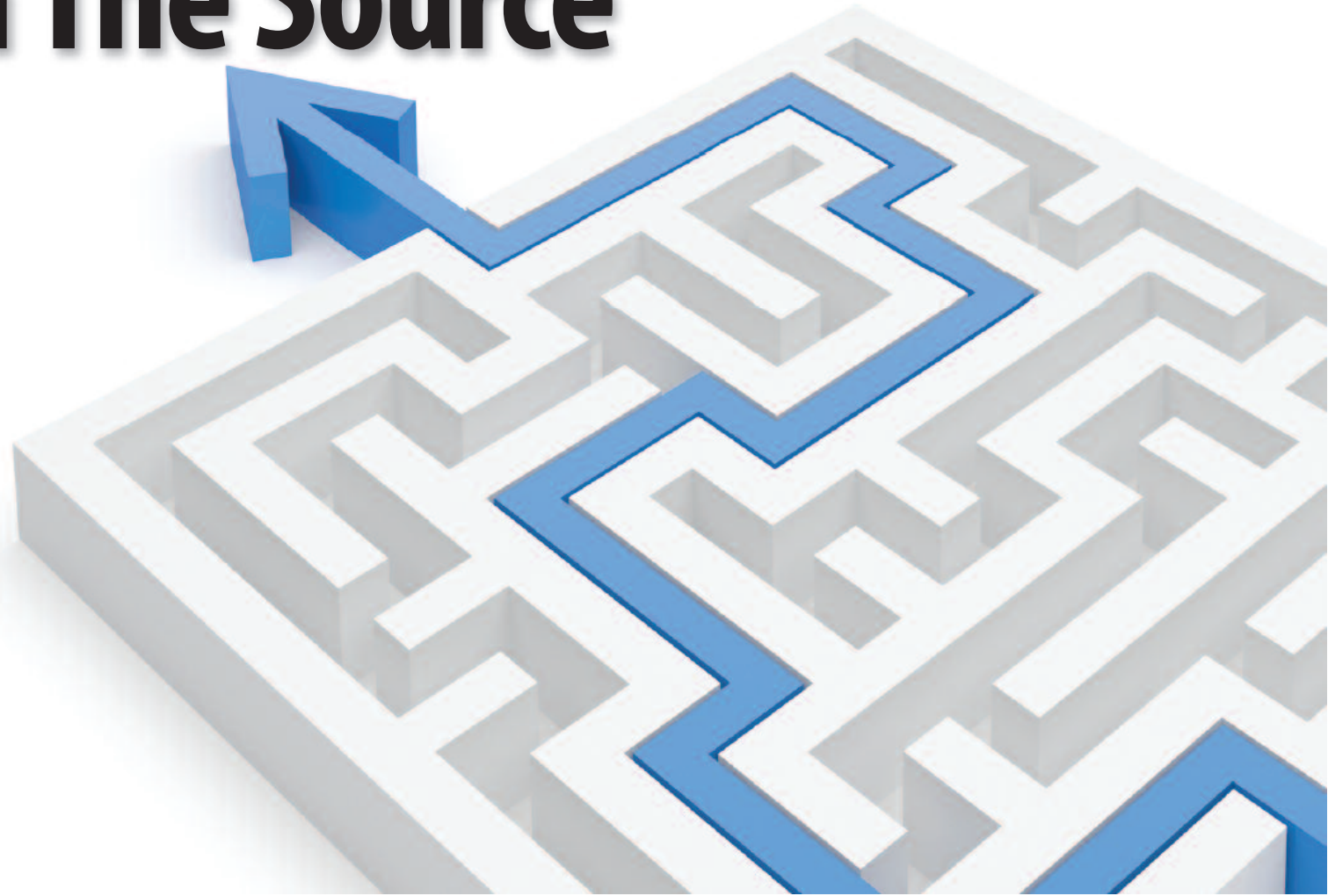
How Did They Get In? A Guide to Tracking Down The Source of APTs

If you think that your organization hasn't been affected by an advanced persistent threat, you probably haven't looked hard enough. Identifying that your organization is under attack is difficult enough; determining the scope of infiltration and damage presents a whole new level of challenge. To effectively protect against APTs, security pros will need to employ an arsenal of tools in a coordinated fashion, as well as develop new understandings of and approaches to system and data exploits.

