Abstract

Botnets and their attacks pose a significant threat to the Internet. If these are launched simultaneously at a large number of systems, they pose greater risks. Bots are generally designed to carry out a variety of attacks like spamming, traffic sniffing, harvesting information etc. Botnet is a collection of bots that act under the control of a head bot or herder to launch an attack. Although much has been written on various aspects of botnets, this paper concentrates on exploring the ease of recruiting bots, operating a botnet, and also discusses the threats of such botnets. As detection techniques improve, botnet design will continue to evolve to evade detection; thus, it is important to predict potential future botnet models for the purpose of developing defense mechanisms. This paper presents several strategies for botnet defense mechanisms and activities which should be carried out in order to establish successful defense.

Keywords

Botnets, Computer Security, Malware, Network Security

Introduction

The continued growth of the Internet has lead to an explosive increase in the number of Internet attacks. There is, however, a shift in the motive for the attack from curiosity and recognition towards financial benefit [Choo, 2007]. This shift is attributed to the increasing sophistication of attack tools, thereby calling for a more secure world. Lately, a large number of articles and journals discuss the devastating criminal activity of botnets. Botnets has risen to be one of the scariest, continuously evolving and widely discussed blended threats to the Internet world today. Bot, short for robot, is a software script that automatically installs on a victim machine without any user intervention and carries out commands sent from another machine. Bots are not inherently evil. Infact, Google uses a search bot called Googlebot that gathers documents across the Web in response to search requests. However, in this paper, we are only concerned with vicious botnets that perform a whole bunch of major attacks and cause a violation of the security policy.

A botnet refers to a set of compromised machines also known as zombies that work together to intensify the effect of their attacks. These compromised computers can be located anywhere in the world such as homes, schools, businesses, etc. Attackers may use these zombies as anonymous proxies to hide their real identities and can easily amplify their attacks. The individual bots are software programs that run on a host computer allowing the other bots to control the actions of the host. A host itself may be infected by several bots and be a member of several botnets. Controllers can use these bots to perform various types of attacks including distributed denial of service (DDOS), email spamming, key logging, abusing online advertisements, installation of spyware, spreading new malware, and identity theft [Andreea, 2010].

An infected computer (zombie), while carrying out malicious code, spends the resources and obeys certain commands without permission and knowledge of the owner and that activity causes the computer to slow down, display mysterious messages or it can even cause the system to collapse [Symantec, 2009]. The biggest problem with botnets appears when they are used for an attack. Botnet of a million robots, with uploading speed of 128 Kb/s per infected computer (zombie), can reach a size of
128 gigabits in traffic. It is enough to put out of function 500 companies and several countries by applying DDoS attacks. If several big botnets unite, they could threaten functioning of national infrastructure of most countries [Ken, 2006].

Botnet can be either small or big, depending on complexity, sophistication of the robots and number of computers involved. The value of a botnet depends on its ability to use distributed computer power and its ability to retain anonymity through the use of a multi-tier command and control (C&C) structure.

In this paper, we look at several aspects of botnets, life cycle of a general botnet infection, various C&C mechanisms such as IRC, and a few botnet infection methods. A variety of widely and newly discovered approaches for defending botnets will be presented. This involves strategies at different stages such as prevention stage, detection stage and response stage.

Botnet Communication methods

Centralized Model:

Distributed computing allows a user to divide the work for accomplishing a task across the resources of many hosts. A DDOS attack by a botnet comprised of several hundred thousand bots could cripple any computer defense. A botnet composed of a million or more hosts could be used as an attack by terrorists or as an act of war against a nation to cripple its network infrastructure. [Staniford, 2002]

By using a multi-tiered C&C structure the bot herder is able to hide its identity. A bot does not perform any action of its own. Instead, it waits for commands from the botmaster that it follows promptly. The botmaster is most likely a human operator who issues commands to the bots via a C&C server. Typically, the commands are of two types: attacks and updates. While the attack command instructs bots to perform an attack, the update command instructs bots to download a file from the Internet and later execute it. Bots propagate to other machines by exploiting vulnerabilities. Additionally, most bots also have an executable packer, which encrypts and compresses the actual binary executable, hiding it from the general anti-virus software.

