

Identifying Botnet Traffic on an Enterprise Network

Mary Henthorn
University of Arkansas at Little Rock

Abstract-The efficiency and resilience of botnets is evidenced by their continued effective distribution of spam, phishing attempts, denial of service attacks, and fear that the most sensitive data on our systems can be collected and delivered to criminals. Determined botnet hunters and sophisticated network appliances give us the ammunition to fight back. This study describes the comparison of data available about the behavior of botnets to the current approach to botnet identification on a large enterprise network. Querying the network monitoring system for evidence of known botnets that use less common means of command and control communication revealed that such malicious networks are present and active in the environment. Armed with this knowledge security analysts can now recognize additional malware, expand their sources of current information, and continue to advance their means of identification, containment, and suppression of network attackers in anticipation of the more stealthy botnets of the future.

I. INTRODUCTION

The Internet has been a marvelous tool for education, business, and individuals because of its ubiquitous nature and wealth of information. The release of the Morris worm [1] in 1988 demonstrated that the Internet erases the effects of distance and time for harmful activity as well as for innocent users. Recent news describes the arrests of people engaged in criminal money-making activities in two separate cases involving the use of malicious networks of hundreds of thousands of computers [2]. The profit motive promises to spur more misuse of the Internet than curiosity and the thrill of hacking have in the past.

The battle to maintain the confidentiality, integrity, and availability of legitimate networks is challenged by unlawful people armed with open source, well designed malware. The Agobot system, for example, contains about 20,000 lines of C/C++ code with a modular structure that facilitates customization [3]. A user-friendly interface

makes it relatively simple for someone with limited technical experience to use. Since the Internet has countless systems vulnerable to attack, it is easy for a hacker to infect and control virtual armies of computers.

Network security analysts must take advantage of all resources available to protect their assets. Antivirus, anti-spyware, and firewalls alone will not stop the spread of malware on the Internet. Because the attacking software is so easy to customize, defense systems must be continually tuned to recognize and respond to new types of intrusions. Information gathered by those who study the characteristics of malicious code must be quickly incorporated into the network tools that recognize the appearance of potentially harmful intrusions.

This study takes a collection of information available about the behavior of botnets and compares it to the identification methods implemented with an enterprise network monitoring and threat mitigation system on a large network. It is intended that patterns of malicious Internet traffic identified in this study will augment the existing network defense system.

II. BACKGROUND

Botnets are networks of compromised computers or zombies that can be directed to perform tasks according to instructions received from an attacker. Most are some variation of a three-layer configuration including an attacker, command and control (C&C) nodes, and zombies. Some botnets use IRC channels for communication (Figure 1). Other botnets make their communication traffic more difficult to detect by using plain text or encrypted HTTP or peer-to-peer (P2P) protocols [4,5] (Figure 2).

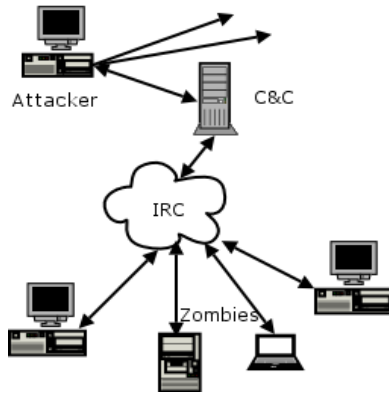


Figure 1. IRC Botnets

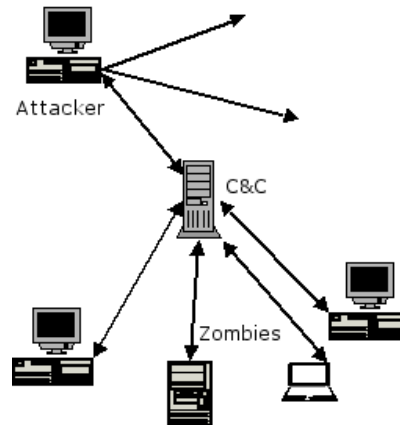


Figure 2. Botnet controlled without IRC

The spreading mechanisms used by botnets are similar to other worm propagation methods. Most scan networks to locate and exploit some type of software vulnerability. A few well known vulnerabilities account for most infections: Microsoft-DS Service, NetBIOS Session Service, NetBIOS Name Service, and Remote Procedure Call services. The network traffic generated through these propagation methods can be observed on ports 135, 137, 138 and 445 [6]. Botnets can also make use of any number of less well known vulnerabilities to gain entry into a system. Patching Microsoft vulnerabilities alone will not stop the spread of botnets.

