Bots and Botnets:
Risks, Issues and Prevention.

Martin Overton, IBM Global Services, UK

Email: ***overtonm@uk.ibm.com***
WWW: ***http://www.ibm.com/uk***

Tel: *+44 (0) 2392 563442*

Abstract:

Many corporate security staff have a rather vague understanding of bots and botnets, not just what they are but how they work. Furthermore many have little understanding of the risks to their company or their own home computer.

In many institutions and corporations bots and botnets are rife and causing significant damage to the infected network owner, both physically due to lost bandwidth, intellectual property and reputation; loss of credibility and brand damage.

This paper will explain what bots/botnets are and how they work. It will also discuss ways to combat them using methods that range from simple security methodologies through to technical solutions.

Disclaimer:
Products or services mentioned in this paper are included for information only. Products and/or services listed, mentioned or referenced in any way do not constitute any form of recommendation or endorsement by IBM or the papers author.

This paper was written for, and presented at, the 2005 Virus Bulletin conference at the Burlington, Dublin, Ireland between October 5th – 7th 2005.

I would welcome any constructive feedback on this paper and its content.

1 Introduction

This paper has been written for the corporate stream of the conference and therefore it will not delve into very technical details of bots and botnets. However, links will be used [where possible] to point the reader to more details on a topic.

Before we get stuck in and look at the relevant areas of concern and interest with regard to bots and botnets, let us quickly cover some terms that will be used throughout the paper. This will hopefully ensure that all readers be they; layman, technician or researcher will be all be able to get something useful from it. Other terms will be defined as required elsewhere in the paper.

To further aid those readers that have not been involved with IRC or have limited knowledge of bots and botnets, I will then present a short history of bots and how they evolved to what they are today.

1.1 Definitions:

1.1.1 Bot

'Bot' is a contracted (truncated or short) name for a software robot. A bot is a piece of software that allows a system to be remotely controlled without the owner's knowledge; it can also be used to automate common tasks such as on IRC[1].

1.1.2 Zombie

Another name for a system controlled via a bot; may also be known as a drone.

1.1.3 Botnet[2]

A group ['Herd' or 'Network'] of Zombie systems controlled by the 'Bot Herder'. These botnets are told what to do by the botnet owner. This can be anything that the bot has been programmed to do....including updating itself or installing new malicious software.

If you saw the film 'iRobot'[3], this is similar to the way the C-5 robots are controlled when commanded to carry out tasks that are in breach of the 'Three Laws of Robotics'.

1.1.4 Bot Herder

The person [or group] which "own" and control a herd of bots. Also known as the Bot Master aka Zombie Master.

1.1.5 DDoS [aka Distributed Denial of Service]

A distributed denial-of-service attack is an attack on a computer system or network from multiple co-ordinated systems connected to the same network which are performing a denial of service attack. The aggregated volume invariably causes a loss of service by consuming most, if not all the bandwidth of the victim network, and/or overloading the computational resources of the victims systems, such as web servers, etc.[4]

1.1.6 IRC

"Internet Relay Chat (IRC) is a form of instant communication over the Internet. It is mainly designed for group (many-to-many) communication in discussion forums called channels, but also allows one-to-one communication.

IRC was created by Jarkko Oikarinen (nickname "WiZ") in late August 1988 to replace a program called MUT (MultiUser Talk) on a BBS called OuluBox in Finland. Oikarinen found inspiration in Bitnet Relay Chat which operated on the Bitnet network.[5]"

---

[1] See section 1.2 'A Very Short History of Bots' for other uses of bots
[2] More data can be found here: http://en.wikipedia.org/wiki/Botnet
[3] Very, very, loosely based on the book written by Issac Asimov. Only a few characters are used from the book and the story used for the film does not feature anywhere in the book of the same name.
[4] More data can be found here: http://en.wikipedia.org/wiki/DDoS
[5] Source: http://en.wikipedia.org/wiki/Internet_Relay_Chat

1.2 A Very Short History of Bots

Back in the early 1990s IRC[6] [Internet Relay Chat] users created tools to automate certain tasks and to defend them against attacks; such as "net split". Other groups or individual users on IRC created attack tools to kill channels [chat rooms] and remove users from these channels, taking over their 'handles' [nick names] so that they couldn't get back in. Not surprisingly, new tools were developed to fight back…and so the bot was born.

