

Think Outside the Botnet

By Benjamin Tuttle
tuttleb@gmail.com

Introduction

It's common to hear warnings about not keeping your PC up to date with all of the newest security patches, antivirus definitions, and firewall updates. Unfortunately, it's also common to ignore these warnings. Consensus is that it's a chore to update software, and it's better to spend the fifteen minutes required for updating software doing just about anything but actually updating software. This philosophy is the reason that IRC bots and botnets are becoming an increasingly popular and ruinous tool for the hacker community.

A botnet is defined as a collection of software robots, or bots, which run autonomously, and can be remotely controlled as a group, usually through a means such as IRC, and usually for nefarious purposes¹. Approximations for the number of botnets on the Internet are currently in the multiple tens of thousands, and some of these botnets are known to contain more than 50,000 bots. Given these statistics, it would be a conservative estimate that there are 1,000,000 compromised computers (computers that are the unknowing hosts for bots) doing the bidding of hackers worldwide. The enormity of this number is staggering, especially when considering what can be accomplished with a large number of bots. This article is intended to explain exactly what bots and botnets are capable of, and then to explain the different types of bots that are currently at the disposal of hackers worldwide. This intent will be realized by discussing the most typical uses for botnets, including DoS (Denial of Service) attacks, traffic sniffing, keylogging, illegitimate advertising, spamming, phishing, and spreading. Following this discussion, Phatbot, SDBot, and GT-Bot (as well as their variants) will be discussed and analyzed. Although this article won't go into every detail of every IRC bot, it will provide a sufficient foundation of knowledge to get a reader up to speed quickly on the botnet problem facing both the security community and the public at large.

What Bots are Used For

There are as many uses for bots as there are motives for hackers, and as such there are constantly new and ingenious, albeit malevolent, ways being discovered for bots to steal and cause chaos. The following list represents the most commonly employed practices for bots and botnets.

DoS Attacks

The most typical use for a botnet is to cause a DoS (Denial of Service) attack. A DoS attack can come in multiple flavors including a regular DoS attack, a DDoS (Distributed Denial of Service) attack, or a DRDoS (Distributed Reflection Denial of Service) attack. A regular DoS attack involves a single bot attacking another machine. This attack could consist of several different bandwidth or resource attacks, but the most common attack is to flood a target

with UDP, ICMP, or TCP SYN packets. Regardless of the particular protocol used, all DoS attacks have the purpose of causing the target to lose network connectivity, resulting in a loss of service to users. A DDoS attack uses the same basic approach as the DoS attack, but is much more effective as it uses many (sometimes tens of thousands) bots to attack a single target. It is quite easy to see that this type of attack has a far greater probability of bringing down a target than a standard DoS attack. It is for this reason that the DDoS attack is the primary type carried out by botnets. This strength in numbers is also the reason that botnet owners, which from this point on will be referred to as zombie masters, try to obtain as many bots as possible. Like a DDoS attack, a DRDoS attack uses many bots to carry out the attack; however, that is where the similarities between the two end. Where a DDoS attack would send SYN packets directly to the target with a spoofed return IP address, a DRDoS attack will send SYN packets to a TCP reflection server and falsify the return address to be the IP address of the target. This causes the target machine to receive a flood of SYN/ACK packets from valid TCP servers. The advantage of this approach is that a TCP server will send up to four SYN/ACK packets for every SYN packet it receives (the TCP server assumes that its SYN/ACK packets were lost when it does not receive an ACK packet in return). Furthermore, this attack makes it more difficult for the victim to trace the attack back to the attacker because of the added layer of complexity.

The targets for any of the above types of DoS attacks can be individual Web sites or virtually any other service found on the Internet. The people who carry out these attacks are commonly referred to as script kiddies, and they are known to come from all walks of life. There have been cases ranging from where a pre-adolescent boy brought down corporate Web sites for fun, to where paid commercial DDoS attacks were targeted against competing companies.

Traffic Sniffing

Many bots have the built-in ability to sniff network traffic. This enables the zombie master to see any sensitive information that happens to be unencrypted, such as usernames and passwords. Often times, when a computer is vulnerable to a certain type of attack, there will be multiple hackers trying to exploit the weakness; because of this fact, a compromised computer could be infected with more than one bot. If this is the case, the traffic sniffing capability of a bot can enable the zombie master to see the connection settings for another hacker's botnet. This information can then be used in attempt to steal bots from the other hacker.