Once a computer has been compromised with a botnet infection, a communication channel is established with a C&C node. At that point the zombie may download instructions and tools. Sometimes the zombie can even hide the evidence that the machine has been infected.

Most commonly, zombies receive instructions and software tools to deliver spam, retrieve and report passwords, perform key logging, collect software keys, or conduct click fraud. The least destructive activities use network bandwidth and the CPU time of the infected zombie. Of more concern, a botnet can stealthily capture highly sensitive data or shut down a system with a denial of service attack.

Simulations have demonstrated that botnets can be extremely efficient and resilient [7]. According to this study, a system with a million nodes could deliver a large exploit to every zombie in a matter of minutes. The same study demonstrated that botnets could be exceptionally resilient. Random removal of 50% of a botnet's nodes had a minimal effect on the connectivity of the remainder of the nodes.

Recently the motivation for Internet abusers has shifted from curiosity to financial gain [8]. Botnets are particularly attractive to anyone who can profit from delivering spam, phishing attacks, or even extortion. Since botnets use infrastructure that belongs to others, botnet masters enjoy

financially asymmetric systems where a very low investment can produce a large profit.

Botnets can be instructed to install adware, collecting fixed fees for each installation for the botnet master. Some commit click fraud, again collecting fees for each click delivered to sites that pay for each hit to the advertiser's website. A botnet's capabilities can be so profitable that they are rented out to "customers" who pay for its services. Some botnet propagation methods even look for established botnets and steal them.

To counter the threat of botnets, individuals and organizations collect data about their behavior and develop tools for their detection and removal. Commercial antivirus companies track worms that infect computers and expose them to botnet systems that can control them as zombies. The code used for many botnets is publicly available. While the sophisticated modular design of Agobot, for example, allows for countless variations, still the variables and commands are readily available to those trying to track botnets. The Honeynet Project and Research Alliance [9] encourages the study of botnets through malware capture and analysis. One of their members, the German Honeynet Project, has developed a program to capture malware by simulating vulnerabilities.

III. PROBLEM

Because of the resiliency and efficiency of botnets, the approaches designed to defend against attacks must go beyond randomly dropping nodes [7]. The multi-faceted threat posed by botnets requires a comprehensive defense including prevention, monitoring, and reaction. Preventive measures reduce the spread of malnets by removing vulnerabilities. Preventive measures alone are not adequate to stop all appearances of malnets in a network of any size. Vulnerabilities may be exploited before they can be patched and even educated users may fall victim to social engineering. Reactive methods are available to stop denial of

service attacks that produce fast bursts of attack traffic [10]. In response to this protection, developers of malnets have learned to slow their traffic.

Botnet activity can be monitored on individual devices, local area networks, and enterprise networks. If antivirus and anti-spyware are kept up-to-date, they should detect the presence of known malware including botnet worms and Trojans. Suspicious traffic on a local area network can be dropped or examined. Packet analysis can even identify botnet commands if the traffic is not encrypted. Tools at the gateways of enterprise networks can recognize behaviors such as scanning or repeated login attempts and relate the activities to known botnet IP addresses or ports.

Freiling and others describe disruption of malnets by using an automated approach to analyzing IRC-based malnets [11]. Although their proposed methods of shutting down identified command and control nodes are manual, in practice similar methods have been automated and demonstrated to be effective against IRC-based malnets. Identified patterns in the traffic of IRC-based malnets can facilitate the use of systems such as Cisco's Monitoring, Analysis and Response System (CS-MARS) to automatically prevent communication with C&C nodes and to notify administrators responsible for decontamination of infected devices.

IV.METHOD

The network to be observed is large, encompassing about 2,000 routers and 500,000 users. The infrastructure up to the point of the routers is managed as a single entity. Even with the control of IRC-based communication, botnets continue to be an issue for enterprise network administrators. Security analysts have used the CS-MARS threat mitigation system to identify and respond to harmful activity on the network for over a year. Because of a history of disruptive botnet activity on this network, these analysts have been particularly mindful of the need to disrupt IRC C&C nodes and to identify infected machines so they can promptly be removed from the network and cleaned.