By the close of the 1990s we started to see DDoS tools, these were used to attack IRC hosts and were largely non-automated tools; a few were semi-automated[7]. In both cases these tools needed a lot of tuning, coaxing and cursing to get them to do what the miscreant who was trying to use them wanted. Most of these tools although designed to attack IRC didn't use IRC to communicate by.

At the start of the new millennium Mafiaboy[8] used tools like the ones mentioned above to carry out DDoS attacks against Yahoo!, eBay, Amazon and CNN. The next phase was for the tools to be automated allowing larger networks to be created quicker than ever before and wielded with more devastating effect. To this end we saw the addition of worm and Trojan code to the mix. There are a number of examples of the offspring of this move; Staheldracht was bundled with the t0rnkit rootkit while the Lion worm included the TFN2K agent, being just two. During 2002 we saw the move to using IRC for command and control, or as a communication protocol of bots and botnets for those bots that didn't use IRC for command and control directly. What we were witnessing was the birth of the modern day bot.

From the beginning of 2003 we started to see further moves to incorporate many new ways for bots to get onto systems, and once there to stay hidden. This included the use of vulnerabilities, buffer overflows, droppers, bots that were fully share-crawling [SMB], used dictionary attacks, and dropped via other malware, and so on. Nowadays bots can be delivered in almost all the ways that current malware can.

Today we frequently see botnets used for extortion ["pay-up or we'll DDoS you off the 'net"], as SPAM and Phishing scam proxies [route e-mail through], identity theft [many bots have keylogging facilities, etc.] and malware seeding, amongst others. Although most modern [IRC] bots are designed to be almost invisible, and extensible they still comply with the RFC 1459 standard[9].

2 The Size of the Problem

The following are quite disturbing statistics for 2004[10]:

- Britain has largest zombie PC population in the world
- Over 1m connected computers are zombies
- 30,000+ internet connected zombie networks in 2004
- Estimated 25% of all infected PCs are under control of hackers
- Broadband responsible for 93% increase in infected PCs in 2004

Furthermore, a recent paper from the Honeynet project entitled: "Know your Enemy: Tracking Botnets" saw the following trends when observing and tracking over 100 botnets during a four month period [November 2004 until the end of January 2005]:

- They logged 226,585 unique IP addresses logging into one of the IRC botnet C&C channels.
- Botnets ranged from several hundred 'zombies' in size to several botnets with more than 50,000 'zombies'.
- They observed 226 DDoS attacks against 99 unique targets.
- Typical size of a botnet: 2000+ bots ['zombies'].

---

[6] IRC is a simple Instant Messaging or group chat tool.
[7] These include Trinoo, Tribe Flood Network, Shaft and Stacheldracht.
[8] The Canadian hacker responsible for a number of high profile DDoS attacks.
[9] RFC 1459 can be found here: http://www.irchelp.org/irchelp/rfc/rfc.html
[10] Source: http://news.bbc.co.uk/1/hi/technology/4579623.stm

- Logged 'zombies' being ordered to update themselves from websites, or to download and run new malware, such as an updated bot client software with more features; keylogging, CD key search, new exploit code [Lsass, RPC, etc.] or new malware for it to seed.

From this data they worked out that the number of bots required to successfully DDoS a typical company were just 13. This assumes that the company is on a T1 [1.544Mbit] and that each 'zombie' has a 128Kbit link [128Kbit x 13 = 1.664Mbit].

In June 2004 a large European IRC service recorded between 200,000 and 600,000 connections from bots each and every day. Of these they confirmed between 150,000 and 400,000 unique 'zombie' systems per day[11].

