Detecting Internal Malware Spread with the Cisco Cyber Threat Defense Solution 1.0

April 9, 2012

Introduction

One of the most common network defense strategies is to construct a hardened network perimeter to prevent attacks from reaching the internal network. While this is a good starting point, malware writers have adapted their techniques to compensate for this strategy, using polymorphic malware, encryption, obfuscation, and social engineering to bypass these traditional security measures. As a result, enterprise security professionals are now battling malware on their internal networks, where traditional perimeter defense technologies are difficult to deploy and do not scale well.

Because perimeter defenses can be easily bypassed, security practitioners may find themselves in the difficult position of having to balance the usability of the network against the scalability, manageability, and effectiveness of the security infrastructure. Further hardening of the network perimeter quickly reaches a point of diminishing return—and often proves frustrating for internal network users who need seamless access to perform their jobs. Deploying traditional security products like firewalls and IPSs pervasively on the internal network is costly and difficult to manage. What is needed is a scalable, high-performance, cost-effective way to monitor internal network traffic and identify malware that has penetrated the perimeter and is spreading internally.

The Cisco® Cyber Threat Defense Solution addresses the problem of internally spreading malware by providing the visibility mechanisms needed for detecting it. The solution integrates Lancope® StealthWatch® with Cisco's hardware-supported NetFlow and Identity Services Engine, providing a simple and effective way to address the problem of detecting and managing internally spreading malware.

Prerequisites

This document assumes the reader has read the Cisco Cyber Threat Defense Solution 1.0 Overview, Design and Implementation Guide, and the Introduction to Cisco Cyber Threat Defense "how-to" document. Readers will gain the maximum benefit from the examples in this guide if they have installed a fully functioning Cyber Threat Defense Solution, including a switch and router infrastructure that is properly configured for sending NetFlow, a fully functioning Cisco Identity Services Engine environment, and a StealthWatch® FlowCollector and StealthWatch® Management Console. With these in place, security practitioners should then plan on following the step-by-step examples while in front of the StealthWatch® console.

Solution Components

The Cisco Cyber Threat Defense Solution 1.0 is composed of three integrated components:

NetFlow data generation devices. NetFlow is the de facto standard for acquiring IP operational data. Traditional IP NetFlow defines a flow as a unidirectional sequence of packets that arrive at a router on the same interface or sub-interface and have the same source IP address, destination IP address, Layer 3 or 4 protocol, TCP or UDP source port number, TCP or UDP destination port number, and type of service (ToS) byte in their TCP, UDP, and IP headers, respectively.

Flexible NetFlow is the next generation in flow technology and is a particularly valuable component of the Cisco Cyber Threat Defense Solution 1.0. Flexible NetFlow optimizes the network infrastructure, reducing operation costs and improving capacity planning and security incident detection with increased flexibility and scalability.

NetFlow can be enabled on most Cisco switches and routers, as well as some Cisco VPN and firewall devices. In addition, select devices now employ special hardware acceleration, ensuring that the NetFlow data collection process does not impact device performance. This enables NetFlow data collection pervasively throughout the network—even down to the user edge—so that every packet from every network segment and every device is completely visible.

Cisco Identity Services Engine. The Identity Services Engine delivers all the necessary identity services required by enterprise networks—AAA, profiling, posture, and guest management—in a single platform. In the context of the Cisco Cyber Threat Defense Solution 1.0, the Identity Services Engine can be deployed as either a network appliance or virtual machine and answers the "who" (user), "what" (device), and "where" (which NetFlow-enabled device) questions that tie network flow data to the actual physical network infrastructure.

In an enterprise deployment, the Identity Services Engine provides the central policy enforcement needed to govern a network. The Identity Services Engine can provision and deliver cross-domain application and network services securely and reliably in enterprise wired, wireless, and VPN environments. This policy-based service enablement platform helps ensure corporate and regulatory compliance, enhances infrastructure security, and simplifies enterprise service operations. The Identity Services Engine can gather real-time contextual information from the network, users, and devices and make proactive governance decisions by enforcing policy across the network infrastructure.

Lancope® StealthWatch® system. This NetFlow visibility, network performance, and threat detection solution provides an easy-to-use interface that enables both monitoring and detailed forensics. The solution is composed of two core components: the StealthWatch® Management Console and one or more StealthWatch® FlowCollectors. Additional optional components include a StealthWatch® FlowSensor and a StealthWatch® FlowReplicator.

Operating Concepts

The Cisco Cyber Threat Defense Solution 1.0 detects internally spreading malware by using a set of rules within StealthWatch® that detects *replication activity*. These rules consist of three parts:

- Rules that watch for reconnaissance activity
- Rules that determine if hosts that were the target of reconnaissance in fact responded to the scan (a *touched* host)
- Rules that determine if that host subsequently exhibits the same type of behavior as the original host that scanned it

When a host matches all of these rules, it is considered a host that is spreading malware, and is tracked by the Cisco Cyber Threat Defense Solution 1.0.

Determining Which Flow Traffic Is "Visible"

Within an enterprise, when two endpoints communicate across a network connected by NetFlow-enabled network access devices, the NetFlow-enabled devices generate NetFlow data that describes the characteristics of that flow. An endpoint in this network can either be inside or outside the boundaries of the corporate network.

Note: In this document, the terms "endpoint," "computer," "host," and "device" are used interchangeably to mean any network-attached system.

The visibility of a flow is subsequently defined by the endpoint's location, expressed as one of four possible "classes" of flows:

		From	
		Inside	Outside
To	Inside	Visible	Visible
	Outside	Visible	Not visible

To summarize, the Cisco Cyber Threat Defense Solution 1.0 has visibility to flows that are "inside to inside," "inside to outside," and "outside to inside." Because the solution is deployed inside the internal network, it typically would have no visibility to traffic that is "outside to outside."