Keylogging

If the network traffic that is being sniffed (as described above) is encrypted, then it is of very little use to a zombie master without the key

required for decrypting it. Keylogging is the zombie master's attempt to bypass this shortcoming. Zombie masters can implement filtering systems to only log key presses that occur at bank Web sites, PayPal, or other financial institutions. If a zombie master has several thousand bots at his disposal, it is very likely that he will be able to collect a large amount of sensitive financial data and use it for personal financial gain.

Illegitimate Advertising

Zombie masters will frequently set up Web sites that host pay-per-click advertising; they will then command all of the bots in their botnets to go to the Web site and click on these advertisements. The end result of this command is that several thousand clicks occur in a very short amount of time. One Quebec-based company pays the pay-per-click advertising. Using the revenue calculator on their Web site, and assuming a collection of 10,000 bots (all of which have US IPs), it is possible to make between \$6,060–\$60,600 USD a month. Although not every zombie master will have that many bots at their disposal, they will in all likelihood use more than one company to pay for advertising; for this reason the figure presented above is an achievable monthly income for a zombie master.

Spamming and Phishing

One of the most classic uses for bots has been to send out spam e-mails. Armed with thousands of bots, a zombie master is capable of sending out many thousands of e-mails. Many bots have the ability to harvest e-mail addresses themselves. It should be noted that it is common for a zombie master to open a SOCKS v4 or v5 proxy on the compromised computer before spamming. In addition to regular spamming, bots are increasingly being used to send out phishing e-mails. Phishing e-mails pretend to be from a legitimate source, such as a bank or PayPal, and require the intended victim to submit usernames, passwords, and other personal information. Zombie masters have become quite adept at making phishing e-mail look legitimate, and thousands of people are tricked by them. The phishing emails usually contain a link which appears to connect to an institution's legitimate website; in actuality, the Web site is hosted by the zombie master. If the victim enters his/her personal information on the Web site, the zombie master then has access to everything that was entered.

Spreading

Most of the more powerful bots have the ability to spread themselves to other vulnerable computers. The bots that have this functionality can scan all computers within a specified network range for vulnerable ports. When a vulnerability is found, the bot can exploit it to spread itself, similar to the way a worm propagates. This process allows botnets to grow without any interaction from the zombie master. When the zombie master wants to add new features to an existing bot, he can command the bots to connect to a specified URL to download and execute a file. In this way a zombie master doesn't need to re-infect computers every time an update is desired, he simply has to provide a link to the newly coded bot binary.

The Various Types of Bots

There are several different types of bots that can be found on the Internet, and a myriad of versions of each type, but there are three main categories that most bots fall into: Phatbot, SDBot, and GT-Bot. Although

this list is not entirely comprehensive, these types of bots account for the vast majority of the bot problem.

Phatbot

Phatbot, and its variants (Forbot, Agobot, Gaobot, XtremBot, etc.), are some of the most powerful bots currently in existence. There are known to be more than 600 different variants of this bot, and there are probably more created every day. All versions of Phatbot are published under the GPL, are written in C++, and are capable of being run on several different platforms (Windows, Linux, etc.). Phatbot's source code is written very professionally, and it's easy to add new commands and functionality. It can be very difficult to detect Phatbot on a compromised computer due to the fact that it implements file and process hiding. Also, it can be very difficult to reverse engineer Phatbot because it can detect both being run in a virtual machine and debugging programs such as SoftICE.

SDBot

SDBot, and its variants (UrBot, RBot, Randex, etc.), tend to have similar functionality to Phatbot, but typically SDBot is not as powerful and doesn't have as many commands. SDBot is written in C, and the code is not as easy to follow as Phatbot; nonetheless, SDBot is an extremely popular tool in the hacker community. Like Phatbot, SDBot is published under the GPL and there are hundreds of different versions that can be found on the Internet.