By Michael Cobb

Presented in conjunction with

dark SECURITY **READING**
Protect The Business  Enable Access



CONTENTS
TABLE OF

- 3 Author's Bio
- 4 Executive Summary
- 5 How Did They Get In? A Guide to Tracking Down The Source of APTs
- 5 Monitoring and Logging
- 5 Figure 1: Digging for Clues
- 6 Figure 2: Timespan of Events by Percent of Breaches
- 8 Behavioral Analysis
- 8 Figure 3: Time Between Initial Attack and Initial Compromise-Larger Organizations
- 9 Picking Through Processes
- 10 Group Effort
- 12 Related Reports



ABOUT US

InformationWeek Reports' analysts arm business technology decision-makers with real-world perspective based on qualitative and quantitative research, business and technology assessment and planning tools, and adoption best practices gleaned from experience. To contact us, write to managing director **Art Wittmann** at awittmann@techweb.com, content director **Lorna Garey** at lgarey@techweb.com, editor-at-large **Andrew Conry-Murray** at acmurray@techweb.com, and research managing editor **Heather Vallis** at hvallis@techweb.com. Find all of our reports at reports.informationweek.com



[Previous](#)[Next](#)[Table of Contents](#)

InformationWeek
:: reports

dark
READING

How Did They Get In? A Guide to Tracking Down the Source of APTs



Michael Cobb

InformationWeek Reports

Michael Cobb, CISSP-ISSAP, is a renowned security author with more than 15 years of experience in the IT industry. He is the founder and managing director of Cobweb Applications, a consultancy that provides data security services. He co-authored the book *IIS Security* and has written numerous technical articles for leading IT publications. Michael is also a Microsoft Certified Database Administrator.

Want More?

**Never Miss
a Report!**



Follow



Follow

SUMMARY

EXECUTIVE

Advanced persistent threats are just what their name implies—sophisticated and stubborn. It is difficult to identify that your organization's systems and data are under an APT attack, let alone uncover all of the attack's component parts, find out the source of attack, determine the scope of infiltration and damage, and identify the attacker (with the latter being the most difficult task of all). To get at the root of the problem, security professionals must leverage a great many tools and employ in-depth (and often manual) analysis of log files, network traffic and program code. Logging and monitoring, behavioral analysis and training are important components of any efforts to identify and dissect APTs. Indeed, many organizations will find that they cannot go it alone. Combining the experience, knowledge base and resources of the business and security communities will be a critical factor in mitigating—and ultimately eliminating—APTs.

How Did They Get In? A Guide to Tracking Down The Source of APTs

Advanced persistent threats are a complex security problem, but there are two things that all APTs have in common: They are hard to detect and come into your network in unusual (often zero-day) ways. It is difficult to uncover an APT, but, once you do, the hard work really begins: finding the source of the problem, identifying the attacker and figuring out to what extent the attack has affected your organization's systems.

Discovering the actual APT attack code requires a proactive, hands-on approach involving in-depth analysis of log files, network traffic and program code. The goal is to uncover behavior indicative of APT activity: network exploration and data exfiltration. Even the best and brightest security teams may be challenged by the sophistication of some of the attacks we have seen lately, but security professionals should at least have an understanding of the methods used to carry out an APT. In this report, we will examine the types of tools and processes that can be used to find and isolate

an APT, as well as provide insight into leveraging these tools and processes to build additional defenses against APT infection.