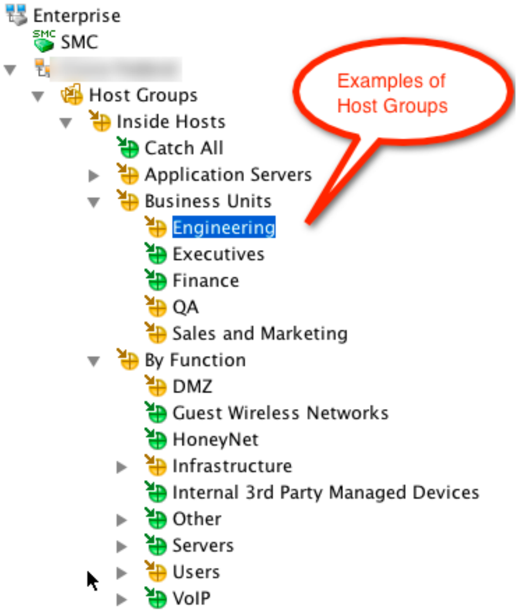
Flows Have "Direction"

Furthermore, flows have a *direction* that the Cisco Cyber Threat Defense Solution 1.0 is able to discern. Consider the following example: A customer who has deployed the Cisco Cyber Threat Defense Solution 1.0 in their enterprise network, "acme.com," initiates a web browser request to http://www.cisco.com/go/security, an external domain. This communication will be captured by the Cisco solution as an inside-to-outside flow because it was initiated by a client inside the boundaries of acme.com and destined to a device outside of the corporate network at cisco.com.

Subsequently, clients making connections to other devices within acme.com would be classified as "inside to inside," and any device outside the enterprise that could initiate a connection to a host inside the enterprise (for example, in a DMZ) would represent an outside-to-inside connection.

Flows Have Behavior Profiles That Can Be "Baselined"

The Cisco Cyber Threat Defense Solution 1.0 captures and analyzes historical data about flows on the network and uses this to create a *baseline* of behavior. To create this baseline, a user first describes the various zones of their network to the StealthWatch® Management Console. These zones, called *host groups* in StealthWatch®, can be described in many different ways to conform to an enterprise's unique requirements. Host groups are often defined by IP address ranges or VLANs, and might have descriptions such as "Server Farm," "Data Center," "VPN IP Address Pool," "Engineering Clients," or "IP Phones."



Examples of Host Groups

The Cisco Cyber Threat Defense Solution 1.0 automatically creates *behavior profiles* of all hosts, tracking many different parameters of the flows over a period of seven days. This creates the initial behavior baseline for the host. Cisco's Cyber Threat Defense Solution 1.0 automatically creates *behavior profiles* of all hosts, tracking many different parameters of the flows. The initial behavior baseline for the host is built within the first seven days of monitoring. Subsequent to those first seven days, the solution continues its baselining functions an additional 21 days to create a 28-day *rolling baseline* for the host. This baseline is used to trigger alarms when host begins to change in behavior. All 28 days of statistical information is used to trigger alarms, but the most recent 7 days are more heavily weighted

Network Reconnaissance: A Deeper Look

Unless an attacker already has intimate knowledge of a network, one of the first tasks they will want to perform is to map the network to determine what devices they can attack. There are many different ways to accomplish this.

One of the earliest reconnaissance methods was simply to sequentially "ping" every IP address on a network, starting with the local subnet, and then expand outward. If an IP address responded to a ping, the attacker knew there was a device active at that IP address, and would add it to a locally list of potential attack targets. This "ping" method would require the attacker to "guess" what subnets existed on the network.

This proved to be a very noisy method of identifying vulnerable hosts. Every time a ping was sent to a subnet, the router for that subnet would generate a Layer 2 Address Resolution Protocol (ARP) request for the target IP address. A router-generated ARP request says: "I have someone looking for IP address a.b.c.d. If you are that IP address, please respond with your MAC address so I can forward this packet to you." A host at that IP address would send an ARP reply that included its MAC address, and then the router would use that MAC address to forward the packet(s). However, if a ping "missed" a target IP address because it was not active on a network, no ARP reply would occur.

On most networks, a sizable amount of the IP address space is unused. A device performing reconnaissance would generate a large number of ARP requests but receive fewer ARP responses, which would result in an ARP "imbalance." Simple tools that looked for ARP requests with no ARP replies could be used to detect network reconnaissance. Over time, this simple capability of looking for unfulfilled ARP requests has proven to be a reliable method of reconnaissance detection.

To adapt, attackers have become more sophisticated. They have learned to slow down the speed of their reconnaissance in order to "hide" the reconnaissance in background "noise" of the local network, to make it indistinguishable from other network activity. This significantly slows down the process of network reconnaissance, but it improves an attacker's chance of not being detected.

Another strategy is to send other kinds of packets, such as UDP or TCP packets, to highly randomized addresses. Instead of sequentially sending packets to each IP address in a range, hackers randomize the destinations and disguise them to look more like normal network traffic. Using specialized software, an attacker can even bypass the normal network stack to send custom crafted packets—packets that might, for example, contain illegal flag conditions such as "SYN/FIN."

This technique provided a way to evade IDS products that had not adapted to this possibility, and allowed the attacker to study the responses. Attackers learned that different network stack vendors would respond to these strange flag conditions in different ways. If you had a list of how each operating system would respond, an attacker could "guess" with fairly high accuracy the kind of host at the other end—Windows, Unix, Linux, Mac OS, etc.—and would allow the infected host to target only those types of devices most vulnerable to attack.

In summary, network reconnaissance can take many different forms. Being able to recognize these forms is key for early detection of network-based threats.