And more recently another group monitoring botnets used to send SPAM recorded that they had seen 1.5 Million compromised computers, with another 1 Million extra unconfirmed computers. They went on to estimate that it would take 5,763 man years to disinfect all of them and cost $600 Billion! [12].

Right, now let us look at the size of the problem in a different way, the size of the threat. Below you will see a table of common and widespread bot families, with the known number of members [variants] known at the time of this paper being written [May-June 2005].

Family	Number of Variants	Source	Notes
Sdbot	~12,800	McAfee	Last count, as McAfee no longer counts individual variants. Includes Forbot, Rbot, Wootbot and IRCbot.
Agobot	3,821	McAfee	+ 396 Non-viable. Includes Phatbot.
Spybot	2,116	McAfee	+ 69 Non-viable
Polybot	106	McAfee	+ 8 Non-viable
Mytob[13]	228	TREND	

3 Risks

Risk is a difficult area to cover. Most of us understand what constitutes a risk, however everyone's perspective on risk is different; one person's acceptable risk [such as mountaineering, motorcycle racing, bungee jumping, or keeping venomous animals (snakes, tarantulas and/or scorpions)] is another person's unacceptable risk. So, I will cover the areas of risk assuming that we are dealing with a company/institution/person who is inherently risk sensitive or averse.

So, let us for clarities sake look at each major area of risk that bots and botnets bring to the table; a relative feast awaits us.

3.1 DDoS

Denial of Service attacks have been around for a number of years and these have generally been replaced by DDoS attacks as these are more effective and harder to combat than plain DoS attacks.

A number of very large companies have been affected by DDoS attacks, these include:

- Amazon
- eBay
- Yahoo
- Microsoft

It is reported that 11% of small to medium sized businesses suffered DDoS attacks in the last 12 months[14]. Unfortunately there seems to be no figures for large businesses, academia or other institutions for the same period. Historical DDoS attacks however have been analysed and estimated

---

[11] Source: http://www.cscic.state.ny.us/msisac/webcasts/05_05/info/5_18_05presenter.htm
[12] Source: http://www.cscic.state.ny.us/msisac/webcasts/05_05/info/5_18_05presenter.htm
[13] The first version of Mytob was discovered in February 2005
[14] Source: http://news.bbc.co.uk/1/hi/technology/4579623.stm

loss data has been published. These include[15]:

- Microsoft in 2001; estimated at costing a cool $500 Million for two days of disrupted network connectivity due to a DDoS attack.
- An estimated cumulative loss of $1.2 Billion for the DDoS attacks against eBay, Yahoo and Amazon in February 2000.
- Cost predictions for a 24 hour DDoS attack on a large e-commerce company would be in the region of $30 Million.
- Average cost of mission critical services compromised $100,000 an hour[16]

In the early days, bots and botnets were mainly used to attack other botnets, IRC servers hosting competing bot creators and occasionally for attacking others; ISPs, vendors, corporates, etc. The trend now seems to be a 'for-profit' model of business when it comes to using DDoS attacks. The most common form of this is the co-called 'Cyber-Extortion' scam in which a company is informed that if they don't pay for 'protection' they will be attacked.

3.1.1   DDoS/DoS Types and techniques:

Syn Flood
This is the most common attack performed by Botnets. The zombies that make up the botnet are ordered to send TCP requests to a certain port on a specific IP address for a period of time. The idea here is to exhaust the connection queue. In many cases the IP address in the packet has been forged to make it look like it has come from an IP address other than the one it was created on. The effect of this forging [or spoofing] of the source IP address is to make it harder for the 'real' source to be found and neutralised. Also, if the spoofed IP address belongs to a system that is on a slow link, on the other side of the world or not connected to the internet then completing the 3 way handshake will be delayed or never completed.

