Detecting BotNet Traffic with the Cisco Cyber Threat Defense Solution 1.0

April 9, 2012

Introduction

Since 2009, Botnets have been growing in sophistication and reach to the point where they are now responsible for a significant amount of the data loss and distributed denial of service attacks (DDoS) that businesses encounter each year.

- In late 2010, a software tool called Low Orbit Ion Cannon (LOIC) was used to control an army of thousands of botnets in a well-publicized attack on the recorded media industry dubbed "Operation Payback." Several websites were taken offline for hours to days as a result of the attack.
- Beginning in November of 2009, the "Night Dragon" attack targeted the intellectual property of oil and gas companies and was successful in part by using simple botnet techniques to plant Trojan horse software on computers.
- In March 2011, a massive botnet was used in a DDoS attack against the government of South Korea, taking key government websites offline for hours.

Botnet-controlled computers are a particularly high risk for most enterprises because they can be controlled remotely from anywhere in the world. An attacker could control a botnet-infected host inside a network for any number of purposes, including network reconnaissance, data exfiltration, or denial of service. Unfortunately, locating botnet hosts in a network can be difficult—these hosts often hide their communication to their controllers using standard protocols and ports like HTTP (port 80), HTTPS (port 443), or Internet Relay Chat servers.

The Cisco® Cyber Threat Defense Solution 1.0 addresses botnet problems by providing the visibility mechanisms needed to detect infected hosts within a network. The solution integrates Lancope® StealthWatch® with Cisco's hardware-supported NetFlow and Identity Services Engine, providing a simple and convenient solution to detect and manage this kind of 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.

The Cisco Cyber Threat Defense Solution 1.0 detects botnet-infected hosts using two direct and one indirect mechanisms:

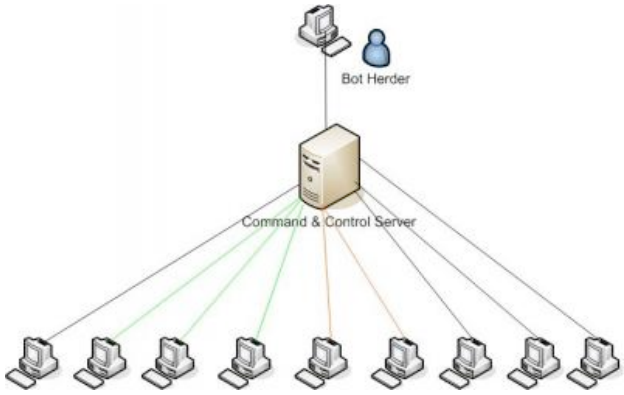
- Blacklists of known botnet command and control servers
- Beaconing host detection
- Flow relationships between inside hosts and outside hosts that have been determined to be botnet controllers, once an inside host has been detected as botnet-infected (a process called *backtracking*).

In this document, we explore all three of these methods.

Operating Concepts

What Is a Botnet?

A botnet is composed of two components: one or more infected "bots" and one or more controllers (but often just a single controller). A bot is a host that has been infected, usually with a rootkit—software that gives a remote attacker full administrative control over the host. A command and control server is software that collects information about the bots under its control; it can be used to issue commands to one, some, or all of the bots simultaneously. The individual in control of the botnet is often referred to as a bot herder. A bot herder could command all of the bots under their control to perform a set of commands by issuing just one command to the control server. Typically, this is how a large botnet is used to perform a distributed denial of service (DDoS). The diagram below shows a simplified view of a botnet.



Note: This is a simplified diagram used for illustration purposes only. Real-world botnets are likely to include hundreds to hundreds of thousands of computers and may employ multiple levels of command and control servers that directly or indirectly communicate with the infected hosts.

For a bot to be controllable by a command and control server, it must periodically communicate with the controller to let it know it is online and able to accept commands. This periodic communication process is nearly always unidirectional—that is, the bot communicates with the controller, but the controller does not respond. Since many of these infected hosts are laptops, they are mobile, and will communicate every time they connect to a network, wherever that may be.

The periodic communication process, while seemingly simple, does something very important—it creates a valid connection through the corporate firewall to the controller that allows bidirectional communication with the infected bot. Most of the time, the command and control server uses this connection to update its

database of infected bots and never responds. However, if the bot herder wants to use the bots for some kind of action, they will issue commands to the command and control server. Then, the next time a bot checks in, it will receive commands that tell it what to do. This method is so efficient that a single command to a control server can be replicated to *hundreds of thousands* of bots, essentially all at once.

Methods of Botnet Detection

In this section, we explore the three ways the Cisco Cyber Threat Defense Solution 1.0 is able to detect botnet-infected hosts.