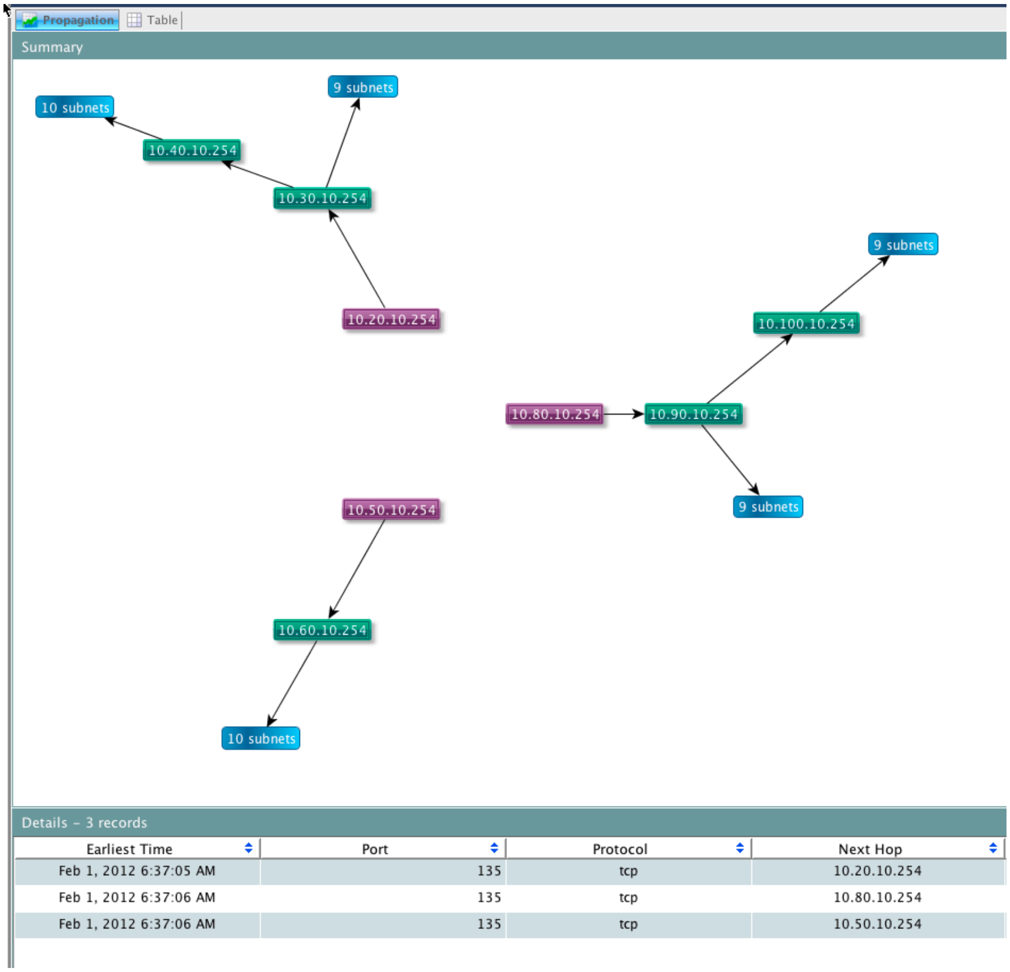
"Touched" Hosts and Malware Spread

If a host is identified as scanning the network, StealthWatch® checks that host's communication to determine if any of the hosts it tried to communicate with actually responded. If they do, StealthWatch® then marks the target hosts as *touched*, and tracks them as well.

If a *touched* host begins to exhibit similar reconnaissance behavior (same ports, protocols, and patterns) as the original scanning host, StealthWatch® marks the original source host as exhibiting *worm activity* and considers the host to have *propagated*.

Note: The term "worm activity" should be considered a general term for any malware that exhibits self-replicating behavior with self-similar flow characteristics, whether or not it strictly conforms to the definition of a worm.

Finally, StealthWatch® puts all of this together in a graphical interface that shows the security practitioner the originally infected host, the devices it has attempted to communicate with, those that have responded, and those that are now operating with the same behavior as the originally infected host.



Details – 3 records

Earliest Time	Port	Protocol	Next Hop
Feb 1, 2012 6:37:05 AM	135	tcp	10.20.10.254
Feb 1, 2012 6:37:06 AM	135	tcp	10.80.10.254
Feb 1, 2012 6:37:06 AM	135	tcp	10.50.10.254

Figure 1 Worm Tracker Interface

Malware Spread Example

The following example steps the user through the process of identifying internally spreading malware using the Cisco Cyber Threat Defense Solution 1.0.

A Note About Concern Indexes

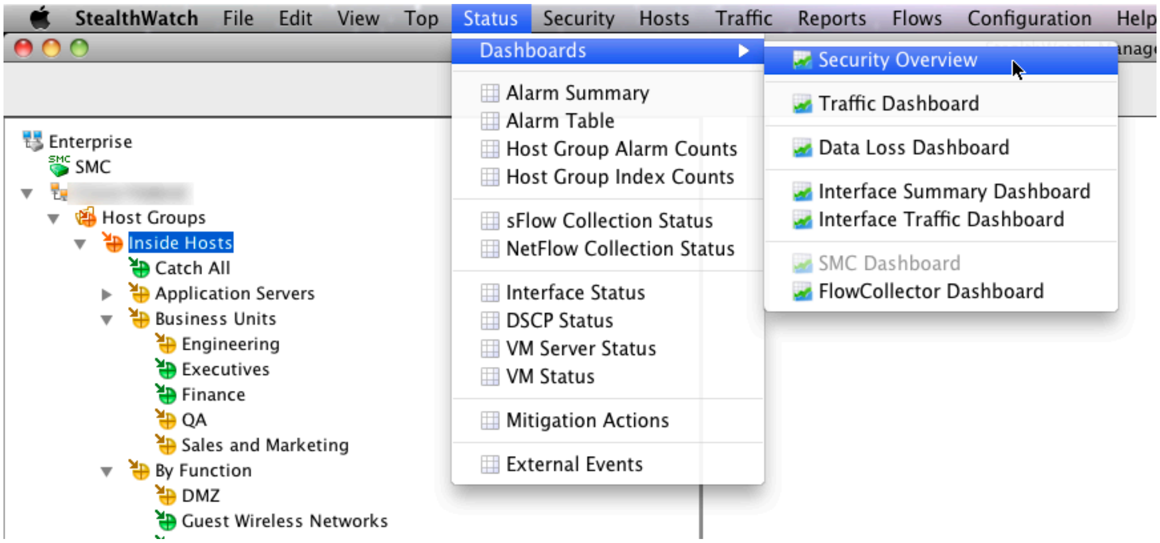
The Cisco Cyber Threat Defense Solution 1.0 provides the ability to detect internally spreading malware based on the analysis of NetFlow data. The StealthWatch® console that is used to view the NetFlow data uses a technology called a *concern index* to reflect the severity of a security event. A *concern index* is a numeric value—a counter of sorts—indicating how many times a specific kind of event has occurred within a window of time. The StealthWatch® detection engine examines each flow as it enters the FlowCollector and then applies a set of rules to each flow. The result of the comparison between the rule and the flow data determines whether various counters should be increased.

Independent of the collection process, StealthWatch® also constantly analyzes these flows to determine if thresholds have been exceeded or suspicious patterns have been detected. When a concern index value exceeds a defined threshold, StealthWatch® raises an alarm, indicating a potential problem. In this document, we explore using concern indexes to detect internally spreading malware.

