White Paper

Cisco Advanced Malware Protection Sandboxing Capabilities

What You Will Learn

How sandboxing is a key part of network security when it operates as an integrated component of a complete solution. Specifically, you will learn three things:

- How to get the most from sandbox technology
- Which tactics malware authors use to avoid detection
- Why modern sandboxing must function as part of an overall solution

Discover that sandboxing is only a first step towards full protection across the attack continuum. Find out how to get an unprecedented level of visibility, control, and protection with the Cisco® Advanced Malware Protection (AMP) solution.

Getting the Most from Sandbox Technology

The Cisco AMP solution stands apart from traditional antimalware technologies by providing augmented protection, improved visibility, and enhanced control. Cisco AMP uses an extensive infrastructure of sandboxes to analyze hundreds of thousands of files each day.

Benefit from this infrastructure in two specific ways.

First, search our database for previously inspected files. Because Cisco AMP processes hundreds of thousands of files every day, often times the diagnosis is already available. If the file has already been analyzed, access a full report of that information immediately and respond accordingly.

Second, upload files that are unknown for individual analysis and get results in minutes. After it is submitted, the file is put into a remote, protected sandbox environment. Analysis is fast, and the results are then cataloged for future queries.

In addition to the file disposition, see details regarding major indicators of malicious behavior. The sandbox catalogs antidebugging techniques and common malware behaviors, such as keystroke logging, data theft, application behaviors, and network activity, including packet capture file (PCAP) traces, and screen shots.

The following section details the overall sandbox infrastructure and substantiates the design choices that we made. Following that, the general benefits and drawbacks of sandbox technologies are discussed. These last two sections are vendor neutral and likely to be of independent interest.

The Cisco Sandbox Infrastructure and Design Choices

You'll experience fast analysis and results because the architecture is fully scalable. The Cisco sandbox infrastructure comprises a series of sandboxes hosted in a secure cloud. We can spin up additional systems to handle any increases in sample submission.

Access a number of cloud-based advantages over a locally hosted offering:

- **Detailed analysis in minutes:** Identify file dispositions in five minutes. When you upload your files to our remote sandbox environment, these submissions are placed in a queue. After completion, you receive the results and a detailed report about the file's disposition, potential impact on an environment, and other indicators of compromise.
- **Prepopulation and community sharing:** Save time and money, and access thousands of reports available in our online database without having to execute a sample. Cisco proactively feed hundreds of thousands of samples into the sandbox infrastructure. Given the volume of samples run through the sandbox infrastructure, it is likely that there is already a report on a particular file. View recently analyzed samples to better understand the threat landscape trends that are relevant for your organization. Our entire community benefits from this extensive repository of identified malware threats.
- **Infrastructure redundancies:** Stay up and running 24 hours a day with system redundancies. The Cisco sandbox resides on a multinode infrastructure. If one sandbox goes offline, files continue to be processed by the other available instances. In contrast, companies that build local sandboxes on single hardware platforms are constantly at risk of downtime. Without redundancies, you are at the mercy of a single system.
- **Hassle free set-up:** Access sandbox technology instantly. A cloud infrastructure enables you to access the Cisco AMP sandboxes immediately without any hardware setup or configuration.
- **Seamless configuration updates:** Run samples against the latest security information without having to augment the sandbox. Whenever we learn of a new technical vulnerability in a software application, we augment the sandbox configuration to include instances of this vulnerability. By augmenting the sandbox configuration, we can determine whether a suspected malware sample exploits a technical vulnerability in the application and better assess the potential scope of damage. It is much more difficult to make this type of change to a local sandbox.
- **Immediate software updates:** Get instant updates effortlessly thanks to the cloud infrastructure. Cloud hosting allows for immediate and seamless updates as opposed to the manual updates required with local hosting. This makes a big difference to you because malware authors are constantly inventing new methods for evading detection in a sandbox. The coevolution of threats and countermeasures will always be an issue, but our belief is that we can best mitigate against the risk by developing agile identification methods. The fourth section of this technical brief surveys mechanisms employed by malware authors for bypassing sandbox-based detection.

There are benefits to a locally hosted infrastructure, such as greater configuration control, but these six benefits far outweigh them.

Benefits of Sandbox Analysis

Traditional signature-based detection engines often miss a large number of today's threats. While signature detection is great for known malware, detecting new forms with signature profiles is extremely difficult.