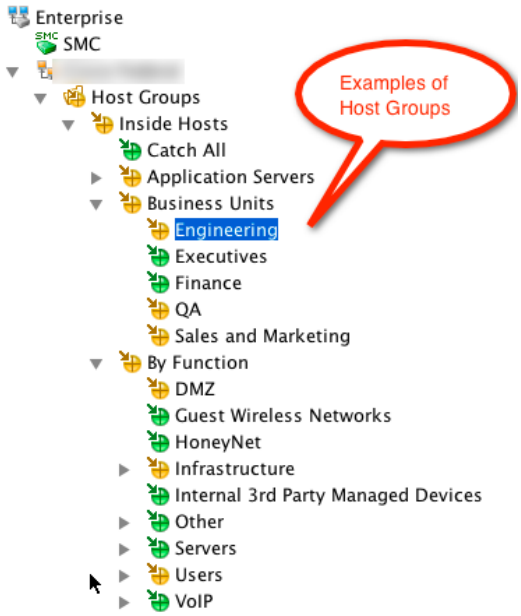
Blacklists - Lists of known botnet controller IP addresses are loaded into StealthWatch®. These IP addresses are publicly available from a variety of sources on the Internet, such as https://zeustracker.abuse.ch/. When packets traverse a NetFlow-supported device, StealthWatch® detects any inside host communicating to any IP address on the botnet list and immediately triggers an alarm. Botnet lists can be loaded either manually or using an automation script run by an external service, such as a cron job. Note: This document only discusses manually loading IP addresses; however, it is possible to automate this process using scripts.

StealthWatch® beaconing hosts - A beaconing host is defined as IP communication between an inside and outside host (with traffic in only one direction) that exceeds the number of seconds required for a flow to qualify as long duration. An alert is generated if StealthWatch® detects suspicious channels of communication that might be spyware, remote desktop technologies (e.g., gotomypc.com), botnets, or other covert means of communication. As mentioned earlier, the periodic communication process of an infected host is usually unidirectional, and thus looks exactly like the type of communication tracked by the beaconing host rule.

Backtracking - If an outside host is discovered to be a botnet command and control server, StealthWatch® can historically backtrack any communications from or to that outside host from inside hosts. Any inside host that can be shown to have communicated with that outside host should then be treated as potentially infected. This is usually the last step performed when an inside host has been shown to have contacted a botnet command and control server.

The Role of Host Groups in Botnet Detection

StealthWatch® makes use of *host groups* for botnet detection. In StealthWatch®, a host group is simply a collection of devices defined by IP address. Host groups can be described many different ways to conform to an enterprise's unique requirements, and might have descriptions such as "Inside Hosts," "Application Servers," "Server Farm," "Data Center," "Executives," "VPN IP Address Pool," "Engineering Clients," or "IP Phones."


Enterprise
SMC
Host Groups
Inside Hosts
Catch All
Application Servers
Business Units
Engineering
Executives
Finance
QA
Sales and Marketing
By Function
DMZ
Guest Wireless Networks
HoneyNet
Infrastructure
Internal 3rd Party Managed Devices
Other
Servers
Users
VoIP
Examples of
Host Groups

