

UNDERSTANDING AND TEACHING BOTS AND BOTNETS

Randy Abrams

ESET Research, 610 West Ash Street, Suite 1900,
San Diego, CA 92101, USA

Email abrams@eset.com

ABSTRACT

Bots and botnets suck, so what better teaching aid to help people understand them than a vacuum cleaner?

The second in the 'Understanding and teaching...' series, this presentation is designed to both educate those who are not familiar with the topic or have misconceptions, and at the same time to present effective methods to take a technical topic and present it in a user-friendly manner that those who are not fluent in geek-speak can understand.

Where the Catahoula Leopard Dog was used (along with other canines) in 'Understanding and teaching heuristics', this presentation will use *iRobot Roombas* to help explain what a bot is and what botnets are. Command and control your Rumbas from the comfort of your office, or let them create their own peer-to-peer network to perform attacks on unwitting domesticated animals.

Once users understand what can be done by bots and botnets, and the real risks presented by the malware on their computer, they will be more likely to become interested in safe computing practices.

Just as students drop out of universities, it is not expected that everyone will learn from the opportunity to be educated. However, when information is presented in an interesting, relevant and entertaining manner, the desire to learn and be more secure can be fostered in many users.

INTRODUCTION

In a recent survey by the National Cybersecurity Alliance it was reported that 71 per cent of users have never heard the word 'botnet' before. The harm done by bots and botnets is such that the public does need to have a basic understanding of what these threats are and what bots and botnets are capable of. As people become aware of risks they generally become more interested in mitigation. Public education itself will not solve the problem of bot-infected PCs and botnets, but it is a part of the fight against these criminal tools.

It can be extremely challenging to attempt to explain technical concepts to non-technical people. There are two goals for this presentation. The first is to be able to educate non-technical users as to what bots and botnets are, and what they are capable of. To reach this goal it is essential that technical jargon is reduced, as much as is possible, to understandable concepts. Analogies are essential teaching aids for this purpose.

The second goal of this presentation is to share a method of teaching a technical subject in a manner that relatively

non-technical people can understand. This presentation is not intended to teach anyone how to analyse bots or combat botnets.

BASIC CONCEPTS

Before one can begin to grasp the power of a bot or a botnet, one must understand that virtually any activity that can be performed on a computer can be automated. One must also understand some of the basic concepts of the functions of the programs that they use every day. Let's start with a few of these basic concepts.

When we use an email program the software captures each keystroke that we type on the keyboard. The letters and numbers are then placed into the email message. Word processors also capture the keystrokes we type and store them in a document. When one logs into an account, such as *MySpace*, *Facebook*, *MSN*, or *AOL*, etc. the keys that are pressed on the keyboard are captured and stored by another computer. The fundamental concept is that many programs record each keystroke that is typed on the keyboard. There is a type of malicious program called a keystroke logger. This program is invisible to the user, but if a computer has one running on it, then every keystroke can be saved or if the computer is online, the keystrokes can be sent to another computer as well. If a keystroke logger is installed on the computer and the computer is not online, the information can be stored and then sent later when the computer is back on the Internet. A program can be running and doing things, yet still be invisible to you. If you look at *Windows Task Manager*, the processes you see running are all programs, but most of them you do not actually see running.

When one finishes composing an email, one can then send it to one or more people. The simple concept is that a computer can be used to send email. There are many programs that can be used to send email. *Outlook*, *Outlook Express* and *Lotus Notes* are a few examples. Any skilled computer user can also write a program that sends email. It can also be invisible and the emails can be composed automatically to say whatever the programmer wants them to say. The emails can then be sent to people in an address book, or a list of email addresses can be downloaded from the Internet and then be used to send email to.