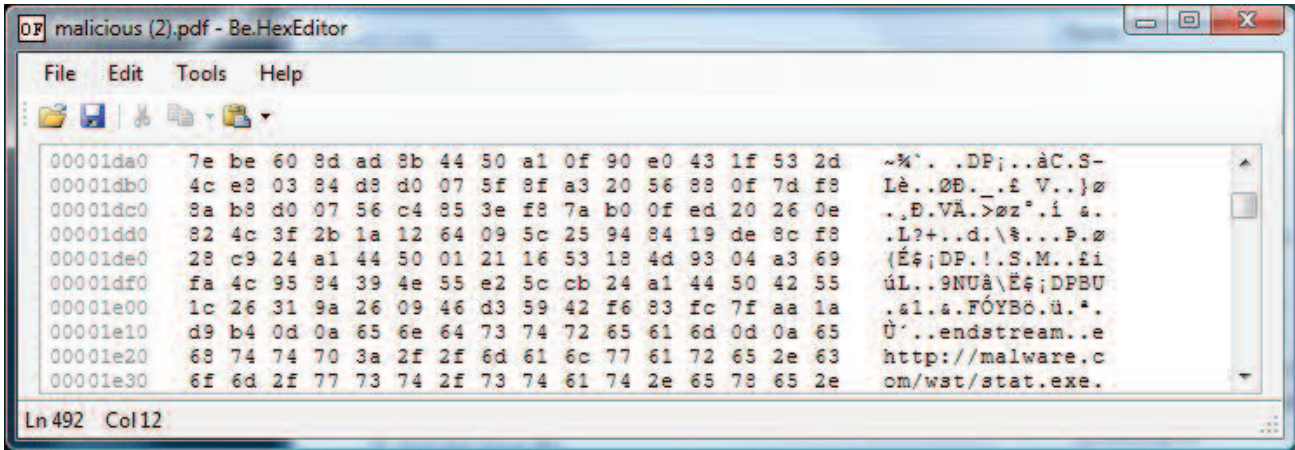
Monitoring and Logging

The Achilles' heel of any APT is that it has to send the data that it has collected back to a command and control server (CCS) to success-

fully complete its mission. This network activity, as well as the APT's attempts to explore the network in search of data, will provide the few (if you are lucky) chances you will have to identify and halt the threat. It is therefore essential that you extensively monitor and log network traffic—in particular, outbound traffic.

By collecting and analyzing records of traffic

Figure 1
Digging for Clues



Once the obfuscated attack shellcode has been extracted from an infected file (in this case, a PDF document), examination in a hex editor can sometimes reveal malicious embedded links. Here, stat.exe will be downloaded once the shellcode is executed.