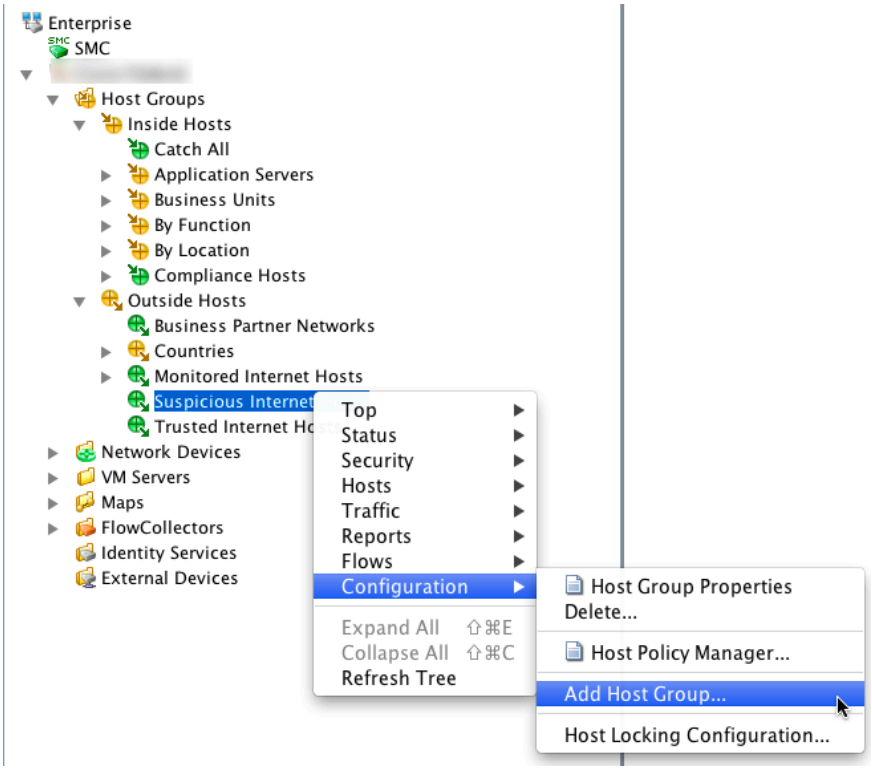
Botnet Detection Example

Botnet detection begins by creating a new outside host group underneath the *Suspicious Internet Hosts* group. The group we create will be composed of a list of IP addresses of known botnet command and control servers.

Step 1 Select *Suspicious Internet Hosts.*

Step 2 Right-click to bring up the sub-menu.

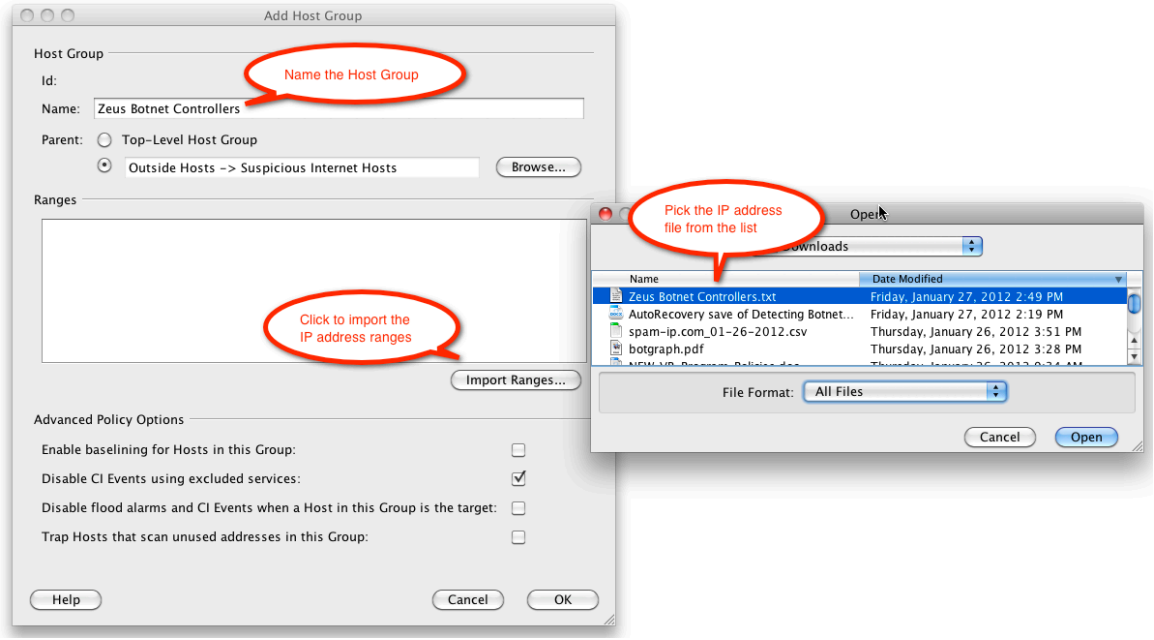
Step 3 Select Configuration ➔ Add Host Group.



This results in a dialog box used to describe the new host group.

Step 4 Complete the name field.

Step 5 Copy and paste the IP addresses or ranges within the ranges section, or click *Import Ranges* button to import the IP addresses.

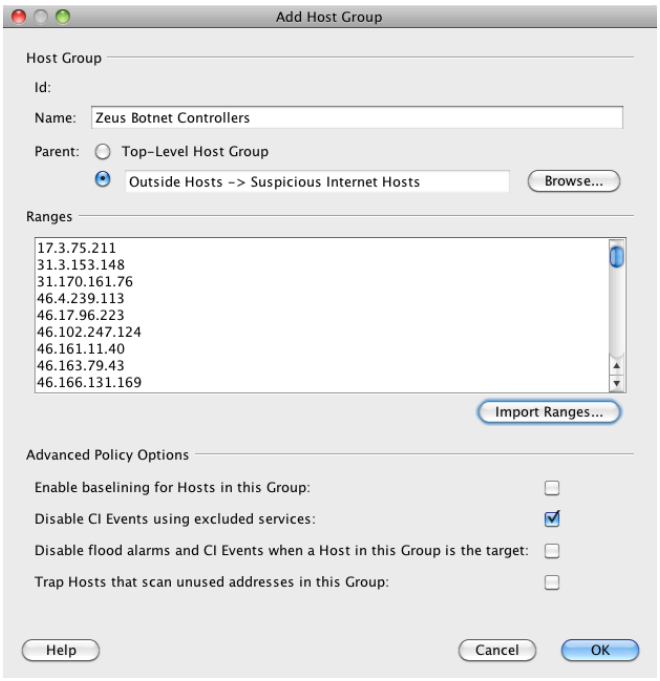


This will bring up another dialog box you can use to navigate your file system to the file that contains the IP address blacklist.