When we use *Internet Explorer*, *Firefox*, *Safari*, or another program to surf the web, we send data to other computers. If I go to www.google.com, my computer sends a request to *Google's* computer to show me their web page. Each computer can handle a finite amount of data, or requests for their web pages. There are also other types of requests for data that can be sent from one computer to another. Sometimes if a web page is very slow to appear it is because the computer you are trying to get information from is very busy. If too many people are requesting the web page at the same time you may get an error message that prevents you from visiting the website you wish to see.

Computers are great at being used to automate tasks. A digital alarm clock is really a computer that is used for a dedicated task. A person can program (set) the current time and also program (set) an alarm to wake them up at the same time each day. This is automation and was one of the early uses for a computer chip!

Almost anything you can do on a computer can be programmed to be done automatically. If you can respond to an email, a

program can be written to automatically respond to an email. An 'Out of Office' message is an example of a part of an email program that automatically responds to an email. If you can look in your address book for a person's email address, a program can be written to automatically do this too.

When you visit a website and click on a link for a web page, there isn't a person on the other end who is putting the web page up there for you; the computer with the web page was programmed to 'listen' for requests and then provide the information that is asked for.

With this understanding that almost anything a computer can be used for can be programmed to be done automatically, we can start to talk about what a bot is. The word 'bot' is short for robot. Some, if not all, of the earliest bots were programs that helped people automate dull, boring and tedious tasks on a computer. These bots were created on computers running the Unix operating system. Conceptually it may be difficult for a non-technical person to visualize a program automatically doing something, so using a visual example may help to explain the concept.

THE ROOMBA

We all have dull, boring, tedious tasks in our lives. Perhaps vacuuming is such a task for you. Luckily for you, you can buy a *Roomba*! A *Roomba*, in case you don't know, is a small robotic vacuum cleaner [1]. The *Roomba* is able to sense where walls are and effectively vacuum your house for you. You simply turn it on and tell it to go and then you leave it alone to do the vacuuming for you. You no longer need to push the vacuum across the floor; let the *Roomba* do the work and you can play or do other work. What happens when the batteries on the *Roomba* start to get weak? The *Roomba* is programmed to keep track of how much energy it has and if it starts to get low it will return to its charger and plug itself in automatically!

This sounds good so far, doesn't it? Now consider the situation in which you left the house to go to work, but forgot to tell the *Roomba* to vacuum the house for you. The ability to tell the *Roomba* to start automatically at a specific time each day would be very handy. So we contact *iRobot Corporation* [2], the manufacturer of the *Roomba*, and tell the product developers we would like a more programmable *Roomba*. The next version of *Roomba* now lets us program it so we don't have to think about telling it to start vacuuming. Each day the *Roomba* waits until the specified time and then leaves its charger to start vacuuming the house.

All is well, except that unexpected things happen. Perhaps we have a house guest and don't want the *Roomba* to wake them up by vacuuming at its pre-assigned time of 7.30 a.m. Wouldn't it be handy to be able to control the little robot from our office at work? That shouldn't be too hard. Just add a wireless Internet connection and a simple computer to the *Roomba* and we can now be at work and tell the *Roomba* when to start vacuuming. If we want to get really fancy we can make the program flexible enough that we can tell the *Roomba* which rooms to vacuum.

Perhaps vacuuming isn't enough. Why not attach a power sprayer to the *Roomba* and let it paint your walls for you? You might want to attach an arm to the *Roomba* and put a feather

duster in your *Roomba*'s artificial hand! The things you can add on to a *Roomba* are limited only by your imagination and bank balance!

A *Roomba* can only vacuum a limited number of rooms before it has to recharge. Perhaps I wish to complete all of the vacuuming more quickly, so I purchase two or three *Roombas* and set them all up with remote controls. To make sure that the *Roombas* don't duplicate each other's work I program them to tell each other what rooms they have already cleaned. By doing this, I no longer have to tell each *Roomba* what to do, I just give a general command to vacuum and away they go. Each *Roomba* has a list of the rooms in the house and each *Roomba* will start with one room, and tell the other *Roombas* which room it is vacuuming. When it finishes it will check the list to see what the other *Roombas* are working on and take another room to work on. One day I have a party at my house and the neighbours all see my little *Roombas* at work and decide that they too want to simplify their lives with identical *Roomba* armies.