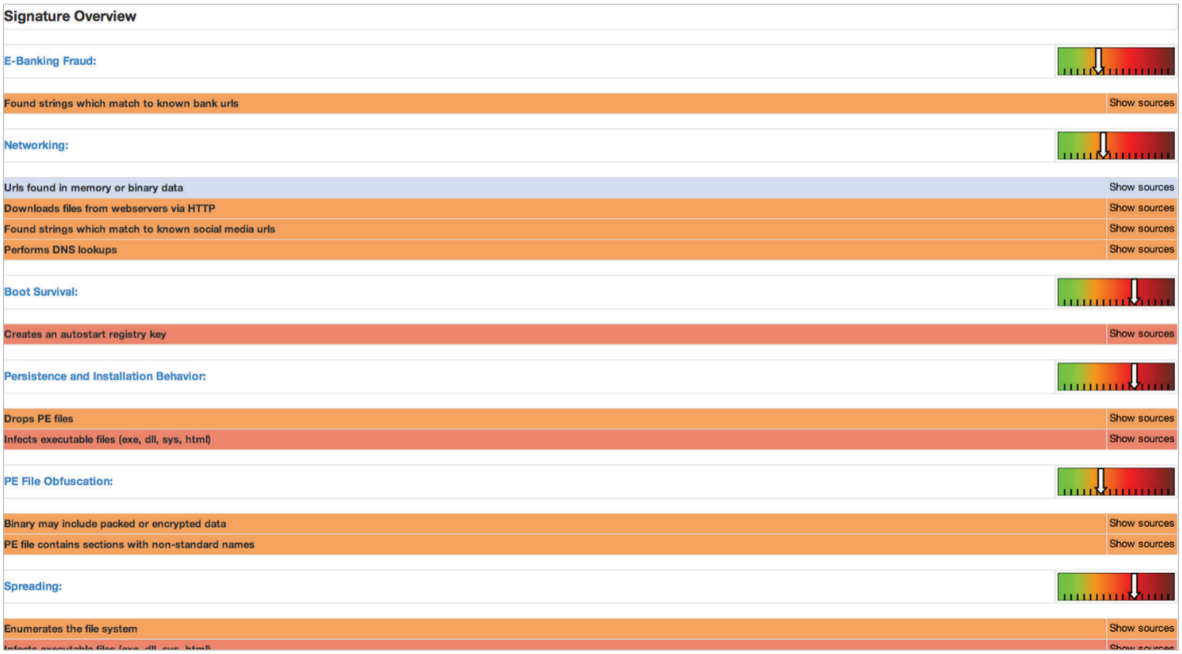
Polymorphic malware is one of the main reasons signatures are less effective today. Polymorphic threats change their appearance frequently, making signature-based detection a futile effort. In many cases, we see malware exhibit server-side, polymorphic behavior, meaning that new signatures are created for each victim.

Many malware authors use packers that act as turnkey mechanisms for making polymorphic threats. Some of them even have packer harnesses, which they use to keep repackaging threats until they evade detection by all the major vendors. Attackers test their malware on copies of the vendor's software and do not release their malware until it bypasses the exact defenses they are up against.

For this kind of attack, sandboxing technology is very useful. Sandboxes detonate unknown files in a safe environment and then record its actions. You can use the reports to identify whether a corresponding file appears to be malicious. Because packers change only the outer appearance of a threat, its underlying behavior generally stays the same. In a sandbox, two polymorphic copies of the same threat yield virtually identical reports. Therefore, sandboxing can be thought of as a noteworthy antidote to malware packers.