Figure 1 shows the distribution of botnet C&C servers (as dots) across the world as captured by Shadowserver Foundation [Shadowserver 2009], a group of security professional volunteers who gather and report botnet activity, as of December 5, 2009. The size of the dot represents the relative number of C&C centers in that region. Evidently from the picture, the largest number of C&C centers is located in the United States of America, indicating that it has the most bot-infected machines.

Figure 1: Botnet C&C Servers Map (Source: Shadow Server, http://shadowserver.org)
Decentralized Model (P2P):

Besides botnets which mostly use centralized command and control, there are botnets which use peer-to-peer control (P2P), and in that case there is not a central controller (Botmaster) [SANS, 2009]. After the initial attack begins, these botnets create a “chain reaction” causing “Storm” and that’s why these botnets are called “Storm” [Carlton, 2008]. P2P botnet communication has several advantages over centralized networks. P2P communication system is much harder to break up compared to other type of communication systems. It means single bot compromise will not destroy the entire botnet. However, the design of P2P communication systems is very complex and there are no guarantees on message delivery or latency. There is another type of botnet called the random botnets. Table 1 provides information about the various types of botnets.

<table>
<thead>
<tr>
<th>C&amp;C Model</th>
<th>Description</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Centralized</td>
<td>A central C&amp;C server communicates with all its bots</td>
<td>1. Low message latency due to well known hops 2. System design is simple</td>
<td>1. Easier to detect since all bots connect to the same server 2. Single point of failure, so easy to pull off</td>
</tr>
<tr>
<td>P2P</td>
<td>Every machine acts as both client and server</td>
<td>1. Relatively harder to disrupt 2. Message delivery is not guaranteed</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Classification of C&C Models Based on Communication Method

Botnets can also be described on the basis of the protocols they use. The most commonly used virtual C&C center for botnet propagation is Internet Relay Chat (IRC). IRC is an Internet protocol that enables real-time text messaging, it is a text-based open protocol developed for users to “teleconference”. IRC, created in 1988 by Jarkko Oikarinen, is an open protocol that is mainly used for group communication in discussion forums called channels [Srdjan, 2009]. Many IRC networks have been established at universities and other institutes. The program mIRC and other IRC clients are readily available to anyone who wants to use the IRC networks.

A typical IRC network consists of IRC clients that connect to the IRC server. IRC servers are interconnected to other IRC servers and are mainly responsible for transferring messages between their clients. Each discussion on the IRC server occurs on a different channel. IRC bots run on victim user machines and join the pre-configured channels where they wait for commands from the IRC server. The IRC server is usually a compromised machine that the attacker uses to launch attacks.

IRC is the most common technology used by attackers to propagate bots. The main advantage of using IRC as C&C is that it is very easy to use and implement. Moreover, a large number of public IRC networks already exist, requiring minimal effort from the attacker. Private IRC servers are usually hosted on bulletproof hosting services that can never go down. Web based C&C servers are the second type in this classification. Here, the attackers use a Web interface to monitor and control their botnets. The last type in this classification is a DNS based C&C server, which allows the attacker to send commands over a covert channel in DNS. Since the DNS traffic is permitted through firewalls, the C&C traffic looks genuine and cannot be recognized. Some botnets also use HTTP protocols.

Life Cycle of Botnet Infection

Figure 2 illustrates five common stages found in a typical IRC botnet infection. In the first stage, a Bot looks for additional victims by exploiting vulnerabilities in the victim’s machine. Most Bots look for security holes in the operating systems. Once vulnerability is found, the victim is tricked into downloading a script called shellcode that causes the bot binary to install and automatically start each time the victim restarts his machine.
The bot then tries to connect to the C&C center, which in this case is the IRC server. It typically has to go through a DNS server to resolve the name of the IRC server. This additional step will enable the botmaster to hold on to its botnet even though its IP address is blacklisted. Generally, the bot is required to authenticate itself to both the IRC server and the channel it is trying to get into. The passwords for these authentication phases are already available within the bot binary. Additionally, the botmaster has to authenticate to all its botnets. This phase is required to ensure that the botmaster has not been seized by other botnets. When a bot joins a channel, the botmaster updates the IRC with commands for the bot to execute.