You can see the difference in figure 1:
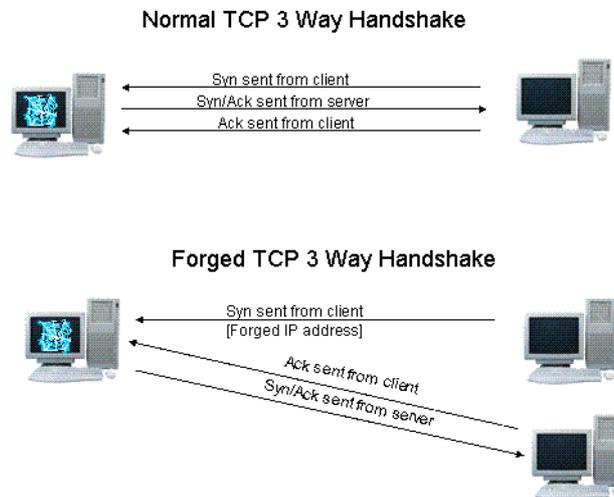
Figure 1. 3 way TCP handshake

Now, if you have a medium sized botnet, say of five to ten thousand zombies and they are all requested to continuously connect to a specific site, and use forged/spoofed IP source addresses, you can see how hard it will be for the victim to deal with this attack.

UDP Flood[17]
UDP is a connectionless protocol and it does not require any connection setup procedure to transfer

---

[15] Source: Yankee group, Forrester and IDC.
[16] Source: http://news.bbc.co.uk/1/hi/technology/4579623.stm
[17] Source: http://www.anml.iu.edu/ddos/types.html

data. A UDP Flood Attack is possible when an attacker sends a UDP packet to a random port on the victim system. When the victim system receives a UDP packet, it will determine what application is waiting on the destination port. When it realizes that there is no application that is waiting on the port, it will generate an ICMP packet of destination unreachable to the forged source address. If enough UDP packets are delivered to ports on victim, the system will go down.

ICMP Floods[18]

ICMP attacks can come in many forms. There are 2 basic kinds, Floods and Nukes.
An ICMP flood is usually accomplished by broadcasting either a bunch of pings (Not IRC pings, ICMP pings. Similar purpose, but handled differently) or UDP packets (which are used in software like PointCast). The idea is, to send so much data to your system, that it slows you down so much that you're disconnected from IRC due to a ping timeout.

Nukes exploit bugs in certain Operating Systems [OS], like Windows 95, and Windows NT. The idea is to send a packet of information that the OS can't handle. Usually, they cause your system to lock up.

3.1.2   Case Studies

The best documented case of an individual being regularly DDoSed is the strange case of Steve Gibson's site [www.grc.com][19]. Steve writes:

"*I determined that we had been attacked by 474 Windows PC's. This was a classic "Distributed" Denial of Service (DDoS) attack generated by the coordinated efforts of many hundreds of individual PC's*."

Steve found that a single individual was behind these attacks - a thirteen year-old from the United States wielding the power of a bot net.

3.2 Privacy

Many bots either already have Keylogger functionality or can be used to install keyloggers at a later date on a Zombie infected PC. Furthermore, some bots have the capability to sniff the packets which pass by or through the infected system. In fact most bots now are frequently used to download and install new components to quickly add new features and functions that may not yet be in the main executable. So, these can be either added as stand-alone programs, plugins, or if new functionality is added to the code-base for the bot, it can be simply ordered to update itself to a new version.

What personal, financial and other data are they currently interested in? The following for starters:

- Your name and address.
- Your telephone number(s)
- Your Social Security Number (or the equivalent used outside the US)
- Web site login data, such as Banks, Building Society, eBay, PayPal, ISP, etc.
- Credit-card numbers and .CVV or CVV2[20]
- Bank account details; account number and sort code
- CD Keys [Software Registration Keys/Numbers]
- Software registration keys
- Passport details
- Driving license details
- Identity card details

So, really it is not so much Privacy at stake here, although that is a concern, what the malware authors and those renting the botnets are really up to in this area is identity theft and fraud. In many ways this area is very similar to what the 419 [Advance Fee Fraud] and Phishing scams are trying to do. However, many bots are using keylogging, packet sniffing and are searching your hard drive for the data and handing it over without your knowledge, consent and without any assistance from you.