Although IRC botnets have been reported by one analyst as "disturbingly quiet", undesirable activity characteristic of zombies can still be observed. In a preliminary examination of traffic with the CS-MARS tool, dozens of network computers could be seen engaging in sessions with a known click fraud site. Disruption of IRC-based botnets did not eradicate all botnet-like behavior.

It is known that botnets communicate through protocols other than IRC. However, it can be difficult, even in a small network to identify HTTP traffic that carries botnet commands. P2P activity on known ports can be monitored, but botnets are not likely to use well known P2P ports.

Botnet hunters have gathered data about the behavior of these malicious constructs in environments more conducive to examining the full content of malicious data streams than a large enterprise network. Rather than attempting to duplicate their work, the methodology used in this study leverages what they have learned together with the vast amount of traffic available for examination on the enterprise network.

Through this study, data is gathered by direct observation of a large network in an attempt to identify patterns in the communication networks of the botnets that exist after IRC-based malnets have been eliminated. Hopefully these patterns of malicious communication will be found suitable to adopt as CS-MARS rules effectively automating tasks of botnet identification, containment and suppression.

This method of identifying patterns in the communication networks of the botnets that exist after IRC-based botnets have been controlled includes:

- Learning the capabilities of the CS-MARS system
- Studying the current implementation of CS-MARS
- Categorizing botnet behavior from related research
- Matching behaviors from research with CS-MARS implementation
- Reviewing collected data to identify potential new patterns
- Testing of a selection of these potential patterns as CS-MARS queries
- Analyzing the results of the queries

V.CS-MARS

The Cisco Security Monitoring, Analysis, and Response System Appliance is a Security Threat Mitigation (STM) tool appropriate for use on a large enterprise network. For the purpose of this study, an account was activated for observation of incidents and reports generated by the system and the running of simple queries. This global level access provides a view of data aggregated from several local controllers.

A. CS-MARS Capabilities

With global access to the system, network-wide traffic can be observed through a browser interface. The Dashboard view offers a quick look at recent incidents. Reports can be accessed as HTML or CSV files can be downloaded for import into Excel or a database system for further analysis.

Rules in the CS-MARS system identify undesirable patterns of network traffic. Source and destination IP addresses and ports can be specified. Combinations of ports and protocols can be identified as services. Events are pre-defined activities. These elements can be sequenced with "and", "or", and "follows" conditions. Observed network activities that violate these rules can trigger notifications or firewall actions.

B. CS-MARS Implementation

Sixty-four unique rules were available on the CS-MARS implementation used for this study. As displayed in Figure 3, the 64 rules in place on this CS-MARS implementation can be roughly grouped into five categories: infection spreading activity; attack activity; network maintenance; botnet communication; and, other.

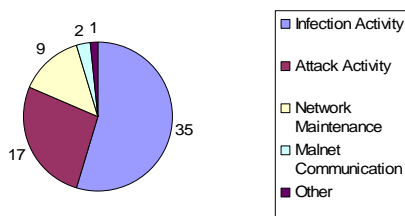


Figure 3. CS-MARS Rules

CS-MARS presents incidents as collections of data observed when a rule is violated. Through the time of this observation about 1,400 incidents occurred each week. Most incidents were the result of the violation of a few rules. Incidents for one typical week primarily resulted from mass mailing worms and IRC-botnet activity. (Figure 4)

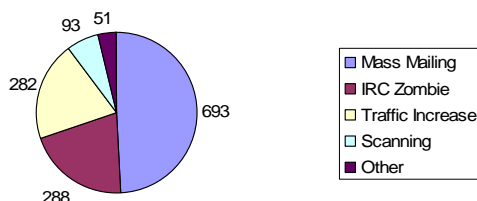


Figure 4. Incidents 3/27/2006 – 4/3/2006

The CS-MARS Dashboard as implemented on this network displays recent incidents in an HTML format suitable for drilling down to more detail and a summary of events and incidents observed over the last 24 hours. The data storage system on this network can hold about six hours worth of data at any one time. Queries can be issued to count matches on source or destination IP addresses, ports, and protocols. Over 200 reports that collect more information were available to run on demand or as scheduled tasks. Because of the amount of data and the fact that this is a production environment, review of regularly scheduled reports was the best way to observe top IP and port destinations and IRC-malnet traffic without placing any additional load on the system.