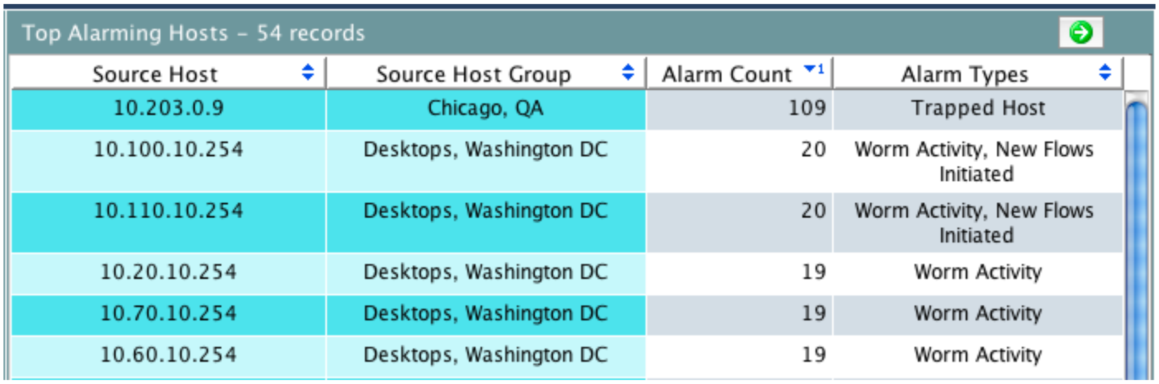
Identifying Internally Spreading Malware

In this section, we explore how to use the StealthWatch® interface to identify flows representing internally spreading malware.

Step 1 From the main menu, select Status ➔ Dashboards ➔ Security Overview.



This displays the Security Overview Dashboard. The panel in the upper-right corner is the Top Alarming Hosts table, as shown below. In this example, there are several active *worm activity* alarms.

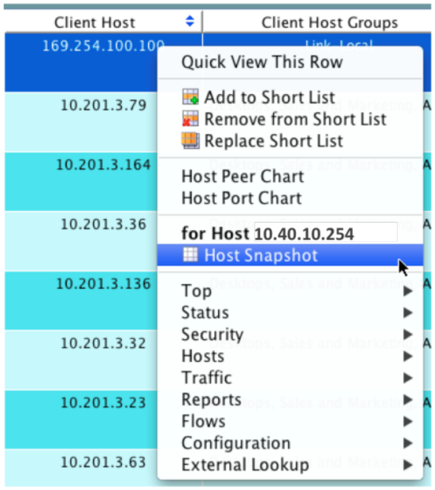

Top Alarming Hosts – 54 records

Source Host	Source Host Group	Alarm Count	Alarm Types
10.203.0.9	Chicago, QA	109	Trapped Host
10.100.10.254	Desktops, Washington DC	20	Worm Activity, New Flows Initiated
10.110.10.254	Desktops, Washington DC	20	Worm Activity, New Flows Initiated
10.20.10.254	Desktops, Washington DC	19	Worm Activity
10.70.10.254	Desktops, Washington DC	19	Worm Activity
10.60.10.254	Desktops, Washington DC	19	Worm Activity

Step 2 Select an IP address from the list.

Step 3 Right-click on the IP.

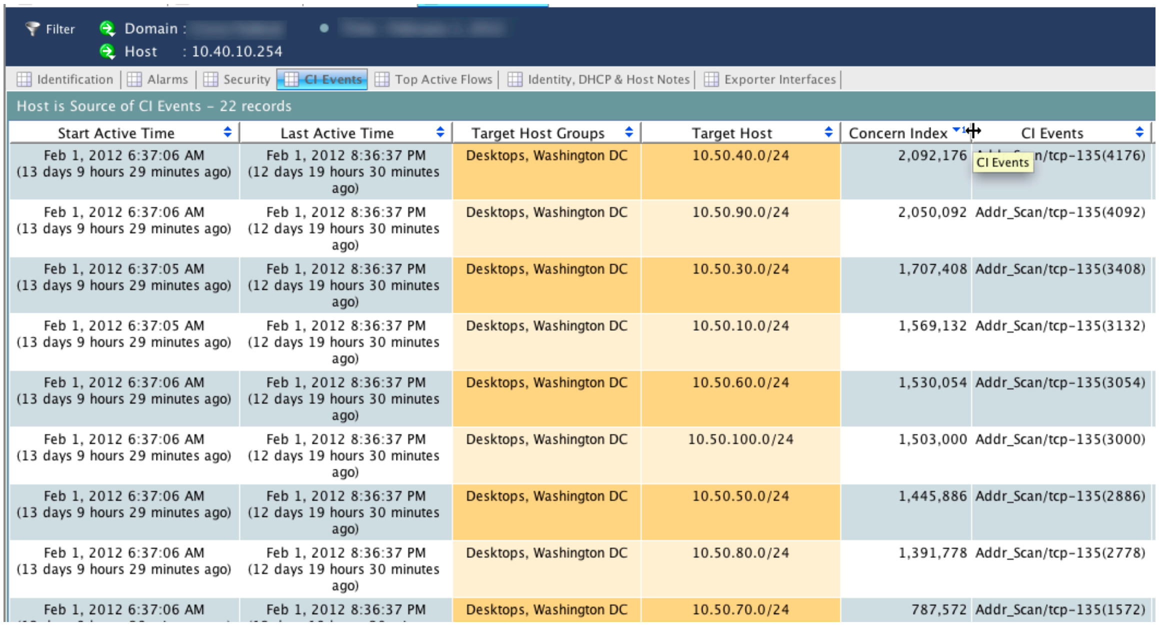
Step 4 From the popup menu, select *Host Snapshot*.



This produces the Host Snapshot for this host (10.40.10.254).

Step 5 Click on the *CI Events* tab to show the other hosts this host has attempted to communicate with.

You will notice the *CI Events* column of this table displays address-scanning activity for every host. This means the source host (10.40.10.254) is scanning the network.