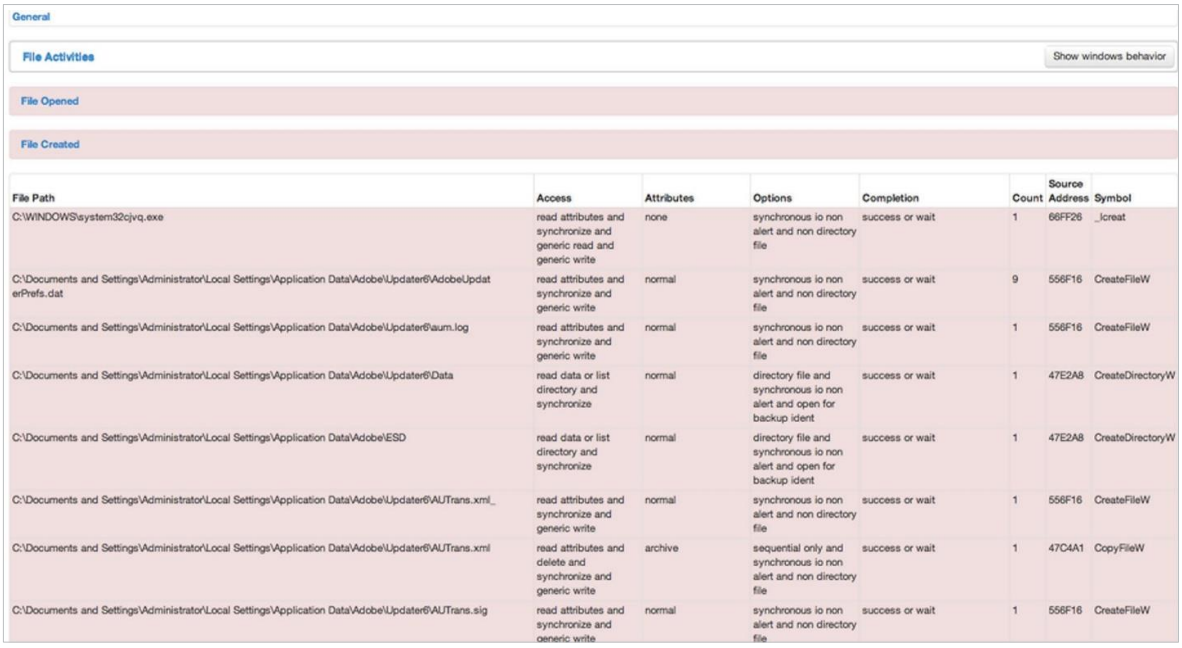
There are a number of specific attributes that our sandboxes examine. The first thing Cisco AMP looks for are obvious indicators of a threat, like antidebugging techniques or keystroke logging. Aside from that, it also searches for other suspicious activity, such as accessing specific registry keys, specific system files, or dynamically linked libraries (see Figure 1).

Figure 1. Sandbox Data Showing Common Suspicious Indicators



The attributes shown here are those that are intrinsic to a specific file. You can have accurate insight into the action that a certain file took in the sandbox. Cisco AMP shows you how the file executed on the system (that is, the startup process) and what other files it might have spawned (or downloaded and/or dropped by the original file into the system in question). In many cases, an unknown piece of malware intentionally downloads a known piece of malware. By identifying the known malware and then tracing its lineage, Cisco AMP can also identify the unknown piece of malware (see Figure 2).