Here is where the fun starts. Nobody stopped to think that I could 'log into' their *Roombas* too because there was no security built into the remote controls. I can control all of the *Roombas* in the neighbourhood since I programmed them and told them what to listen for. I can surf the *Roomba* web and command them all! Now I can make all of these *Roombas* attack the pesky neighbourhood dog that keeps digging in my yard!

BOTS AND BOTNETS

A bot is kind of like a *Roomba* for PCs. Bots can be programmed to run on *Windows*, *Unix*, *Linux*, *Macs*, or pretty much any general-purpose computer. A bot can be programmed to listen for commands over the Internet and to do a variety of tasks. There is virtually nothing you can do on your computer that cannot be done by a bot.

As I said earlier, many of the first bots were simply useful little programs that did good things. The modern bot is programmed not only to do things automatically, but also to listen to the Internet for commands. This means that the bot can be told to do a variety of things and their instructions can be changed very quickly. The program that tells the bot what to do is often called a command and control centre. The person who tells the bot what to do is called a bot master or a bot herder. When you have several computers that each have a bot installed, and each of these bots listens to the same command and control centre, this is what we call a botnet.

An easy way to think about this is the example of an army. A bot is like a soldier in an army. Each soldier belongs to one army and responds to one commander. The soldiers in the Swiss army listen to a Swiss commander. The soldiers in the US army listen to a US commander, and so on.

Before we discuss the dark side of bots, let's take a look at the largest botnet in the world. If you are a *Windows* user, your computer is probably part of this botnet, but don't worry, this is a good botnet for a computer to be a part of... usually.

Each month, on the second Tuesday, hundreds of millions of computers 'wake up' and start downloading security updates automatically. Most users never see this happening. This is

called 'Automatic Updates' and is provided by *Microsoft*. Yes, there is a program installed on *Windows 2000*, *Windows XP* and *Windows Vista* computers that is called 'Automatic Updates'.

These computers are programmed to 'listen' to the Internet for *Microsoft* to tell them to start downloading programs. Some of the types of programs that *Windows* computers have been instructed to download include security patches, a mini anti-virus program, and even a program that spies to see if you have a licensed copy of *Windows*. After complaints about *Microsoft* installing spyware on customers' computers, *Microsoft* changed their euphemistically named 'Windows Genuine Advantage' program to allow users to choose whether or not it runs.

Windows Update will silently download critical security patches, which are euphemistically called 'updates' to help protect your computer from hackers, viruses, trojan horse programs, and even bots. When the updates, which are computer programs, are downloaded from *Microsoft*, the computer is then instructed to run the update program to install the software. In theory, *Microsoft* could command all of these computers to do malicious things as well. All of the computers could be told that they need to delete almost everything on the hard drive. All of these *Windows* computers could be instructed to attack the *Apple* website! The computers could be told that they need to send spam to millions of people. This is simply theory. In practice *Microsoft* doesn't want to go out of business, so will not do anything like that. Still, *Microsoft* commands the largest potential botnet in the world! My computer runs *Windows Update* and I'm not concerned about it. I encourage you to make sure you have *Windows Update* enabled on your *Windows* PC.

THE DARK SIDE

So how did the bots get led to the dark side of computing? The precursor of the modern malicious bot, and subsequently botnet, was a type of program called a RAT. RAT stands for Remote Access Tool. Once again, this software has its roots in good. The idea of a remote access tool is that a system administrator can manage computers in a company without having to be physically present at each computer. Software was developed so that each computer could be managed remotely from another computer. The way this software works is that a program, referred to as a client, is installed on each user's computer, and a complementary program, called the server is used by the administrator. The administrator uses the server to connect to the client so as to be able to control the client (end-user's) computer.