Source: Michael Cobb

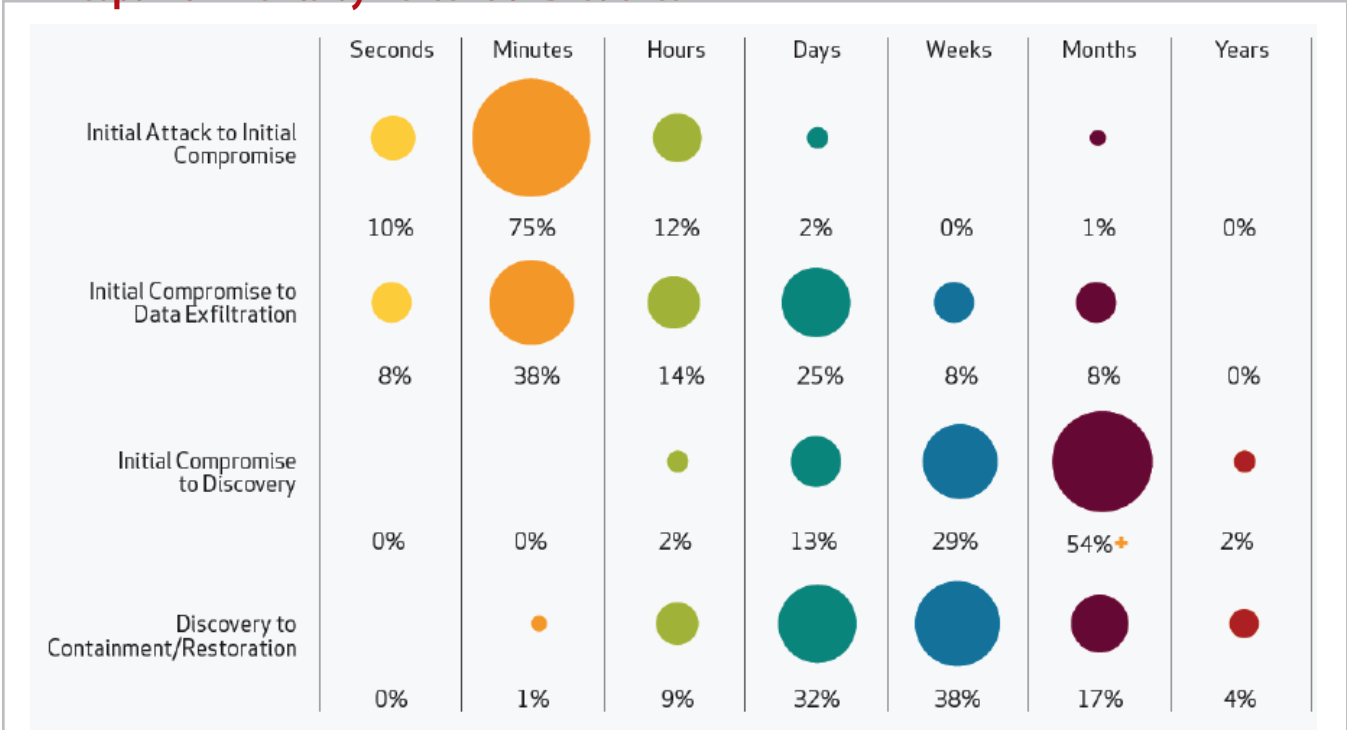
S4750412/1

flow, security teams can increase the chances of spotting intrusions and other potentially malicious activity. Unexpected and therefore suspicious behavior might include a desktop scanning ports or a file server sending traffic outside of the network. Activity such as this, carried out without permission, should ring alarm bells and trigger your company’s escalation procedure.

Comprehensive logs will help enormously in unraveling the spread of an infection. A machine making connections to an unknown server, for example, could indicate malicious activity of some sort. If administrators can query a Cisco NetFlow or similar database to search for other machines making connections to the same IP address and port, they can target those machines for further investigation. Analysis of the communications among these machines can help determine where an attack originated and which other machines have been affected.

Indeed, bandwidth monitoring and network traffic analysis tools such as NetFlow should be used to collect IP traffic passing

Figure 2
Timespan of Events by Percent of Breaches



Source: Verizon 2012 Data Breach Investigations Report

S4750412/2

through routers and switches for analysis. To make analysis more productive, a baseline of well-known, expected behavior should be established. This makes the creation of effective rule sets easier because thresholds can be

set based on expected activity. Baselineing can also provide early warnings of a zero-day attacks because traffic on your network will start to fall outside of what’s deemed normal. If resources are tight, concentrate on estab-



Strategy: Detecting and Defending Against Advanced Persistent Threats

APTs are a growing problem for enterprises big and small. Protecting your organization from these targeted threats requires constant vigilance, ongoing employee training and a concerted effort to align security systems to address every phase of an APT. Companies also need to develop a remediation and response plan if, despite best efforts, defenses are breached.

[Download](#)

lishing a baseline for high-risk network segments. Rules that enforce network and security policies should be reviewed to ensure that connections to these sensitive systems and data are expected as well as allowed.

OSSEC is a free, open source host-based intrusion detection system (IDS) that can provide log analysis, file integrity checking and Windows registry monitoring. It works with event logs from a variety of firewalls, IDSes, Web servers, switches and routers to provide real-time correlation and analysis, policy monitoring and alerts. This type of tool is essential for blocking and catching malicious information-gathering processes such as port scans and brute-force attacks.

Zero-day exploits are the nightmare of most administrators responsible for perimeter security, but one defense is to try and detect the payload. A major step toward identifying APT activity is to configure IDS signatures to block exfiltration by firing alerts based on the characteristics of outgoing traffic. For example, an APT using PI-RAT, or the Poison Ivy Remote Access Toolkit, can be trapped

with an IDS rule that checks traffic going outbound on Port 3460 for the presence of the string used in the PI-RAT initial communication sequence flags.