Domain : [redacted] [redacted]
Host : 10.40.10.254

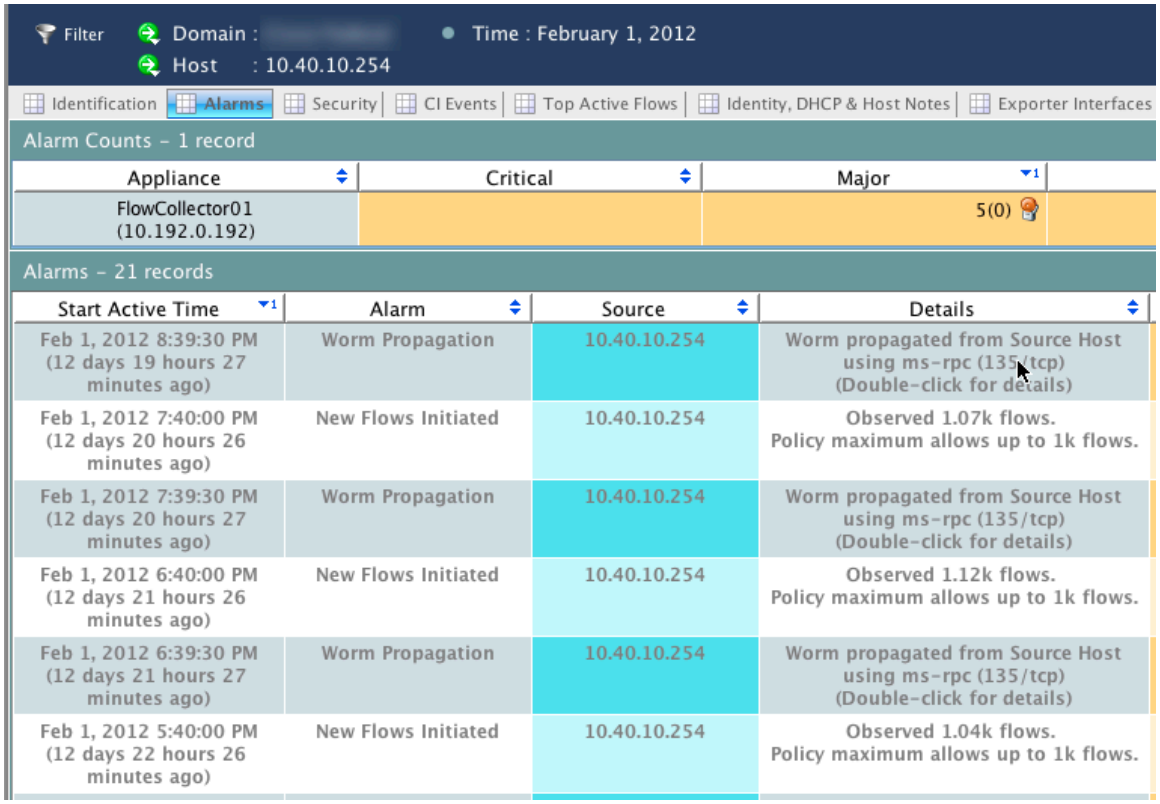
Host is Source of CI Events – 22 records

Start Active Time	Last Active Time	Target Host Groups	Target Host	Concern Index	CI Events
Feb 1, 2012 6:37:06 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.40.0/24	2,092,176	Addr_Scan/tcp-135(4176)
Feb 1, 2012 6:37:06 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.90.0/24	2,050,092	Addr_Scan/tcp-135(4092)
Feb 1, 2012 6:37:05 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.30.0/24	1,707,408	Addr_Scan/tcp-135(3408)
Feb 1, 2012 6:37:05 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.10.0/24	1,569,132	Addr_Scan/tcp-135(3132)
Feb 1, 2012 6:37:06 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.60.0/24	1,530,054	Addr_Scan/tcp-135(3054)
Feb 1, 2012 6:37:06 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.100.0/24	1,503,000	Addr_Scan/tcp-135(3000)
Feb 1, 2012 6:37:06 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.50.0/24	1,445,886	Addr_Scan/tcp-135(2886)
Feb 1, 2012 6:37:06 AM (13 days 9 hours 29 minutes ago)	Feb 1, 2012 8:36:37 PM (12 days 19 hours 30 minutes ago)	Desktops, Washington DC	10.50.80.0/24	1,391,778	Addr_Scan/tcp-135(2778)
Feb 1, 2012 6:37:06 AM	Feb 1, 2012 8:36:37 PM	Desktops, Washington DC	10.50.70.0/24	787,572	Addr_Scan/tcp-135(1572)

The next step is to check if additional alarms were raised by this scanning activity.

Step 6 Click on the *Alarms* tab.

If StealthWatch® has determined that a host has responded to the reconnaissance scans, it will be reported as *worm activity* as shown below in the *Details* column.



Domain : [redacted]
Host : 10.40.10.254
Time : February 1, 2012

Alarm Counts - 1 record

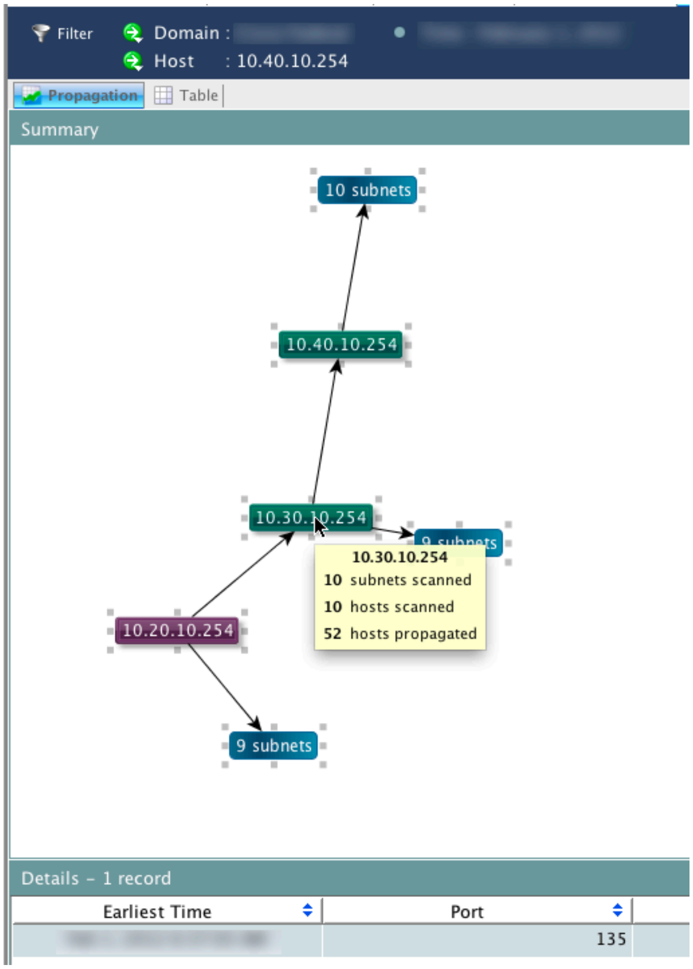
Appliance	Critical	Major
FlowCollector01 (10.192.0.192)		5(0)

Alarms - 21 records

Start Active Time	Alarm	Source	Details
Feb 1, 2012 8:39:30 PM (12 days 19 hours 27 minutes ago)	Worm Propagation	10.40.10.254	Worm propagated from Source Host using ms-rpc (135/tcp) (Double-click for details)
Feb 1, 2012 7:40:00 PM (12 days 20 hours 26 minutes ago)	New Flows Initiated	10.40.10.254	Observed 1.07k flows. Policy maximum allows up to 1k flows.
Feb 1, 2012 7:39:30 PM (12 days 20 hours 27 minutes ago)	Worm Propagation	10.40.10.254	Worm propagated from Source Host using ms-rpc (135/tcp) (Double-click for details)
Feb 1, 2012 6:40:00 PM (12 days 21 hours 26 minutes ago)	New Flows Initiated	10.40.10.254	Observed 1.12k flows. Policy maximum allows up to 1k flows.
Feb 1, 2012 6:39:30 PM (12 days 21 hours 27 minutes ago)	Worm Propagation	10.40.10.254	Worm propagated from Source Host using ms-rpc (135/tcp) (Double-click for details)
Feb 1, 2012 5:40:00 PM (12 days 22 hours 26 minutes ago)	New Flows Initiated	10.40.10.254	Observed 1.04k flows. Policy maximum allows up to 1k flows.