GT-Bot

GT-Bot is the generic term used for any bot that is mIRC-based (mIRC is an IRC client for Windows). GT-Bots launch a hidden instance of mIRC that is preconfigured with a zombie master's desired connection settings (the location of the botnet). As with Phatbot and SDBot, there are many different versions of GT-Bots that can be found on the Internet. Although GT-Bots have similar functionality to Phatbot and SDBot, they are limited to targeting Windows machines due to the fact that they are mIRC based.

How Bots Operate

Now having sufficient background information as to what bots are and what they are capable of, it is time to discuss exactly how bots operate. Once a bot is installed on a host computer, it attempts to connect to an IRC server, or IRCd (short for IRC daemon). Many botnets use IRCds that are stripped down to only include the features that the zombie master requires. For this reason most full featured IRC clients won't be able to connect to these special case IRCds. The location of the zombie master's IRCd is hard coded into the bot itself and includes an IP address, a port number, and a channel to connect to. There will almost always be a channel password in attempt to keep non-bots out of the channel. When a bot connects to an IRC channel it typically starts to idle. Depending on how the zombie master coded the bot, it may or may not announce joining the channel. In channels where bots do not announce themselves, and they do not show up on the channel's user list, it is very difficult to get an accurate assessment of how many bots are actually in the channel. If a bot thief (a hacker that tries to steal a zombie master's bots) or security professional was able to infiltrate the channel, they would be handicapped in this regard; this is precisely the reason why the zombie master would set up his bots and channel to not display this information.

A bot will continue to idle in the IRC channel until it is given a command by the zombie master. Most bots will require authentication from the zombie master before it will accept any commands; this authentication comes in the form of a password, and is simply another layer of security to avoid bot theft. Once a zombie master is authenticated, he has free

reign to issue any commands of his choosing. These commands can cause the bot to perform any of the actions discussed above, as well as to carry out any specialty functionality that may have been implemented by the zombie master. It was mentioned previously that many bots have similar functionality; however; it should be noted that bot command syntax is varied significantly between the various versions of bots. Once again, the purpose of this variation is to avoid bot theft.


How to Defend Against Bots

The most important step that a user can take to ensure that their computer is not turned into an unknowing host for bots is to run fully updated antivirus and firewall software. There are several vendors of antivirus software, and many offer full security suites that contain both the antivirus and firewall software in one bundle. Before committing to a single brand, it's important to read reviews for the products under consideration, not all antivirus software is created equal. This does not mean going to a vendor's Web site and reading the praise for their own product; find a third-party Web site and read the honest reviews and comparisons. Deciding on a product is only half the battle; it's very important to keep it up to date with all of the newest security patches, antivirus definitions, and firewall updates post install; neglecting to do this defeats the purpose of having antivirus and firewall software in the first place.

On top of having antivirus and firewall software, there are free bot scanning utilities, such as SwatIt, that can be downloaded and used in much the same way as antivirus software. Another way to determine if a computer is infected with bots is to view the current TCP/IP connections in attempt to see if there is an IRC session in progress. On a Windows machine this can be accomplished by entering the `netstat -an` command at the Command Prompt, which will numerically display all the network connections and listening ports. Historically, IRC server connections have used port 6667 and local Ident servers have used port 113. If a user is not running an IRC client (that they know of) and they observe the above ports being used after issuing the `netstat -an` command, they could conclude that they are indeed infected with a bot. Unfortunately, zombie masters have started using random ports and this trick is not as useful as it once was.

Conclusion

DDoS attacks, traffic sniffing, keylogging, and phishing are all powerful attack and exploitation techniques on their own accord. The fact that a single bot can have the ability to use all of these techniques is a testament to how powerful a tool it is to a hacker. Only after compounding this fact with the ability of a bot to spread itself in an automated fashion, and then to join forces with thousands of other bots in a botnet, is the true danger of a bot finally realized. It is of the utmost importance that this danger not be underestimated.

Although bots and botnets represent a big problem on the Internet today, there are relatively simple measures that can be taken to ensure that computers are not turned into unknowing bot hosts. These measures include not only running antivirus and firewall software, but taking the time to keep your PC up to date with all of the newest security patches, antivirus definitions, and firewall updates. Also, there are free bot scanning utilities that can be downloaded from the Internet to provide an extra layer of security. 

Benjamin Tuttle currently works for Lockheed Martin.

¹ <http://en.wikipedia.org/wiki/Botnet>