---

[18] Source: http://www.anml.iu.edu/ddos/types.html
[19] You can read the whole story here: http://www.grc.com/dos/grcdos.htm
[20] More details on CVV and CVV2 can be found here:
http://support.worldpay.com/kb/product_guides/worldaccess/help/security_code_verification.html

Identity Theft definition:

Identity theft results from the theft of key personal information. The perpetrator uses the information to create identification and credit cards. Identity theft often results in months of turmoil for the victim. Many people find that they need to restore credit ratings and to be freed from liability for illegal purchases.[21]

The following statistics about identity theft show a worrying trend[22]:

- According to 2 studies done in July 2003 (Gartner Research and Harris Interactive), approximately 7 million people became victims of identity theft in the prior 12 months. That equals 19,178 per day, 799 per hour, 13.3 per minute.
- The incidence of victimization increased 11-20% between 2001-2002 and 80% between 2002 -2003 (Harris Interactive). This same study found that 91% of respondents do not see an "end to the tunnel" and expect a heavy increase in victimization. 49% also stated that they do not feel they know how to adequately protect themselves from this crime.
- The Federal Trade Commission (FTC) reports that 27.3 million Americans were victimized by identity theft in the past five years, costing consumers $5 billion and businesses nearly $48 billion in 2002 alone.

So, as you can clearly see, the malware authors and those renting the botnets from them are in this for the money, not the fame or the intellectual challenge as has been used as justification for writing malicious code in the past. They have grown up from being the electronic equivalent of vandals and graffiti artists, and have become thieves, nothing more, nothing less. The really worrying part is that many malware authors are in the pay of organised crime syndicates and it can only be a matter of time before we see our first millionaire malware author.

3.3 Intellectual Capital

In many ways this is very similar to the issues with privacy and identity theft; except the data is not of a personal nature, it is sensitive or valuable information that can be sold to information brokers, or maybe even blackmail the company by threatening to send the information to a competitor unless they, the cyber-extortionists, are paid. They may even threaten to implicate a particular employee as being in league with them, either to get money or more data [intellectual property] from them or the company they work for.

Let us say that you work for a military, financial, medical or indeed any organisation or institution that has intellectual property that would be worth something to another party, let us say a competitor in your area of business or expertise. How much damage would it cause you personally or your company/institution if that data was copied? Millions, Billions?

Many bots have the ability to search for data, open backdoors to allow full access to the file system of the infected system as well as any system that it is connected to, such as file and print servers. Imagine a firewall administrator whose system is infected; imagine if the firewall rule base was remotely modified by a malicious third party.

Oh, did I mention that many bots have keylogging functionality? I did, well what about being able to turn or microphones and webcams and record you or your meetings? Well, they can.

3.4 Loss of Confidence

If a bot infected system is traced back to 'your' company [organisation or establishment], what would be the impact on 'your' companies credibility? What about a loss of confidence from other companies that you partner with, what about the potential loss of faith by your customers?

These are real concerns and they shouldn't be underestimated. In this case there is such a thing as bad publicity no matter who tells you otherwise.

---

[21] Source: http://www.cscic.state.ny.us/msisac/webcasts/05_05/info/glossary.htm
[22] Source: http://www.wholesecurity.com/threat/identity_theft.html

Just as I was finishing this paper, there was a report that over 40 Million credit card account details were stolen from CardSystems solutions. This included not only the customers name and their card number, but also the CVV of the cards too. According to reports the data was allegedly stolen using bots to infiltrate some systems on their network which were not fully patched.

According to the data released the following is a breakdown of the card types involved:

- ~22 Million Visa Card account details.
- ~13.9 Million Mastercard account details.
- The rest were Discover or American Express account details.

Luckily the data stolen did not include customer's addresses, date of birth or social security numbers, which would be required for their identity to be effectively stolen.

The point here is that not only the company 'CardSystems Solutions' have had there reputation damaged, but also the credit card companies themselves who's data was stolen will have had their reputation and brand(s) damaged. Furthermore, one report asked 'Was Microsoft to Blame?[23]'

3.5 Network Stability

Like the pain that those on the end of a DDoS attack experience, a network that contains a number of Zombie or Drone systems under the control of the cyber-criminals will suffer from reduced network bandwidth. The more systems on the network that are under the control of the bot-herders the more pain they will suffer when these very same systems are used as part of a DDoS attack.