Figure 2. Sandbox Analysis Showing Files Spawned by an Initial File



General

File Activities — Show windows behavior

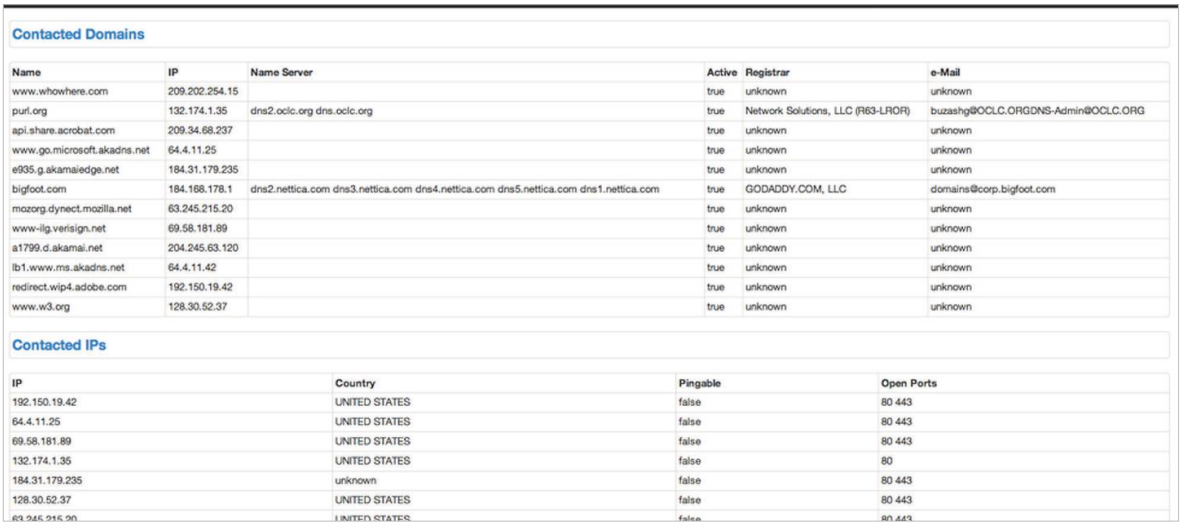
File Opened

File Created

File Path	Access	Attributes	Options	Completion	Count	Source Address	Symbol
C:\WINDOWS\system32cjvq.exe	read attributes and synchronize and generic read and generic write	none	synchronous io non alert and non directory file	success or wait	1	66FF26	_lcreat
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\Updater6\AdobeUpdaterPrefs.dat	read attributes and synchronize and generic write	normal	synchronous io non alert and non directory file	success or wait	9	556F16	CreateFileW
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\Updater6\aum.log	read attributes and synchronize and generic write	normal	synchronous io non alert and non directory file	success or wait	1	556F16	CreateFileW
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\Updater6\Data	read data or list directory and synchronize	normal	directory file and synchronous io non alert and open for backup ident	success or wait	1	47E2A8	CreateDirectoryW
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\ESD	read data or list directory and synchronize	normal	directory file and synchronous io non alert and open for backup ident	success or wait	1	47E2A8	CreateDirectoryW
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\Updater6\AUTrans.xml_	read attributes and synchronize and generic write	normal	synchronous io non alert and non directory file	success or wait	1	556F16	CreateFileW
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\Updater6\AUTrans.xml	read attributes and delete and synchronize and generic write	archive	sequential only and synchronous io non alert and non directory file	success or wait	1	47C4A1	CopyFileW
C:\Documents and Settings\Administrator\Local Settings\Application Data\Adobe\Updater6\AUTrans.sig	read attributes and synchronize and generic write	normal	synchronous io non alert and non directory file	success or wait	1	556F16	CreateFileW

Sandboxing also gathers essential data on network activity. You can examine the entire PCAP trace generated by our sandbox. This analysis report includes information about the most salient network characteristics. For example, if the file in question connects to a host that is known to be a botnet command and control server, then the file can be identified as malicious (see Figure 3). Along similar lines, Cisco AMP examines Domain Name System (DNS) and HTTP traffic for botnet characteristics and use this traffic to determine the disposition of a particular file. Network analysis can be taken a step further by examining an overall HTTP hierarchy graph, which can be used to identify web exploit kits and how they connect to the sites they infect.

Figure 3. Sandbox Data Showing Domains Connected to a Binary of Interest



Contacted Domains

Name	IP	Name Server	Active	Registrar	e-Mail
www.whowhere.com	209.202.254.15		true	unknown	unknown
purl.org	132.174.1.35	dns2.oclc.org dns.oclc.org	true	Network Solutions, LLC (R63-LROR)	buzashg@OCLC.ORGDNS-Admin@OCLC.ORG
api.share.acrobat.com	209.34.68.237		true	unknown	unknown
www.go.microsoft.akadns.net	64.4.11.25		true	unknown	unknown
e935.g.akamaiedge.net	184.31.179.235		true	unknown	unknown
bigfoot.com	184.168.178.1	dns2.nettica.com dns3.nettica.com dns4.nettica.com dns5.nettica.com dns1.nettica.com	true	GODADDY.COM, LLC	domains@corp.bigfoot.com
mozorg.dynect.mozilla.net	63.245.215.20		true	unknown	unknown
www-ilg.verisign.net	69.58.181.89		true	unknown	unknown
a1799.d.akamai.net	204.245.63.120		true	unknown	unknown
lb1.www.ms.akadns.net	64.4.11.42		true	unknown	unknown
redirect.wip4.adobe.com	192.150.19.42		true	unknown	unknown
www.w3.org	128.30.52.37		true	unknown	unknown

Contacted IPs

IP	Country	Pingable	Open Ports
192.150.19.42	UNITED STATES	false	80 443
64.4.11.25	UNITED STATES	false	80 443
69.58.181.89	UNITED STATES	false	80 443
132.174.1.35	UNITED STATES	false	80
184.31.179.235	unknown	false	80 443
128.30.52.37	UNITED STATES	false	80 443
63.245.215.20	UNITED STATES	false	80 443

Sandboxes help address many of the weaknesses of signature-based detection, so you can see exactly what a file does before it is labeled malicious or benign. Although this functionality is extremely valuable, it is not without specific limitations.