**Botnet Attacks**

If anyone wants to bring down a country’s network infrastructure and without anyone knowing who did it, the weapon of choice is a distributed denial of service attack. Using rented botnets, one can launch hundreds of thousands or even millions of infobombs at a target, all while maintaining total deniability. An extensive list of attacks has been discussed in [Keizer 2008] [Larkin 2008] [Ianelli 2007] [Romano 2005] and [Puri 2003]. This list is growing everyday as new capabilities are incorporated into bots. Understanding the botnet attacks will help in analyzing botnet defenses in a better way. This section will describe some of the common attacks.

**DDoS (Distributed Denial of Service):** The goal of this attack is to exhaust the resources of a server in such a way that the server will be unable to process any further requests from genuine clients. Variants of DDoS include ICMP, SYN and UDP flooding. DDoS attacks are similar to DoS attacks where someone is simply sending packets as fast as possible, but DDoS attacks can originate from thousands of computers at once.

**Spamming:** The most ideal way to propagate Spam is through the use of botnets. Spam mail is transmitted to the bot and later to other machines.

**Phishing:** As more people are becoming victims to phishing attacks, bots now have in-built ability to redirect users to fake websites where they would be asked to enter personal and confidential information.

**Host illegal data:** Attackers use the victim machines to host illegal software, movies and files. Even if the victim machine is traced for illegal activity, the attacker is safe and can re-host the files on another victim machine.
**DDoS extortion:** This attack is a sample DDoS threat to a commercial server. The goal is to obtain extortion money for not extending the sample attack to a real one.

**Click fraud:** This attack tricks the user into clicking a Web link that is directed to Internet advertisements of affiliates there by increasing the affiliate revenues paid by advertisers.

**Gateway and proxy functions:** Sometimes attackers host server class services to avoid being detected. In generic port redirection where all incoming traffic is redirected to another machine. The idea here is to obfuscate the real location of the attacker.

**Keylogging and sniffing:** Bots can be programmed to capture all the data entered via the keyboard, which usually is confidential. Similarly, bots can sniff the traffic to intercept valuable information.

**Screen capture:** A bot can capture screens on the trigger of an event. Some bots are capable of capturing audio as well as video feeds.

### Botnet Defense Mechanisms

Botnet attacks can be defended at three stages: prevention, detection and response. All the Internet users are responsible for defense, starting from home or business computer users, system administrators, developers, and up to Web service/application administrators.

Preventing computers from becoming part of a botnet, i.e., avoiding infection by Internet worms and other malicious software will be the most effective defense. To prevent a botnet, security awareness must be spread among users. In addition, users should implement multi layer security and start using secure operating systems with updated patches. Setting the auto-patch update facility will free the user from the burden of manually updating patches each time it is dispatched. Turning on the firewall will add another layer of security. Users should learn to safely handle web browsers and e-mail applications.

To detect a botnet attack, a user can use a third-party online resource for scanning the machine, or use commands like netstat to monitor any malicious traffic. System administrators should monitor logs generated by Intrusion Detection Systems (IDS) and look for traffic anomalies.

In response to a botnet attack, a user must disconnect from the machine and update anti-virus software. Any confidential information stored on the machine should be changed immediately. System administrators should take an additional step to avoid spreading of the botnet on the network.

Honeynet [Krasser, 2005] is an advanced technique used to track and analyze botnet activity. Honeynet is a collection of honeypots. Honeypots are systems designed with vulnerabilities ready to be attacked. Honeypots are meant to collect information about attacks and later analyze the data to design better techniques to protect against threats. A honeynet has three parts to it: data control, data capture and data analysis. **Data control** refers to the containment of activity i.e., not allowing it to harm other non-honeynet systems. Data control can be implemented using multiple layers of protection to avoid the consequences of a single point of failure. **Data capture** is the process of monitoring the botnet activity within the honeynet. **Data analysis** is the extraction of valuable information from the data captured. An additional requirement is data collection, which is applicable only to organizations that have honeynets distributed in multiple locations. Such organizations are required to collect the data into a central location before they can start analyzing it.