While all of the tools we have mentioned so far require some manual tuning, they are, for the most part, automated. However, uncovering an APT will likely require manual analysis as well as automated analysis.

APTs most often run as a service. This allows a threat to recover from attempts to remove it while maintaining high levels of access and the ability to read the memory of other processes. It is important to perform regular manual reviews of log data to check that access control lists (ACLs) are being correctly enforced and to evaluate trends in network traffic. For example, if a demilitarized zone (DMZ) contains a Web server and a DNS server and ACLs are set to block traffic from any other type of system, administrators should be checking for alerts regarding traffic on ports other than 80, 443 and 53. Similarly, irregular ratios of upload-to-download traffic could indicate that files are being uploaded

to another server, while a SQL statement that is far larger than the average is probably a sign of a SQL injection attack.

An important tool in your log analysis arsenal is a network protocol analyzer tool such as **Wireshark**, which can either capture packet data from a live network or read packets from a previously saved capture file. Most importantly, though, you can set it up to work with specific types of traffic. This will make it quicker and easier to find what you're looking for among reams of traffic data. For example, to disguise the exfiltration process, some attacks use fast-flux to circumvent IP-based ACLs and make it more difficult to identify the attacker's own network. Just identifying the existence of such traffic in DNS logs could indicate compromise and possible exfiltration.

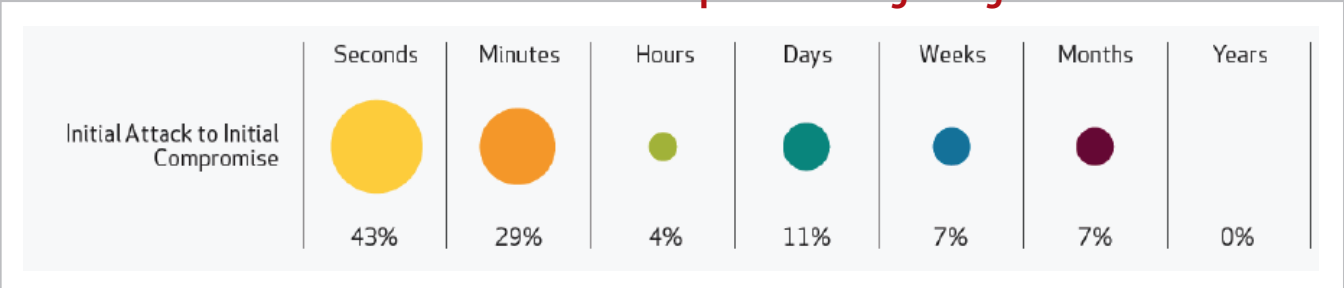
Another way to help security teams identify and escalate an incident more quickly is to make IDS and firewall alerts more user-friendly. For example, assigning location information to IP addresses means a concerned administrator doesn't need to perform a host lookup to see that a database server has

made an outbound FTP connection to a computer outside the network—an event that requires immediate investigation.

APTs tend to use HTTP and HTTPS as exfiltration methods, so filtering outbound Web traffic needs to become as widespread as the use of other egress filters. One method to trap and prevent an APT from sending data out of the network is to use a proxy such as [Squid](#) to permit or deny traffic. For example, if the only the Squid proxy can initiate HTTP/HTTPS sessions outbound on Port 80/443, then an APT would fail to directly exit the network on those ports. Security teams need to keep abreast of the latest attacks being used so that similar rules for such exploits can be generated to allow for automatic blocking and to deny access to specific destinations or limit activity to business hours. Blacklists of known bad IP addresses and domains can be updated automatically using SquidGuard.

Another valuable addition to any intrusion detection and network security monitoring toolbox is [Security Onion](#), which includes a host of tools that can be integrated with

Figure 3
Time Between Initial Attack and Initial Compromise-Larger Organizations



Source: Verizon 2012 Data Breach Investigations Report S4750412/3

Squid. These tools—which include Snort, Suricata, Sguil, Squert, Snorby, Bro, NetworkMiner and Xplico—can be used to build an array of distributed sensors and controls to monitor and allow early detection of APT behavior. When used together, these tools can provide real-time reporting with visualization. Squert, in particular, is a visual tool that provides additional context to events through the use of metadata, time series representations and weighted and logically grouped result sets.