Limitations of Sandbox Analysis

Sandboxes are not a “silver bullet” when it comes to comprehensive malware protection. They can be used to address a portion of the problem, but have three deficiencies:

- **Inherent efficacy:** Running a file in a sandbox is no guarantee that the disposition will show the threat it poses to your environment.
- **Evasion Tactics:** Malware authors deploy a number of techniques to bypass sandbox analysis.
- **Means to an end, not an end itself:** Sandboxing is a great tool for addressing malware in an environment, but sandboxing needs to be coupled with other capabilities to provide comprehensive malware protection.

Let’s take a look at the three limitations in greater detail.

Inherent Efficacy

The truth is that sandboxes cannot determine the disposition of every piece of malware in an isolated environment for a number of reasons:

- **Lack of execution:** Typically, a given file will not execute in a sandbox. Today's malware often infects a system piece by piece and perform its function only when totally assembled. Without the other parts, a discrete part of the malware cannot execute in a sandbox. Alternatively, there may be subtle configuration issues that exist in a sandbox but not on an actual end-user system.
- **Sandbox compared to a real box:** In the long run, you want to know what a piece of malware did, or could have done, on end-user systems (real box). A sandbox environment can loosely approximate a real-box system, but it will never be a perfect replica. A vulnerable program might be installed on a sandbox machine that isn't installed in the actual environment or vice versa. This can lead to malware identification that actually does not have an impact on a system. In general, the actions or inactions exhibited in a sandbox might not match those of a real box, which is what, ultimately, you care about.
- **Malicious actions compared to malicious intent:** Even if a sample runs perfectly in a sandbox, this information itself might not be enough to conclude that it is not malicious. In many cases, illegitimate software applications behave in similar ways to legitimate ones. For example, legitimate desktop support software allows IT administrators to log in remotely to a system and execute commands in much the same way that a malicious remote access trojan (RAT) performs the same action. Legitimate instant-messaging software connects with the Windows keyboard API to collect and transmit keystrokes to a remote server. A malicious keystroke-logging trojan does the same thing. In both cases, diagnosing the actions taken in a sandbox are not sufficient because intent is the key component. Understanding intent is not something that can be readily done through algorithmic means. Because sandbox technologies want to avoid false positives, they err on the side of safety and fail to stop malware when there is even a remote chance that it is possibly nonmalicious.

Notice that all previous points assume that the sandbox received the correct file in the first place. This assumption does not always hold. For example, if a network device contains a sandbox, then it has to be able to parse network traffic and extract any transmitted file content. If network traffic is encrypted, then the file cannot be extracted. Even if the network traffic is in the clear, the sandbox must be able to parse the protocol being used to transmit the content.

Evasion Tactics

Up to this point, we have discussed how malware can inadvertently bypass sandbox defenses. The real problem is when a motivated attacker develops malware with the specific ability to bypass industry detection methods. Attackers use these techniques when they are targeting specific companies. As more and more organizations deploy sandbox technologies, we expect to see an uptick in sandbox evasion tactics. While a comprehensive analysis of evasion techniques is beyond the scope of this technical brief, we want to provide a high-level discussion of some of the main techniques.

- **Sleep and time trigger techniques:** An attacker might build a piece of malware with a sleep timer for a specific amount of time, for example, 5 to 10 minutes, or a specific event trigger. The malware is activated only after the specified amount of time has passed or the particular event takes place. A typical sandbox can afford to analyze malware only for a short period of time, because in the world of security, time is of the essence. If no malicious behaviors are observed during this time, the malware is ignored and remains undetected.