If you double-click any *Details* cell, you will launch a specialized dashboard in StealthWatch® called the *Worm Tracker*.

Step 7 Double-click a cell in the *Details* column.



Domain : [redacted] [redacted]
Host : 10.40.10.254

10.30.10.254
10 subnets scanned
10 hosts scanned
52 hosts propagated

Details – 1 record

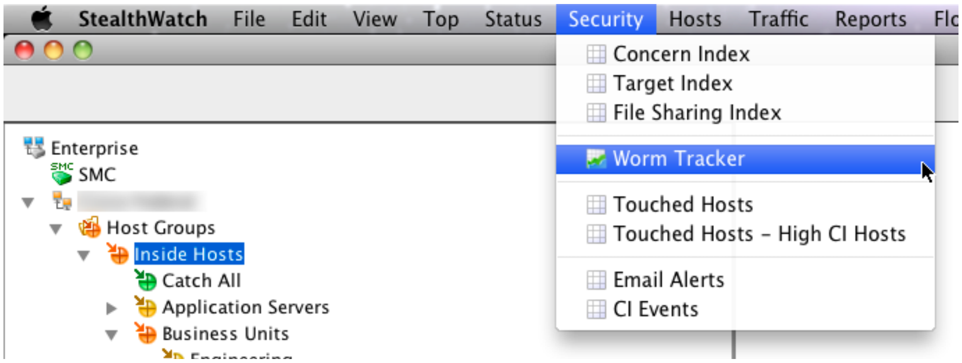
Earliest Time	Port
[redacted]	135

On this screen you will notice a graphical representation of the host and an icon representing the number of subnets it has communicated to. The table below it will show the times of each communication, the port involved, the protocol, next hop, and the total number of hosts in the subnet it attempted to communicate with.

Using the Worm Tracker Document

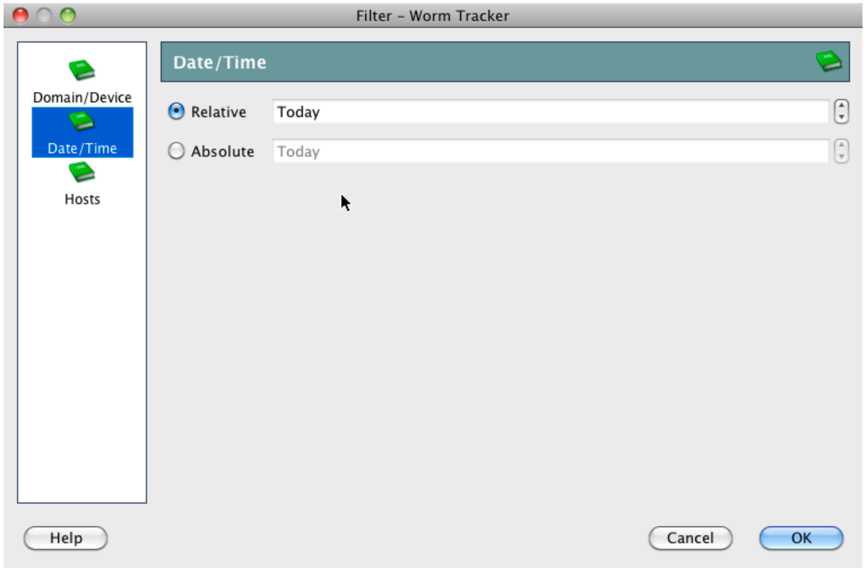
The previous procedure showed the process of identifying internally spreading malware. StealthWatch® includes a dashboard created specifically for this purpose—the *Worm Tracker Document*—that greatly speeds the process of identifying internally spreading malware.

Step 1 Click Security ➔ Worm Tracker.



Before StealthWatch® can display any information, it needs to have the filter conditions for the data set configured.

Step 2 Select the *Domain/Device*, *Time*, and *Host* parameters that interest you, and click *OK*.

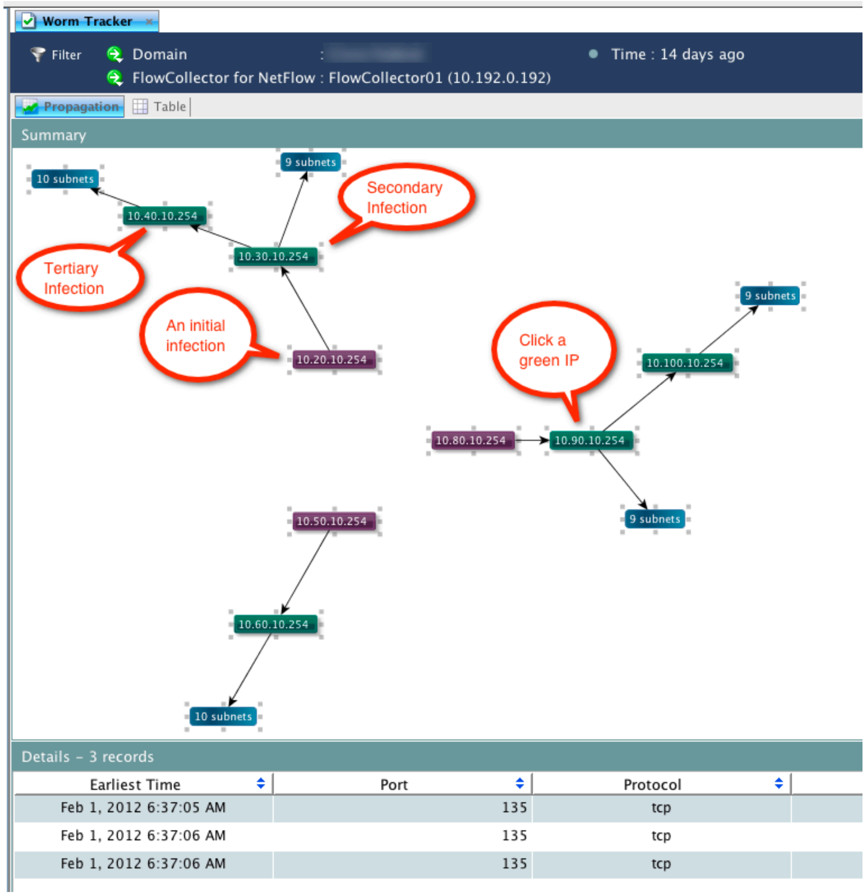


Date/Time

● Relative Today

○ Absolute Today

The resulting dashboard takes you to the Worm Tracker screen, which shows the relationships of the worm activity it is tracking for the time period you selected.