Note: It is up to the security administrator to locate and download any blacklists they want to use. In this example, we have previously downloaded a botnet list from https://zeustracker.abuse.ch/. There are a number of blacklist websites to choose from, and multiple lists can be imported as different host group names. Which address list(s) you choose to import will depend on what kind of hosts you want to protect against.

Step 6 Click *Open* to import the list.

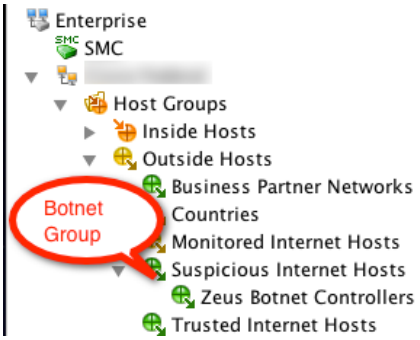
When the list has been imported, it should look similar to the list below:



Note: The IP addresses in the dialog box above are for illustration purposes only. Cisco makes no representation as to whether these addresses represent actual botnet controllers.

Step 7 Click *OK.*

StealthWatch® will commit these changes to the new host group, in this case called "Zeus Botnet Controllers."



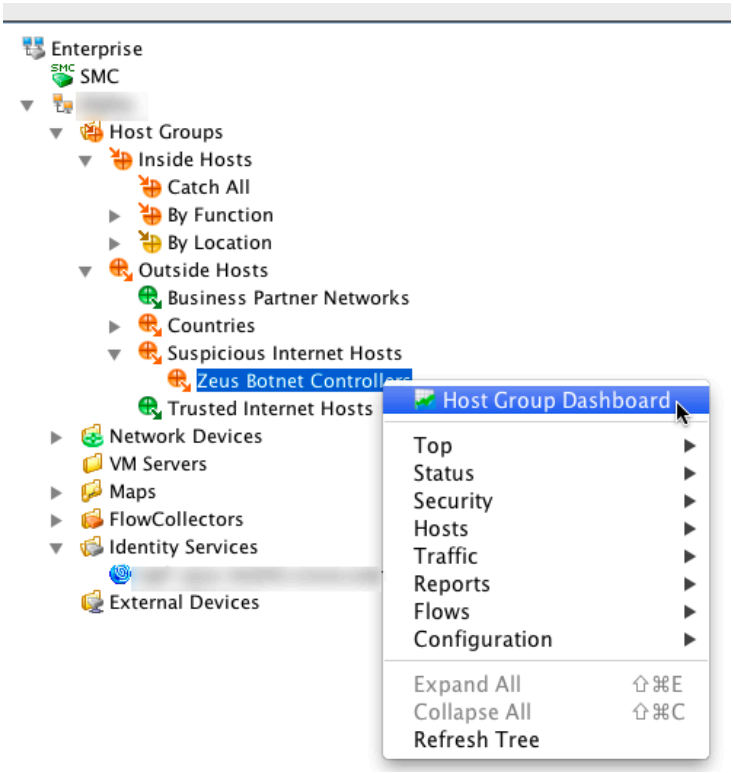
Note: There is a default "Host Locking" rule pre-defined to trigger an alarm if there is communication from Inside hosts to any "Suspicious Internet Hosts" or sub-group of Suspicious Internet Hosts. The rules can be modified or reviewed under Configuration → Host Locking Configuration.

Beaconing Hosts

By default, StealthWatch® rules trigger whenever it detects a beaconing host. To determine if you have any beaconing hosts, look for any internal host group that is orange in the navigation tree.

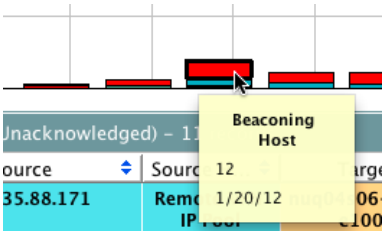
Step 1 Select a host group that is orange.

Step 2 Right-click on the group and select *Host Group Dashboard.*



This displays a *Host Group Dashboard* that includes a graph of all of the alarms and a table of the alarms.

Step 3 Hover your mouse over each part of the bars to see what they represent.