- **Human detection techniques:** Because sandboxes are artificial environments that detonate an unknown file for automated observation, people very rarely are involved in the process. Because of this, malware authors employ techniques to test for the presence of a person observing the sandbox. This might be a protocol that counts and records the number of mouse clicks. A real user almost certainly uses a mouse, while an automated environment does not use a mouse to execute commands. If the malware detects the presence of mouse clicks, it continues to operate as normal because it assumes a person is monitoring the sandbox. Conversely, if a piece of malware does not detect mouse clicks, it concludes that it is in a sandbox environment and remains dormant. To combat this, Cisco AMP artificially generates mouse clicks in the sandbox, but the result is a back-and-forth war of attrition that never ends.
- **Implementation-specific techniques:** Attackers also design malware to bypass specific sandbox environments. No matter what sandbox technology you use, there are always revealing signs of its presence. The way a sandbox is executed on a system, both the file layout and registry keys identify what type of sandbox a company uses to defend its networks. Attackers build malware that initially searches for these indicators and behaves innocently when they find them.

Means to an End

As you can tell from our discussion, sandboxing is far from a “silver bullet” when it comes to detecting malware. A motivated attacker can bypass any approach that focuses purely on detection. Targeted attacks are increasing in frequency, and it is much easier to simply bypass defenses. Most of the time, a malware author has a deep knowledge of the exact security technologies deployed in an organization and then tailors a program to bypass it.

Even if a sandbox does a good job in identifying or running malware, it is only a starting point if you are interested in complete security. There are critical considerations that highlight the scope of sandboxing technologies:

- **Point in time:** Sandbox technologies observe the behaviors of one sample at a time. In practice, however, it is important to understand the broader ecosystem surrounding a piece of malware. For starters, on an actual end-user system, you want to know how the malware got on the device in the first place. What technical vulnerability did it exploit to enter the system? Was the malware downloaded by a dropper application? Did the user double-click and activate a hidden link? Did the malware install anything else on your system? Sandboxing technology cannot provide the answers to these critical questions. It can help start the process, but it is important to investigate these issues further.
- **Organizational impact:** Sandboxes are designed to identify malware and the actions it might take after entering an environment. However, your incident response teams need answers to questions beyond what a piece of malware does. How far does the infection go? Where did it originate? Are there other machines that are compromised? How did it enter the system? These questions also apply to components associated with a piece of malware. If the original malware is connected to a given host, then the incident response team may want to know what other end-user systems also visited the infected host. If the malware downloaded a new application to a device, you need to know what that copy has done as well. Sandboxing does not answer all of these crucial threat response questions.
- **Remediation:** While you want to know what a threat did (or might have done), you are even more interested in removing any instances of that threat from the systems in their environment. Sandboxes excel at arming you with important data, but they do not help with the actual cleanup and containment phase of an attack.

Conclusion

Get fast analysis, community sharing, and minimal downtime with no setup time, no configuration updates, and zero software updates with our cloud-based sandbox infrastructure. Sandboxing is a powerful component that helps address the limitations of traditional antimalware signatures, but it's important to understand that it cannot stand on its own. Even the best sandbox can miss malware, and it addresses only one portion of the problem space. For these reasons, we provide a suite of additional capabilities that are designed to provide you with comprehensive protection against advanced malware. We believe that you are best served, not by any point technology, but by a comprehensive portfolio of technologies that can work in concert to address the threats they face.

With Cisco AMP technology, you can benefit from a holistic set of capabilities for addressing the threats that you face. This technical brief describes one of those capabilities, sandboxing, both its strengths and its shortcomings. You benefit from the extensive volume of samples that we analyze each day, in addition to your ability to submit files for analysis to the Cisco extensive back-end sandbox infrastructure.

Armed with the results of Cisco AMP analysis, you can use other capabilities through the Cisco AMP solution, such as trajectory, threat root cause analysis, custom signatures, and retrospective remediation, which all help address malware attacks in your environments. Get exceptional protection with lower operational costs.

For More Information

If you are interested in learning more about the Cisco AMP technology suite, see our Cisco AMP Buyer's Guide.

About Cisco

Cisco (NASDAQ: CSCO) is the worldwide leader in IT that helps companies seize the opportunities of tomorrow by proving that amazing things can happen when you connect the previously unconnected. Cisco provides one of the industry's most comprehensive advanced threat protection portfolios, as well as a broad set of enforcement and remediation options that are integrated, pervasive, continuous, and open. This threat-centric security model lets defenders address the full attack continuum across all attack vectors—before, during, and after an attack. For ongoing news, go to http://www.cisco.com/go/security

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Printed in USA C78-733277-00 11/14