Although APTs are usually the result of a directly targeted attack, any malware can be used as the starting point. The [RSA SecurID](#) APT attack started with a phishing exploit, for

example, so be sure to make use of the set of rules to identify and block phishing campaigns maintained by the [Sourcefire Vulnerability Research Team](#).

Behavioral Analysis

Once you have located the malware responsible for generating suspicious traffic, you will need to deconstruct it to take your investigations further. To ensure that code is contained, it is important to set up a virtual lab environment to carry out this work. To capture the maximum amount of information while the malware is running, it makes sense to use a variety of monitoring tools, including:

Like This Report?

Rate It!

Something we could do better? Let us know.

Rate

>**Regshot**-an open source registry compare utility that allows users to quickly take a snapshot of one registry and then compare it with another.

>**TCPView**-shows detailed listings of all TCP and UDP endpoints, including the local and remote addresses and state of TCP connections.

>**Process Explorer**-provides information about which handles and DLL processes have opened or loaded.

>**ListDLLs**-a utility that lists all DLLs loaded into all processes or into a specific process, as well as processes that have a particular DLL loaded.

>**VMMMap**-a virtual and physical memory analysis utility that shows summary information and a detailed process memory map to identify the sources of process memory usage.

>**Process Monitor**-shows real-time file system, registry and process/thread activity.

>**Capture-BAT**-a behavioral analysis tool that can be used to monitor the state changes of a system during program execution and

identify which processes are responsible for file or registry changes.

As you can see, several different tools are necessary to help determine what a program does and how it does it. No one says it will be easy, but the use of some or all of these tools will provide the kind of comprehensive view needed to show modifications to the registry, file system and system state—all of which provide insights into the way in which software operates, even if no source code is available.

The good news is that all of the tools we have listed here are free. The not-so-good news is that they require specialist knowledge to really make the most of them. Further, even if you make use of all of these tools, you will still have to do quite a bit of detective work. For example, if you have suspicions about a .chm help file (Microsoft Compiled HTML Help)—often used as a vehicle for delivering malware—uncovered with one of these tools, you should search for instructions on how to create Trojan horses using chm files. This will provide clues as to the possible name of the

executable file hidden within the .chm file. Perl2Exe, for example, is often incorporated into malware because of its ability to parse large amounts of data in search of credit card or other personally identifiable information.

Picking Through Processes

Another area you will have to keep a close eye on is processes. APT authors use the insider knowledge gained via phishing attacks and victim research to build up an understanding of how a target business application works so they can hook into relevant processes. Application-specific malware often looks to identify sensitive data in memory and storage or gain access to it while it's actually being processed. You are looking to establish how the malware program works; what data it is capturing and from where; and where the malware is sending the data after it has been captured. Standard actions include checking the registry key at SOFTWARE\Microsoft\windows\CurrentVersion\Uninstall to see which anti-virus programs are installed and injecting a malicious DLL into a running

process—explorer.exe being one of the most likely targets. Various APIs are commonly called, including GetComputerNameA and GetLogicalDrives, to gather machine and hard disk information. Collected data will most likely be written to temporary files, compressed in cab format and then renamed and located in a folder to await extraction.

However, malware authors have long used packers to obfuscate their binary code to avoid detection by anti-virus programs and

Discovering the actual APT attack code requires a proactive, hands-on approach involving in-depth analysis of log files, network traffic and program code.

make it harder for analysts to get to the bottom of their code. (The primary purpose of a packer is to reduce the size of an executable and provide encryption or reverse-engineering

protection to safeguard legitimate intellectual property and proprietary code within the application code.) Search for hidden executables using a detector such as [ExeInfo PE](#) and a packer such as [UPX](#) to unpack it. However using a common packer such as UPX makes

malware detection easier, so APTs may well be disguised using a proprietary packing routine that will require manually unpacking before further code review. Ideally, you need to step through each action the malware takes using a disassembler such as IDA or the debugger OllyDbg. The goal of this painstaking process is to decipher how the program works internally and determine where it injects processes into legitimate programs.