About 50 reports were downloaded in CSV format. Reports that recorded the 100 most active source or destination IP addresses and ports were collected for various one hour time periods over the several weeks of the study. Several reports of source and destination IPs and ports involving known IRC

C&C nodes were also downloaded for comparison to other botnet research.

VI. RELATED BOTNET RESEARCH

This study is concerned with the observation and control of botnet activity at the enterprise network level. Only shallow packet analysis will be available for analysis. Botnet hunters who monitor local area network traffic and directly examine infected devices assemble detailed information about the behaviors of botnets. Knowledge of these facts is necessary for automating the identification of botnet activity on the enterprise network.

A considerable amount of data is available about botnet behavior. The Honeynet Project [9] includes alliance organizations from around the world. Their white paper, “Know Your Enemy: Tracking Botnets” [6], describes the open source code of botnets, and their methods of spreading, communicating, and delivering attacks. Companies that provide anti-virus software and other security tools, such as Symantec [12] and Lurhq [13], publish information about malware. SANS’ Internet Storm Center (ISC) [14] supports network security investigators by providing Internet traffic statistics. Recently a moderated mailing list was established for information sharing among the community of people observing and trying to contain botnets [15]. One of the early messages sent out on this list was Jose’s list of most frequently observed ports used by botnets.

Little information is available on the Internet about botnets that use communication methods other than IRC. Although HTTP has been used to control network agents for some time, Bobox is one of only a few HTTP botnets found in this investigation. After installation the Bobox Trojan attempts to contact a number of websites. Bobox uses DNS names instead of IP addresses, making it easy to change the C&C host location. Some Bobox variants have the capability of generating DNS name variations to counter attempts to block traffic to the known hosts.

Phatbot has well documented P2P communication capability, although it is secondary to the more efficient IRC methods. It uses a variation of WASTE, an open source P2P tool, but does not make use of the WASTE encryption option.

VII. DATA ORGANIZATION

Some means of organizing the data was needed before effectively associating the massive amount of information available about botnets with the millions of packets moving through the network. 100 items were selected from six collections of data that might indicate the presence of botnets:

- Known botnet behaviors, such as port scanning [3], [4]
- Known ports used by malnets [12], [13], [15], [16]
- DNS names used by Bobox [12]

- Ports noted in CS-MARS reports of top destination ports
- ISC top 10 ports 4/12/2006 [14]
- Jose's top 10 botnet ports 3-3-06 [15]

These 100 items were compared to the CS-MARS rules, noting behaviors, IP addresses, and ports present in the rules. Then the 100 items were compared to CS-MARS reports noting any port numbers that occurred in IRC Command and Control reports. All behaviors and most other indicators were found in the CS-MARS rules and reports, leaving only 23 ports and the list of Bobox DNS names. Each of the Bobox DNS names was searched on DNSWatch [17] to get current IP addresses. These IP addresses were matched to the CS-MARS rules, confirming that none were included. None of these ports or IP addresses would be expected to be used in normal business activity on this network.

VIII. TESTS ON THE NETWORK

Since Bobox and Phatbot botnet systems have the ability to handle C&C communication without IRC, all their indicators not included in CS-MARS reports or rules were chosen for further testing. A few other ports known to recently have unusually high activity were also kept, resulting in the following ten items to test:

1. Bobox servers
2. Phatbot – WASTE P2P, but alternate gnutella port 4387
3. Phatbot WASTE P2P port 1337
4. Phatbot W32.hllw.gaobot.dk port 63809
5. Phatbot 63808
6. #8 Jose's top 10 port 7991
7. #10 Jose's top 10 5555
8. ISC sudden traffic increase 4-9-2006 port 12757
9. ISC sudden traffic increase 4-4-2006 port 27754
10. ISC sudden traffic increase 4-4-2006 38566

Queries were run for each of these tests using the most recent ten minutes worth of CS-MARS network data. The first set of tests run at mid-day on a Saturday resulted in 101 matches. As an indicator of the volume of network traffic at the time of test, a single query of ten minutes of weekend any source to any destination activity counted 788,493 allowed session connections or teardowns. A second set of the same queries was run during normal business hours on a Monday resulting in 157 matches. A ten minute query during the hour of the testing on Monday resulted in 2,691,139 allowed session connections or teardowns.