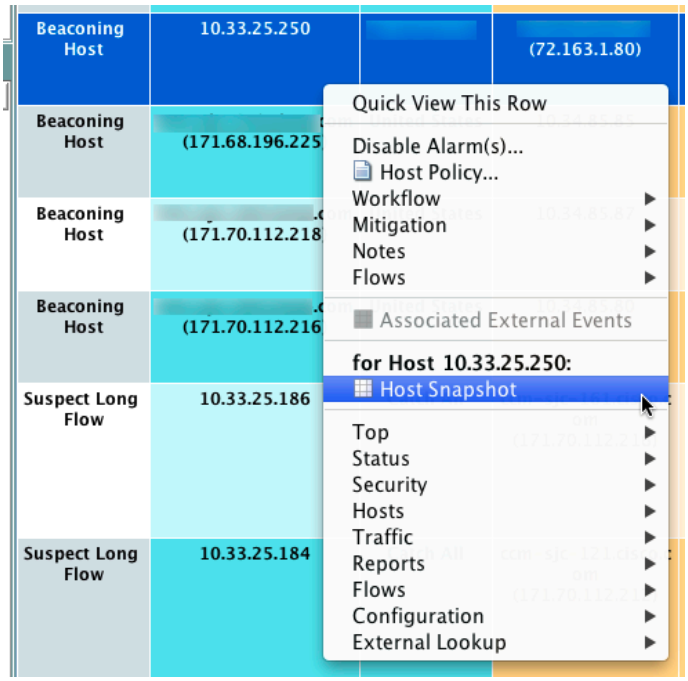


Step 4 Search the alarm table below for a *Beaconing Host.*

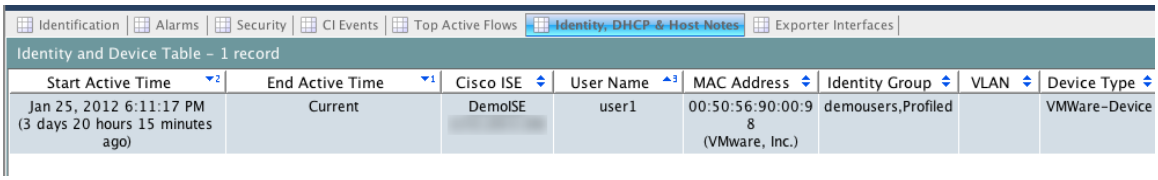
Step 5 Select the column with the source IP address.

Step 6 Right-click the source IP address.

Step 7 Select *Host Snapshot* from the popup menu.

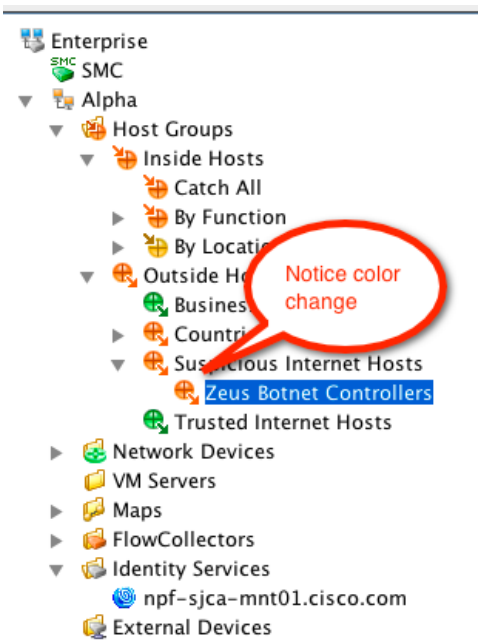


Step 8 Select the *Identity, DHCP, and Host Notes* tab from the Host Snapshot View to view the user identity and posture information from the Cisco Identity Services Engine.