At some point, the malware will try to connect to its CCS. To allow the APT to establish connections with external IP addresses, you can redirect the connections to a [REMnux](#) system listening on the appropriate ports to emulate responses. (This obviously needs to be done inside an isolated virtual environment.) The conversation may well start with a standard HTTP request before the encrypted data is sent. This traffic needs to be captured and analyzed to see if any attempts are made to download more malware or determine whether the traffic can provide further clues as to who is behind the attack. It should be noted, however, that although you may be

able to trace IP addresses to a registered owner, you may not be able to prove beyond doubt that it is the registered owner who has targeted and infected your organization.

Group Effort

Clearly, finding an APT and fully understanding what it does requires comprehensive and skilled analysis. To give you an idea of just how challenging these tasks are, consider the DuQu Trojan horse program.

Late last year, malware based on the Stuxnet virus was uncovered. The threat was named DuQu because it created files with a DQ prefix. Researchers analyzed the code but could not determine what exactly it was and where it came from. In early March, Kaspersky Labs asked for help, using crowdsourcing to help decipher the code in the command and control communications module used in the DuQu program. It was determined that most of the code was written in C++, but one portion looked like it could have been written in a completely new programming language. It turned out to be object-oriented C code



compiled with Microsoft Visual Studio Compiler 2008 using special options. This indicates it was developed and integrated into the DuQu Trojan by a professional team of “old school” developers, possibly reusing older, well-tested code from an existing software project. With that said, targeted malware tends to be very modular, allowing per-attack processes to be constructed—particularly the code that handles exfiltration—and often sending stolen data automatically without the need for commands from the attacker. In short, these attacks are sophisticated and ever-changing. Security professionals’ efforts to find and understand them must be likewise savvy and flexible.

Finally, just as Kaspersky relied on the community to help it dissect DuQu, it is important for the security community to join forces in the fight against APTs. Indeed, sharing any information gathered about an APT is vitally important to improving our understanding, and thus defenses, against these attacks. [FIRST](#) is the global forum for incident response and security teams from

government, commercial and educational organizations. Its mandate is to foster cooperation and coordination in incident prevention, and it can help organizations with information assets that are of interest to foreign states and competitors to coordinate their security efforts. Cooperation is the only way to improve our ability to find and eradicate APTs.

Like This Report?

Share it!

Like Tweet

Share

WE
RE
MORE
LIKE THIS

Want More Like This?

InformationWeek creates more than 150 reports like this each year, and they're all [free to registered users](#). We'll help you sort through vendor claims, justify IT projects and implement new systems by providing analysis and advice from IT professionals. Right now on our site you'll find:

Strategy: Database Defense: The biggest threat to your company's most sensitive data may be the employee who has legitimate access to corporate databases but less-than-legitimate intentions. And while the incidence of insider data breaches has decreased, external attacks often imitate them—and do serious damage. Follow our advice to mitigate the risk.

Strategy: Choosing the Right Vulnerability Scanner for Your Organization: Vulnerability scanners can be used to help detect and fix systemic problems in an organization's security program and monitor the effectiveness of security controls. However, a vulnerability scanner can improve the organization's security posture only when it is used as part of a vulnerability management program, in which products, processes and people are working together to find, identify, prioritize and mitigate threats.

Strategy: Fundamentals of User Activity Monitoring: Benchmarking normal activity and then monitoring for users who stray from that norm is an essential strategy for getting ahead of potential data and system breaches. But choosing the right tools is only part of the effort. Without sufficient training, efficient deployment and a good response plan, attackers could gain the upper hand.

PLUS: Find signature reports, such as the *InformationWeek* Salary Survey, *InformationWeek* 500 and the annual State of Security report; full issues; and much more.

Newsletter

Want to stay current on all new *InformationWeek* Reports? Subscribe to our weekly newsletter and never miss a beat.

[Subscribe](#)