The bad guys figured out that if they could install a 'client' on somebody's computer then they could control the computer from anywhere in the world. When one has to connect to each computer, one at a time, it is a tedious task and limits how much can be done, but computers are great at automation, so the RATs were modified to just listen to specific places for commands. Instead of the sergeant whispering in each soldier's ear, the sergeant yells out orders that all of the soldiers can hear. If the soldiers are not in the same location, radios set to specific frequencies can be used so that all of the soldiers know what the instructions are. Bots are basically remote access tools that have

been programmed to listen to the Internet and automatically carry out instructions.

Now we will delve into what bots and botnets are used for. There are a few primary purposes for bots and botnets, and then lots of other potential uses. One of the most common uses for bots and botnets is to send spam.

If I make a contract with a company to send out one hundred million email advertisements (spam), it will take my one computer a while to do this. When people complain, my Internet service provider might block my computer from using the Internet in order to stop me from sending spam. If I can tell 50,000 computers in 100 different countries to each send 2,000 emails then it becomes much more difficult to stop them, and I run almost no risk of getting caught. An extra benefit is that my own computer is not busy sending emails so I can use it to do other things!

Bot herders can get paid a lot of money to use your computer, and thousands of others, to send spam. This is one of the uses for bots and botnets.

A computer connected to the Internet is able to accept a specific amount of data at any given time. It is not uncommon in some places for a computer to be able to accept five megabytes of data at a time. When you connect to a website, the computer that your computer talks to is downloading information about what you are looking for. If you search for something on *Google*, then one of *Google's* computers has downloaded your request and will send you back (upload) the information it finds. When a person uses an online gambling site, the computer running that website has to download information about what the bet is. If that online gambling computer can handle 200 megabytes of data per second, but enough people are using the site that the computer has to receive 500 megabytes of data each second, then a lot of people will not be able to use the website. A bot master, or bot herder with 50,000 computers can force those computers to send massive amounts of data to an online gambling site. This means that legitimate users cannot connect to the site to gamble. This type of attack is called a distributed denial of service attack (DDoS). It is distributed because tens of thousands of computers are all attacking one computer, as opposed to one computer attacking another. It is called a denial of service attack because the attacked website is no longer able to serve its users. Why would someone cause an attack like this? There are a variety of reasons. Extortion is one reason. Once this type of attack starts, the attacker will anonymously contact the owner of the website and offer to stop the attack in return for a sum of money. The gambling site is losing money each minute that it cannot serve customers. Spammers will sometimes launch a DDoS attack against an anti-spam website to stop it from blocking spam.

Another motivation is revenge. If an employee is fired, they might want to exact revenge on their former employer. A DDoS attack is one weapon that can be used for revenge.

Bots can be used for attacks that are very dangerous to the owner of the infected PC as well. If a bot herder wants to sell access to illegal music files or child pornography, it is dangerous for him to have those illegal files on his own computer. A computer with a bot on it can be instructed to download and

share these illegal files with very little risk to the real criminal. If the police track the files they will find them on the victim's computer. This can lead to the arrest, prosecution and conviction of innocent people. The police often are not technically savvy enough to understand that the presence of a program on a computer does not always mean that the owner of the computer put it there.

Bots can also contain keystroke-logging programs and collect account information for your bank account, *PayPal*, stock accounts, and other personal information used for identity theft. As one types, the bot constantly records the keystrokes and usually saves them in a file to send to a remote attacker.

CONCLUSION

The best way to protect against bots is to learn about safe computing. File-sharing programs are among the most dangerous on the Internet today and pose a high risk to users who do not know much about security. Keeping the operating system patched is essential, as is keeping applications such as instant messaging, audio players, video players, and picture viewers patched. Discretion in downloading programs and visiting websites is also essential, however safe computing habits are beyond the scope of this presentation.

REFERENCES

- [1] <http://en.wikipedia.org/wiki/Roomba>.
- [2] <http://www.iRobot.com/>.