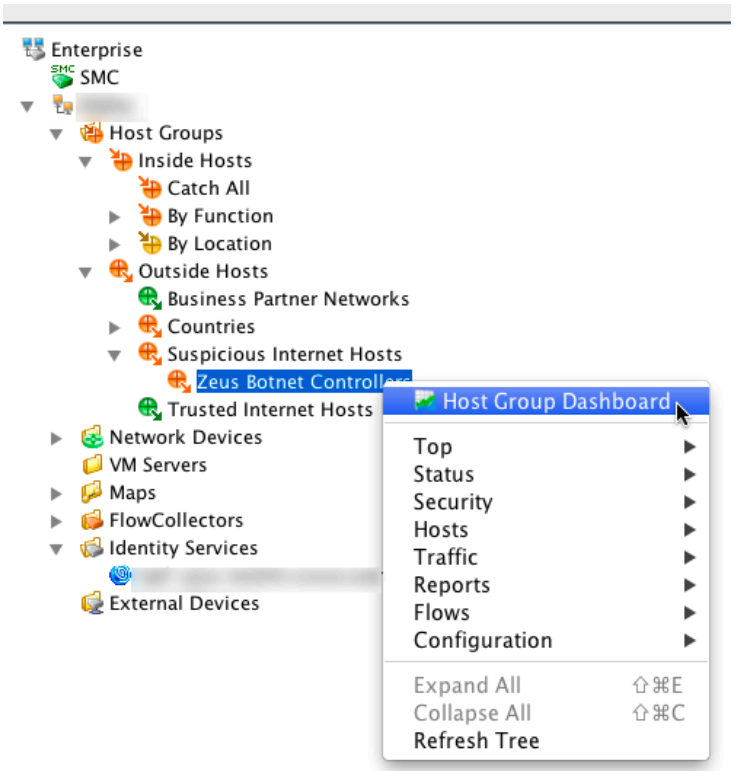
Working with Live Botnet-Infected Hosts

When a live botnet-infected host is detected, the color of the host group in the navigation tree will change from green or yellow to orange, indicating that a host from the inside host group has sent packets to a host in the list of botnet controllers you created.



Step 1 Select the host group with an orange icon.

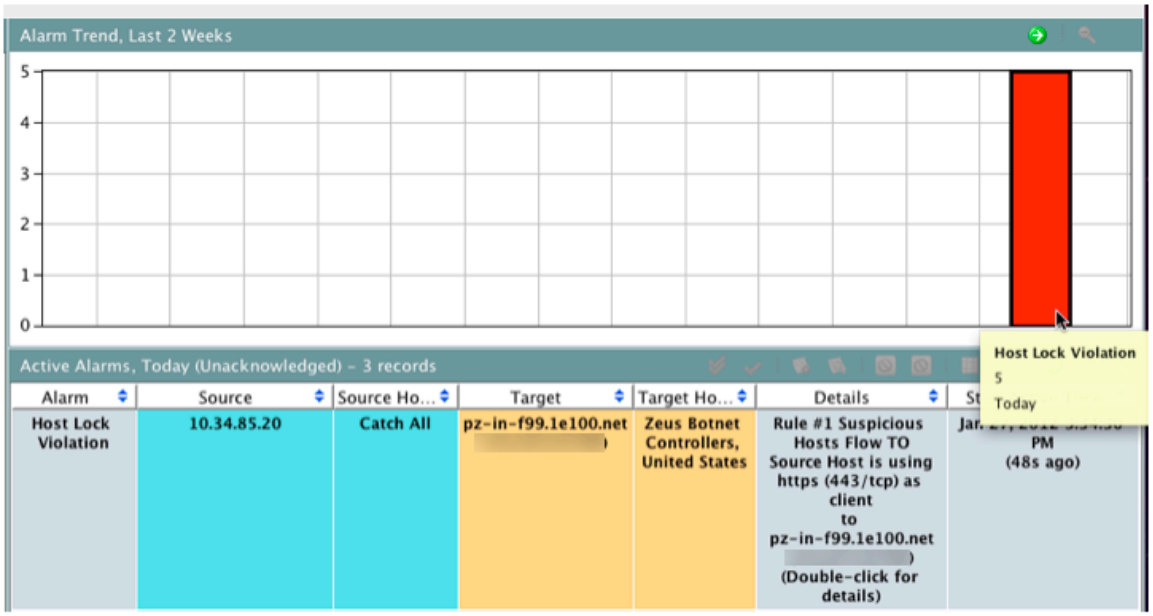
Step 2 Right-click on the host group and select the *Host Group Dashboard.*



This will produce a *Host Group Dashboard* that includes a graph of alarms for this host group.