Table 1. Test results

Test	Source IP	Destination IP	Destination Port	Count 4/15/06	Count 4/17/06
1	204.16.170.100	Any	Any	0	0
1	Any	204.16.170.100	Any	0	0
1	67.15.35.19	Any	Any	0	1
1	Any	67.15.35.19	Any	0	2
1	70.57.227.130	Any	Any	0	0
1	Any	70.57.227.130	Any	0	0
1	68.178.232.99	Any	Any	0	1
1	Any	68.178.232.99	Any	18	23
1	70.84.177.195	Any	Any	0	0
1	Any	70.84.177.195	Any	0	0
1	70.84.177.196	Any	Any	0	0
1	Any	70.84.177.196	Any	0	0
1	70.84.177.197	Any	Any	0	0
1	Any	70.84.177.197	Any	0	0
1	70.84.177.198	Any	Any	13	8
1	Any	70.84.177.198	Any	9	4
1	209.94.121.127	Any	Any	0	0
1	Any	209.94.121.127	Any	0	0
1	204.16.173.40	Any	Any	0	0
1	Any	204.16.173.40	Any	0	0
2	Any	Any	4387	21	28
3	Any	Any	1337	26	43
4	Any	Any	63809	2	4
5	Any	Any	63808	1	5
6	Any	Any	7991	1	0

7	Any	Any	5555	9	12
8	Any	Any	12757	0	5
9	Any	Any	27754	1	21
10	Any	Any	38566	0	0
Total				101	157

A. Observations

No incidents were identified by the CS-MARS system during the time period of the tests on either day, indicating that current rules were not detecting the conditions specified in the queries. No applicable reports were available to compare to the results because none were scheduled to run on Saturday. Reports run during the hour of the Monday tests show no matches to the IPs or ports included in the test queries.

It was particularly interesting that Saturday's queries found the known botnet C&C hosts using IP 68.178.232.99 to be a destination port 18 times, and Monday's queries observed it 23 times. Although it had been assumed that the traffic was on HTTP port 80, the original queries did not specify the destination port. On another business day morning, the IP for this host was again verified to be 68.178.232.99 and a new query was run. At 8:28 a.m. a query for destination IP 68.178.232.99 on any port resulted in 64 hits. Immediately afterwards another query run testing for this destination IP with destination port 80 returned 73 hits.

IX. Conclusions

The queries were selected to expose the presence of botnet traffic on the network that used HTTP or P2P communication, or other botnet communication that was not being detected by rules or reports currently in place on the CS-MARS system. The fact that some of these queries matched sessions running in ten minute samples indicates these queries are potential new patterns for identifying and controlling botnet activity in the network.

The additional test of the probable Bobox host site for port 80 traffic is consistent with the assumption that Bobox traffic would use HTTP rather than IRC as a communication protocol. The observation of a higher volume of traffic to the Bobox host at approximately 8:30 in the morning could mean that when infected zombie computers were started they automatically checked in with the C&C host.

All of the hosts checked for test case number 1 and the ports chosen for the other nine tests were obtained from sources available to network security analysts. These analysts are successfully using CS-MARS to detect, contain, and remove infections of IRC-based botnets. The results presented in this study indicate botnet communication through HTTP and P2P protocols can also be detected with the CS-MARS system.

CS-MARS rules for implementation of the patterns used in these tests could be employed to identify compromised

computers on the network and block C&C nodes. As with other incidents generated by this network's CS-MARS implementation, those responsible for the infected machines can be notified and instructed in how to clean the devices and eliminate vulnerabilities that invite re-infection. Confirmed C&C nodes on the network can be cleaned and all communication with any outside the network can be stopped. When possible, the ISPs that host these C&C nodes should be notified of the abuse.

The tests conducted in this study observed communication patterns of one known HTTP botnet and one botnet capable of using P2P communication. The system must be continually updated with parameters for as many known botnets as possible, as was done to manage IRC botnets with CS-MARS. An ongoing botnet detection, containment, and suppression program should include the following:

- Monitor HTTP DNS names used by botnets
- Monitor ports used by botnets
- Block botnet HTTP servers like IRC
- Generate CS-MARS incidents and respond promptly

Implementation of these measures on a large enterprise network will reduce the resource utilization and risk of damage from attacks caused by many of the botnets known today. Additionally, known botnet hunters, particularly those participating in the botnet mailing list system can be an excellent resource for the identification of botnets and opportunities for containment.