Domain : [redacted]

Time : 14 days ago

FlowCollector for NetFlow : FlowCollector01 (10.192.0.192)

Details - 3 records

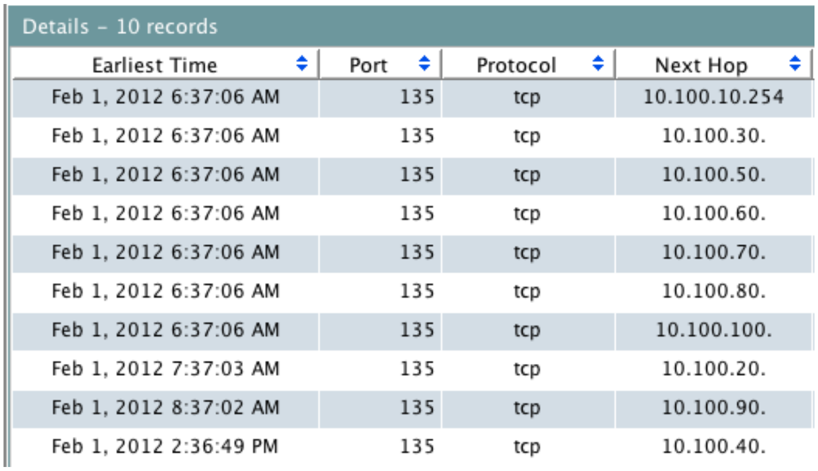
Earliest Time	Port	Protocol
Feb 1, 2012 6:37:05 AM	135	tcp
Feb 1, 2012 6:37:06 AM	135	tcp
Feb 1, 2012 6:37:06 AM	135	tcp

This is a clickable map that can be rearranged and that will show you detail if you double-click any hosts on the map. The table below the map provides additional detail.

Step 3 Click on a green IP address in the diagram. For this example, we have chosen 10.90.10.254.

Note: Purple IP addresses represent the initial infecting device. The green IP addresses represent targets of the malware that are in turn now re-propagating the malware. Blue icons represent the number of subnets an infected device has scanned.

The table below shows detailed information for this host. Notice that this malware is always attempting to propagate using TCP port 135. If you click on any of the other IP addresses in this map, you will see the corresponding propagation activity.



Details - 10 records

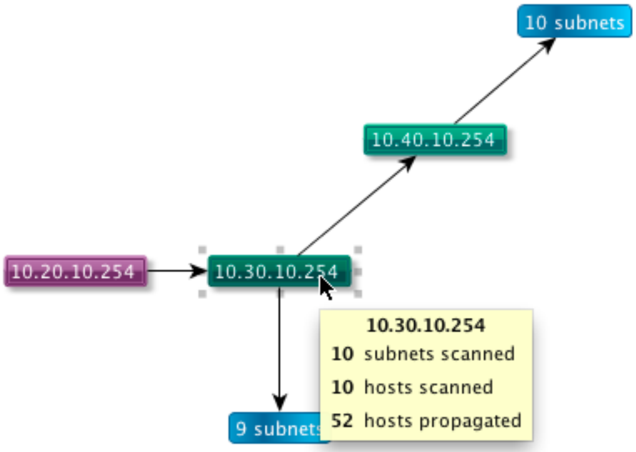
Earliest Time	Port	Protocol	Next Hop
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.10.254
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.30.
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.50.
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.60.
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.70.
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.80.
Feb 1, 2012 6:37:06 AM	135	tcp	10.100.100.
Feb 1, 2012 7:37:03 AM	135	tcp	10.100.20.
Feb 1, 2012 8:37:02 AM	135	tcp	10.100.90.
Feb 1, 2012 2:36:49 PM	135	tcp	10.100.40.

To summarize: The graphical map and underlying table show a single IP address, 10.90.10.254, that is scanning a number of subnets as evidenced by the Addr_scan/tcp alerts. Some of the targets of the scanning activity, for example, 10.100.10.254, are also generating flows with the same characteristics as the original source IP address and are now exhibiting the exact same behavior—scanning other IP address ranges. Finally, we were able to learn this from just two components, the graphical map, and the details table. At this point, we can be fairly certain we have internally spreading malware.

Identify the User ID of the Infected Host

Now that we know we have internally spreading malware, we want to find the devices that are infected and the user IDs logged into those devices. Because StealthWatch® is integrated with the Cisco Identity Services Engine, we can easily find the user identity.

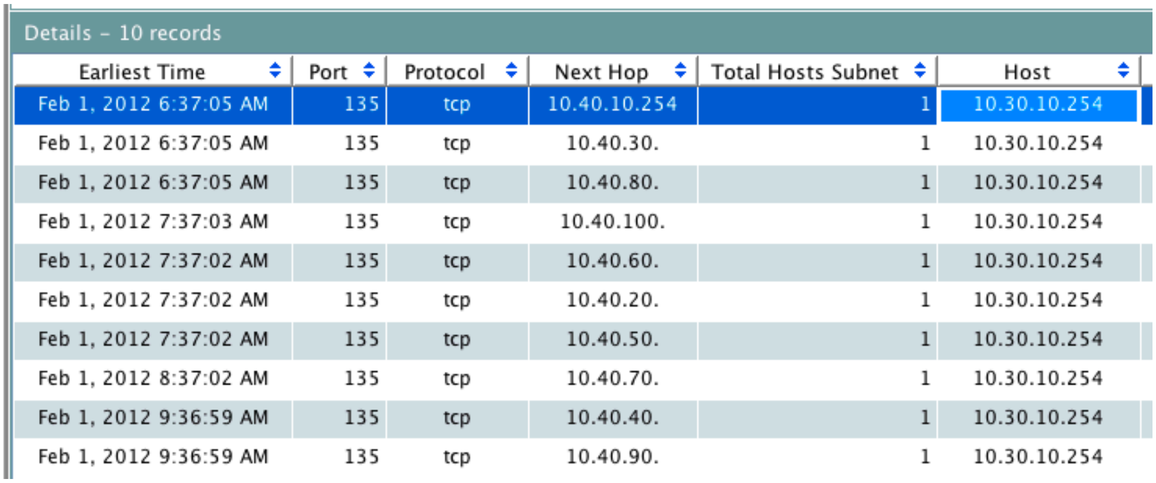
Step 1 Click an infected host IP address on the *Worm Tracker* screen.