Just like any other host that is infected by parasitic organisms, your networks will be affected, as will all the systems on your network that are Zombies. All parasites steal from their hosts, be they biologically or technologically based.

Those of you that suffered when Slammer or Nimda hit and the network problems they caused on infected networks will understand the headache that a poorly performing or swamped network can cause.

3.6 Proxy

A common feature or component installed on a zombie system is a proxy of some variety, so below you will find a basic definition of what a proxy is, and where you can find more information of specific types of proxies commonly used/installed by bots.

Proxy definition:

"A proxy server is a computer network service which allows clients to make indirect network connections to other network services. A client connects to the proxy server, then requests a connection, file, or other resource available on a different server. The proxy provides the resource, possibly by connecting to the specified server, or by serving it from a cache. In some cases, the proxy may alter the client's request or the server's response for various purposes.[24]"

Another way that proxies can be used is as a 'redirector' this will allow TCP and GRE traffic to be redirected as required on the bot infected system to wherever the bot-herder requires.

---

[23] Source: http://software.silicon.com/malware/0,3800003100,39131314,00.htm
[24] Source: http://en.wikipedia.org/wiki/Proxy_server

3.6.1 SPAM

SPAM definition:

1. A meat product sold in tins (**S**piced **P**ork **A**nd **H**am, like luncheon meat).[25]
2. Slang for **U**nsolicited **C**ommercial **E**-mail aka UCE

Use of the term "spam" was adopted as a result of the Monty Python sketch[26] in which the SPAM meat product was featured. In the Monty Python sketch, a group of Vikings sing a chorus of "spam, spam, spam…" in an increasing crescendo, drowning out other conversation. Hence, the analogy applied because UCE was drowning out normal discourse on the Internet.

One of the increasingly common uses of botnets are as conduits to push SPAM through. This way the originator of the SPAM appears to be the system under control of the bot-herder, not the real sender which is either the bot-herder or those that have rented the use of the botnet, or have stumbled upon the installed proxy server function of the bot.

3.6.2 Phishing

Phishing definition:

Phishing attacks use both social engineering and technical subterfuge to steal consumer's personal identity data and financial account credentials. Social-engineering schemes use 'spoofed' e-mails to lead consumers to counterfeit websites designed to trick recipients into divulging financial data such as credit card numbers, account usernames, passwords and social security numbers. Hijacking brand names of banks, e-retailers and credit card companies, phishers often convince recipients to respond. Technical subterfuge schemes plant crimeware onto PCs to steal credentials directly, often using Trojan keylogger spyware[27].

According to one article[28], it is claimed that botnets are responsible for nearly 70 percent of SPAM and Phishing e-mails. I suspect that this percentage is still growing.

Along with phishing, the related attack, known as Pharming[29] [basically DNS poisoning or spoofing[30]] has also been recently linked to bots and botnets.

3.6.3 Malware seeding

Malware definition:

A short name used to describe **Mal**icious Soft**ware**. This includes viruses, worms, Trojans, bots and related threats.

In the 'old-days' [1980s and early 1990s] malware took a long time to spread widely, typically months. However, once the internet and networks became ubiquitous they started to spread wide and far more quickly, typically weeks. Malware that spread via e-mail took the next step, spreading widely in days or less than a day. Then came the likes of CodeRed, Blaster and Slammer which could be widespread in hours. In Slammer's case 90 percent of vulnerable systems were infected in under 10 minutes [mainly because it used UDP instead of TCP and could in theory have fired off 30,000 scans per second on a 100Mbps network. In reality however Slammer averaged around 4,000 scans per second per infected system][31].