This study has not addressed identification of the stealthier botnet behaviors as those advertised in a message received on April 20, 2006 by a member of the botnet mailing list [15]. The sender of this advertisement promised those who would buy his botnet services, bulletproof web servers and five IPs that change every ten minutes with different ISPs. Tomorrow's botnets are expected to hide their activities with encryption, in VoIP traffic, and even in proprietary protocols such as Skype [18].

An article published in *USA Today* on April 24, 2006 states that millions of PCs may be under the control of botnets [19]. Many of these compromised systems are home machines without the protective oversight of skilled technicians. Prevention of infection will continue to demand both prompt elimination of software vulnerabilities through patch management, and the reduction of the effectiveness of social engineering through user awareness training. Identification,

containment, and suppression of botnet activity will be a long-term and constantly evolving effort.

ACKNOWLEDGMENTS

I thank G. Allison for information about deterring IRC-based botnets, access to the CS-MARS system, and permission to examine network traffic.

REFERENCES

- [1] H. Orman, "The Morris worm: a fifteen-year perspective", *Security & Privacy Magazine*, IEEE, September-October 2003 Pages 35-43
- [2] A. Bryant, "Alleged Botnet Crimes Trigger Arrests on Two Continents", *PC World*, November 4, 2005, <http://www.pcworld.com/news/article/0,aid,123436,00.asp>
- [3] "Botnets", *Uninformed*, Volume 1, May 2005, <http://uninformed.org/index.cgi?v=1&a=4&p=14>
- [4] "An Inside Look at Botnets", Paul Barford and Vinod Yegneswaran (University of Wisconsin, Madison), http://www.cs.wisc.edu/~pb/botnets_final.pdf
- [5] N. Elton, M. Keel, "Who Owns Your Network", 2005, <http://www.educause.edu/ir/library/pdf/SPC0568.pdf>
- [6] The HoneyNet Project & Research Alliance, "Know Your Enemy: Tracking Botnets", March 13, 2005, <http://www.honeynet.org/papers/bots>
- [7] J. Li and T. Ehrenkrantz (University of Oregon), Geoff Kuenning (Harvey Mudd College), and Peter Rieher (UCLA), "Simulation and Analysis on the Resiliency and Efficiency of Malnets" Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05) 1087-4097/05
- [8] N. Ianelli, and Aaron Hackworth, "Botnets as a Vehicle for Online Crime", CERT® Coordination Center, December 1, 2005
- [9] The HoneyNet Project, <http://www.honeynet.org>
- [10] Cisco, "Distributed Denial of Service Threats: Risks, Mitigation and Best Practices", http://www.cisco.com/en/US/netsol/ns480/networking_solutions_white_paper0900aecd8032499e.shtml
- [11] F. C. Freiling, Thorsten Holz, and Georg Wicherski, "Botnet Tracking: Exploring a Root-Cause Methodology to Prevent Distributed Denial-of-Service Attacks" Department of Computer Science Technical Report RWTH Aachen University April 2005, <http://sunsite.informatik.rwth-aachen.de/Publications/AIB/2005/2005-07.pdf>
- [12] Symantec, <http://www.symantec.com/index.htm>
- [13] Lurhq, <http://www.Lurhq.com/>
- [14] SANS Internet Storm Center, <http://isc.sans.org/>
- [15] Botnets mailing list, <http://www.whitestar.linuxbox.org/mailman/listinfo/botnets>
- [16] Ports, http://www.bekkoame.ne.jp/~s_ita/port/port1-99.html
- [17] DNSWatch, <http://www.dnswatch.info/>
- [18] J. Leyden, "Botnet controls fears over IP telephony", *The Register*, January 26, 2006, http://www.theregister.co.uk/2006/01/26/voip_botnet_control_fears/print.html
- [19] B. Acohido and J. Swartz, "Malicious-software spreaders get sneakier, more prevalent", *USA Today*, Section B page 1-3 April 24, 2006