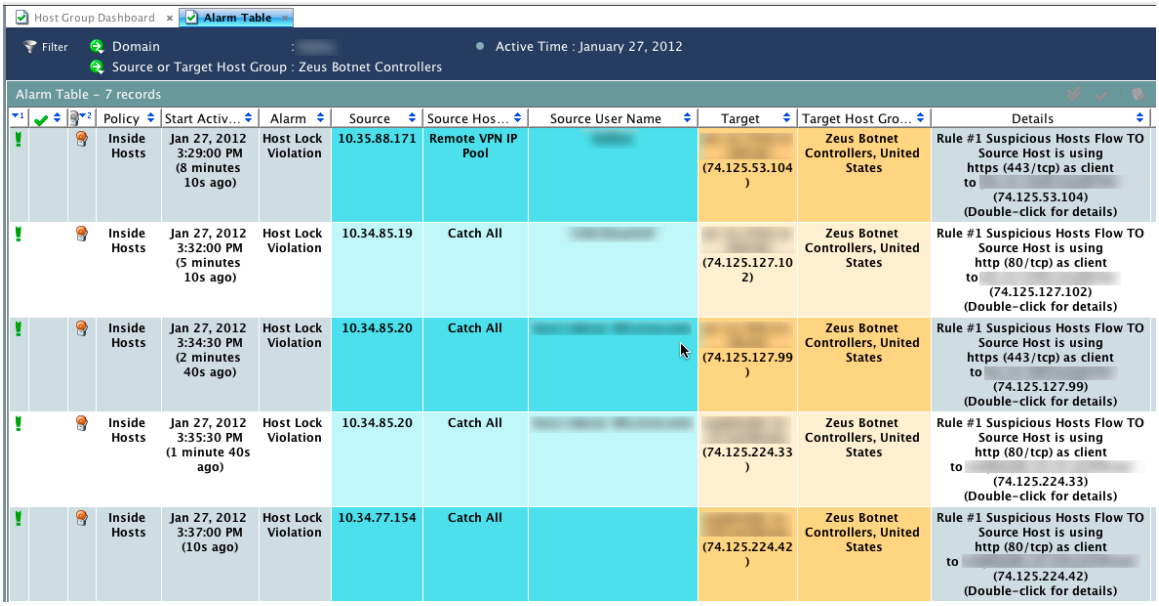
Step 3 Use your mouse to hover over a bar in the group to see the alarm condition.

Whenever an inside host triggers an alarm to a *Suspicious Internet Hosts* group, it generates a *Host Lock Violation* alarm, as is the case below.



Step 4 Double-click on any bar in the graph to look at the data in the underlying alarm table, as shown below.

As you can see from the *Target Host Group* column, the reason this rule fired is because the target of the communication was an IP address in the *Zeus Botnet Controllers* host group.



Policy	Start Activ...	Alarm	Source	Source Hos...	Source User Name	Target	Target Host Gro...	Details
Inside Hosts	Jan 27, 2012 3:29:00 PM (8 minutes 10s ago)	Host Lock Violation	10.35.88.171	Remote VPN IP Pool		(74.125.53.104)	Zeus Botnet Controllers, United States	Rule #1 Suspicious Hosts Flow TO Source Host is using https (443/tcp) as client to (74.125.53.104) (Double-click for details)
Inside Hosts	Jan 27, 2012 3:32:00 PM (5 minutes 10s ago)	Host Lock Violation	10.34.85.19	Catch All		(74.125.127.102)	Zeus Botnet Controllers, United States	Rule #1 Suspicious Hosts Flow TO Source Host is using http (80/tcp) as client to (74.125.127.102) (Double-click for details)
Inside Hosts	Jan 27, 2012 3:34:30 PM (2 minutes 40s ago)	Host Lock Violation	10.34.85.20	Catch All		(74.125.127.99)	Zeus Botnet Controllers, United States	Rule #1 Suspicious Hosts Flow TO Source Host is using https (443/tcp) as client to (74.125.127.99) (Double-click for details)
Inside Hosts	Jan 27, 2012 3:35:30 PM (1 minute 40s ago)	Host Lock Violation	10.34.85.20	Catch All		(74.125.224.33)	Zeus Botnet Controllers, United States	Rule #1 Suspicious Hosts Flow TO Source Host is using http (80/tcp) as client to (74.125.224.33) (Double-click for details)
Inside Hosts	Jan 27, 2012 3:37:00 PM (10s ago)	Host Lock Violation	10.34.77.154	Catch All		(74.125.224.42)	Zeus Botnet Controllers, United States	Rule #1 Suspicious Hosts Flow TO Source Host is using http (80/tcp) as client to (74.125.224.42) (Double-click for details)

You will also notice the *Source IP* address and *Source User Name* (although blurred) of the flow, as provided by the Cisco Identity Services Engine. With this information, it is now a straightforward matter to locate the infected device.