Figure 3 shows the main components of a typical honeynet. The firewall logs all incoming and outgoing connections, provides Network Address Translation (NAT) services, and some protection against DoS attacks. The IDS also logs all the network traffic and looks for known attacks and exploits. A syslog computer logs all the activity and commands carried out in the attack. This information is then stored in a remote syslog computer to avoid loss of data in case of compromise of the local
syslog computer. The last component is the honeypot. It serves as a mirror of a production system. A honeynet usually contains more than one honeypot and it not only gathers information for analyzing but also alerts security organizations when new attacks are detected.

![Figure 3: Structure of a Classic Honeynet (Source: http://honeynetindia.wordpress.com/)](http://honeynetindia.wordpress.com/)

Bots based on P2P architecture could be detected by analyzing and monitoring bots features. In the P2P botnets one can monitor specific range of ports, network traffic etc. Significant evidence to identify the existence of botnet will be IP address lists used by the botnet and also connection failures of bots in the infection step. Since each bot in P2P architecture attempts to link to others, it establishes a number of traffic connections in order to find bot peer and to exchange information. Furthermore, these generated traffic connections by bot have similar degree of fluctuating fixed information. This fixed form is useful to categorize TCP and UDP packets and to group the data which is used to classify the activities of bots.

Large botnets are frequently used to launch DoS attacks. To bring down an e-commerce website or to prevent an organizations networking capabilities, these attacks require several resources, such as a botnet army. Unlike large botnets flooding a network to deny service, micro-botnets are less likely to be detected. Micro-botnets utilize fewer slave computers, and in turn send fewer data packets, and are superior at evading traditional botnet-detection capabilities in firewalls and intrusion detection systems. An organization, to protect itself from micro-botnets must allocate more resources toward detecting botnets rather than focusing solely on preventing them. To detect botnets an organization needs to monitor for any abnormal spikes in network traffic, weird open ports, and accounts suddenly gaining elevated permissions. If a pattern scanner is used, sensitivity level should be turned up and extra time should be spent on determining what is or is not a false positive. Good network hygiene is to exercise log analysis to know what's really happening on the network.

Botnets are polymorphic i.e., their signatures change with each new infection. Also, most bots now include rootkit facility, which hide them from security tools. Hence, using anti-virus software will not help to a large extent. Most popular websites are now capable of spreading malware stealth fully. Therefore it is not just sufficient to be careful while surfing. An effective approach to reduce the botnet activity is to force-feed security upgrades and patches into machines connected to the Internet. It is also very essential to educate users on the importance of security and steps they need to take to keep their machines secure, considering that users are the weakest link in the security chain. Honeynets is a certainly a promising way to analyze...
and defend botnet attacks. However, it again depends on how the honeynet is implemented and to what extent it can contain and track the botnet activity.

Conclusion

Botnets pose challenges for the Internet community. It is a constantly evolving threat that is causing devastating effects on the Internet. The study of botnets shows the growing trend of criminal activities. Bots running on infected machines can use automated methods to propagate and perform a number of nasty deeds like sending Spam, and launching DDoS attacks. The motive behind such criminal activities is to propagate infection to more number of users in a stealth way. With the growth in the use of mobile platforms for all kinds of applications, security issues are not far behind. Trends indicate that the next targets for botnets will most likely be mobile devices.

Botnet armies are at large caused by lack of user awareness and vulnerabilities in operating systems, specifically in the Windows family of operating systems. Although many sophisticated defenses are being proposed and implemented, minor solutions to improve Internet security are being overlooked. One approach to reduce the effect of botnets is to force-feed security upgrades and patches into machines connected to the Internet. It is also very essential to educate users on the importance of security and steps they need to take to keep their machines secure. A thorough and updated security checklist should be provided to users on a timely basis. Users who do not pass a minimum security check should be cut off from the Internet. The goal of our study is to understand how botnets work, its effects, and defense mechanisms. Future work will concentrate heavily on the defense mechanisms.

References


Choo, K. R., and Australian Institute of Criminology. 2007 Zombies and botnets / Kim-Kwang Raymond Choo Australian Institute of Criminology, Canberra.

Ken Baylor, Chris Brown, Killing Botnets, McAfee, October 2006


S. Staniford, V. Paxson, N. Weaver, “How to Own the Internet in Your Spare Time”, Proceedings of the 11th USENIX Security Symposium (Security '02) 2002