Step 2 Select a row from the details table. In this example, we select the first row.

Step 3 Click on the IP address of this host in the *Host* column of the details table.

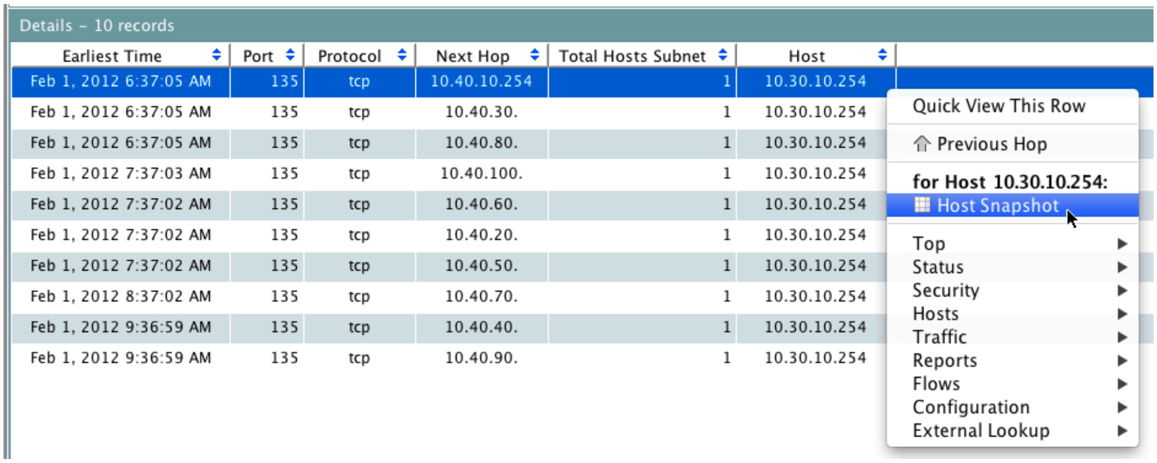
Tip: Each element (cell) in the table is *context-sensitive*. You must be specific about which element in the row you click on, because right clicking on different elements in the row produces different results.



Details – 10 records

Earliest Time	Port	Protocol	Next Hop	Total Hosts Subnet	Host
Feb 1, 2012 6:37:05 AM	135	tcp	10.40.10.254	1	10.30.10.254
Feb 1, 2012 6:37:05 AM	135	tcp	10.40.30.	1	10.30.10.254
Feb 1, 2012 6:37:05 AM	135	tcp	10.40.80.	1	10.30.10.254
Feb 1, 2012 7:37:03 AM	135	tcp	10.40.100.	1	10.30.10.254
Feb 1, 2012 7:37:02 AM	135	tcp	10.40.60.	1	10.30.10.254
Feb 1, 2012 7:37:02 AM	135	tcp	10.40.20.	1	10.30.10.254
Feb 1, 2012 7:37:02 AM	135	tcp	10.40.50.	1	10.30.10.254
Feb 1, 2012 8:37:02 AM	135	tcp	10.40.70.	1	10.30.10.254
Feb 1, 2012 9:36:59 AM	135	tcp	10.40.40.	1	10.30.10.254
Feb 1, 2012 9:36:59 AM	135	tcp	10.40.90.	1	10.30.10.254

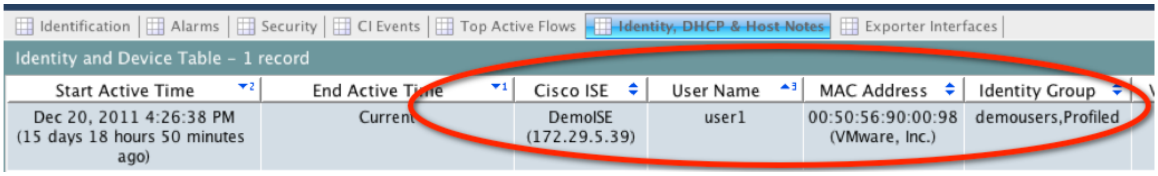
Step 4 Right-click on the selected IP address to produce the context menu for the IP address.



Details – 10 records

Earliest Time	Port	Protocol	Next Hop	Total Hosts Subnet	Host
Feb 1, 2012 6:37:05 AM	135	tcp	10.40.10.254	1	10.30.10.254
Feb 1, 2012 6:37:05 AM	135	tcp	10.40.30.	1	10.30.10.254
Feb 1, 2012 6:37:05 AM	135	tcp	10.40.80.	1	10.30.10.254
Feb 1, 2012 7:37:03 AM	135	tcp	10.40.100.	1	10.30.10.254
Feb 1, 2012 7:37:02 AM	135	tcp	10.40.60.	1	10.30.10.254
Feb 1, 2012 7:37:02 AM	135	tcp	10.40.20.	1	10.30.10.254
Feb 1, 2012 7:37:02 AM	135	tcp	10.40.50.	1	10.30.10.254
Feb 1, 2012 8:37:02 AM	135	tcp	10.40.70.	1	10.30.10.254
Feb 1, 2012 9:36:59 AM	135	tcp	10.40.40.	1	10.30.10.254
Feb 1, 2012 9:36:59 AM	135	tcp	10.40.90.	1	10.30.10.254

This will produce the *Identity and Device Table*, showing the Cisco Identity Services Engine device that captured the user's identity, the user name, and other associated information.



Identity and Device Table - 1 record

Start Active Time	End Active Time	Cisco ISE	User Name	MAC Address	Identity Group
Dec 20, 2011 4:26:38 PM (15 days 18 hours 50 minutes ago)	Current	DemoISE (172.29.5.39)	user1	00:50:56:90:00:98 (VMware, Inc.)	demousers,Profiled

Conclusion

Internally spreading malware is a significant problem facing most companies today. Deploying traditional security products like firewalls and IPSs pervasively on the internal network is costly and difficult to manage. The Cisco Cyber Threat Defense Solution 1.0 provides the visibility mechanisms needed for detecting internally spreading malware and is a scalable, high-performance, cost-effective way to monitor internal network traffic and identify malware that has penetrated the network.