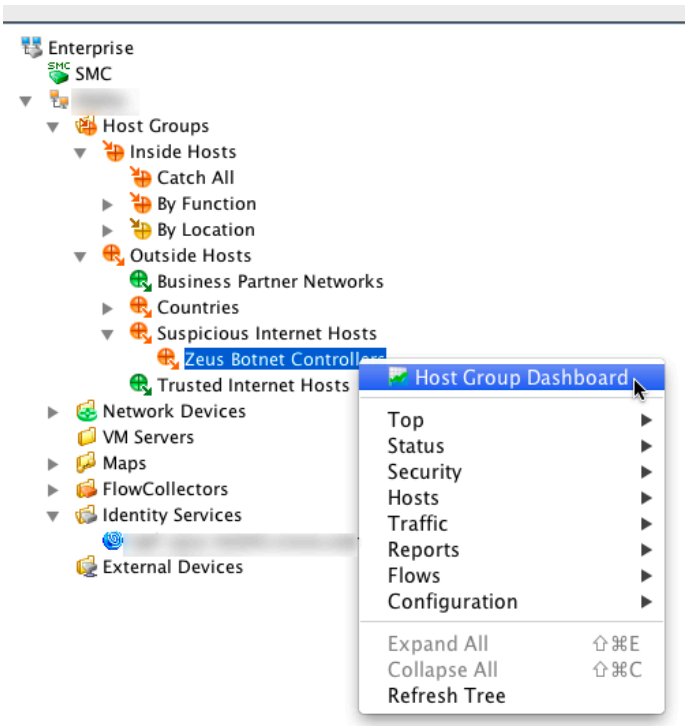
Backtracking to Locate Other Botnet-Infected Hosts

Once an outside host has been determined to be a botnet controller, StealthWatch® makes it easy to determine if other internal hosts may also be infected.

Step 1 Select the host group where the botnet controller IP address appeared.

Step 2 Right-click on the host group.

Step 3 Select *Host Group Dashboard* from the list.

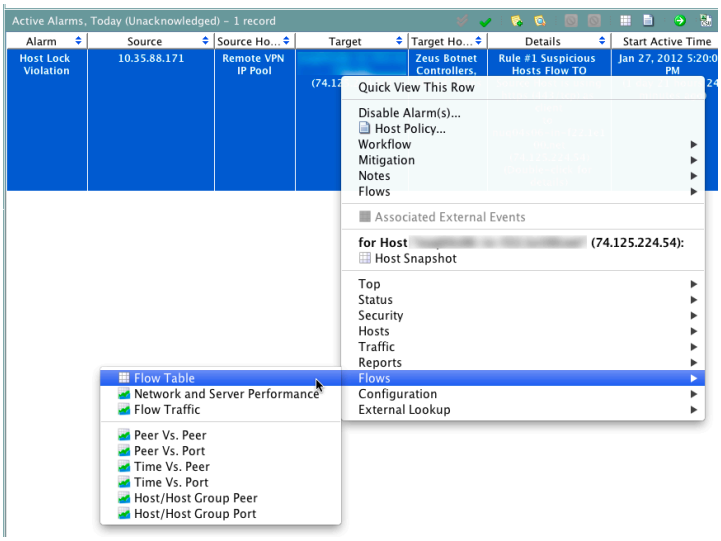


Step 4 Select the target IP address from a *Host Lock Violation* alarm record.

Step 5 Click on the IP address in the *Target* column.

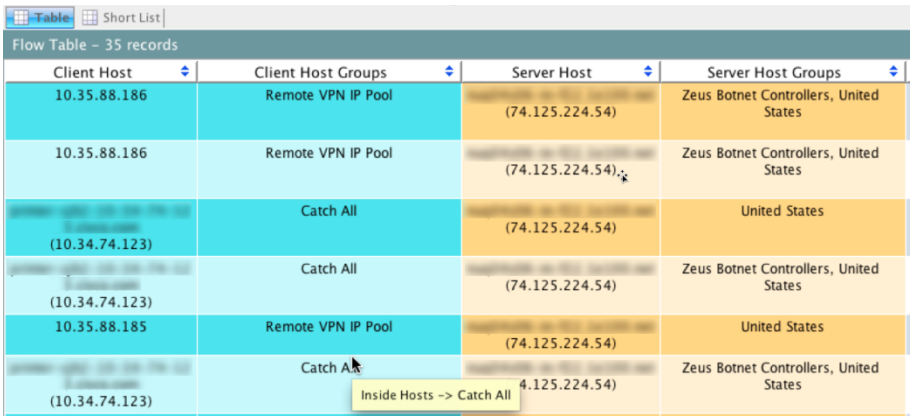
Step 6 Right-click the IP address.

Step 7 Select Flows ➔ Flow Table from the list.



The resulting flow table will show every client IP address in the filter range that has communicated with the target IP—in this case, the botnet controller. Notice that every S*erver Host* IP address is the IP address of the botnet controller, but that there are multiple *Client Hosts* IP addresses. Each client host should be investigated simply by virtue of having communicated with the botnet controller.

By using this simple backtracking technique, it is easy to discover any other IP addresses on the inside network that have communicated with the botnet controller.



Conclusion

Botnets are a significant problem facing most companies today. Detecting them using traditional security measures can be difficult and time consuming. The Cisco Cyber Threat Defense Solution 1.0 provides three easy-to-use methods for identifying botnet-infected hosts—botnet blacklists, beaconing host detection, and IP backtracking.