The almost instantaneous appearance of new mass-mailing worms in all geographic areas of the World has been blamed on the use of botnets as launch points. Imagine a botnet of 10,000 plus systems that are ordered to spam a new mass-mailer [or Trojan] out to the world, or even to infect themselves to

---

[25] Home page http://www.spam.com
[26] Sketch script can be found here http://w3.informatik.gu.se/~dixi/spam.htm
[27] Source: http://antiphishing.org/
[28] Source: Information Security - March 2005 page 30.
[29] An article discussing the use of bots and botnets for 'Pharming' can be found here: http://informationweek.smallbizpipeline.com/security/162600131
[30] More details can be found here: http://isp.webopedia.com/TERM/P/pharming.html
[31] Source: http://www.cs.berkeley.edu/~nweaver/sapphire/

effectively kick-start the infection.

For example, the Witty worm was reported to have been launched from a small bot net of around 4,200 zombies. This allowed it to virtually appear almost instantaneously all over the world at the same time and to start searching for new victims to infect/attack.

It has been widely suspected that many of the recent most successful mass-mailing worms have used botnets to enable faster initial world-wide distribution, effectively giving the worm a head start. These include: MyDoom, Netsky and Bagle amongst others.

3.7 Web Server

The current way this is being used is to host phishing web-sites on a bot infected system. This 'zombie' system's details are then inserted into the phishing scam e-mails which are spammed out via the same infected system or via one or more other zombie systems on the same botnet.

The honeynet project 'Know your Enemy: Tracking Botnets[32]' paper had this to say about bot infected systems running a web server:

"*These same bots can also host multiple fake websites pretending to be Ebay, PayPal, or a bank, and harvest personal information. Just as quickly as one of these fake sites is shut down, another one can pop up*."

Before this the nearest we had seen to a bot running a website was the 'MigMaf[33]' Trojan, which although at first sight appeared to be using compromised systems to host web sites on, was in fact acting as a reverse-proxy. Furthermore, it also acts as a socks proxy server on port 81 allowing SPAM to be 'bounced' through it.

3.8 Mule

We are not talking about four legged creatures that are half horse and half donkey....think more of drug couriers who are more usually referred to as Mules!

Now, in most cases Mules are those that either carry things for others [hence the use of the term] or act as laundering points, such as in organized crime syndicates, they do the dirty work of moving material from A to B [or storing material] and usually have little or no idea that what they are doing is illegal. They may even be acting as a Mule under duress, such as blackmail, etc. In fact in the case of a system infected by a bot, they may not even be aware they are a mule at all!

One of the more disturbing ways that bot infected system may be used, are as a 'Mule', this means that they can be used to store illegal or stolen material, without the knowledge or consent of the real owner of the system that the bot owner now 'owns'.

3.9 Other Tricks

Bots may also use other protocols and techniques than IRC for command and control. For instance, the Slapper[34] worm used a Peer-to-Peer technique rather than IRC. The other interesting thing about Slapper is that it was a Linux and Apache worm rather than a Windows one.

We have also seen a number of bots being merged with rootkits[35]. This allows them to hide very effectively from security software installed on the infected system.

Many bots have used known vulnerabilities to gain access to unpatched systems for some time. What is happening now is that newer vulnerabilities are being added to bots far faster than we've seen in the past. The more vulnerabilities a bot 'exploits' the valuable it is to the botnet owner.

---

[32] Available here: http://www.honeynet.org/papers/bots/
[33] Full details can be found here: http://www.lurhq.com/migmaf.html
[34] Details on Slapper can be found here: http://www.f-secure.com/v-descs/slapper.shtml
[35] More data can be found here: http://www.f-secure.com/weblog/archives/archive-052005.html#00000559

Bots are not just being used to install Browser Helper Objects [BHOs] and advertising Addons [Adware] as well as being used to manipulate online polls, advertising click through links/banners and abusing Google AdSense.

4 How they work

In this section of the paper we will discuss how botnets work and how they communicate. Further examples of commands used to control botnets can be found in Appendix A.

Most modern bots are controlled via IRC. IRC servers by default use Port 6667. However, you should also be aware that IRC Servers also usually listen on several other ports by default including 6660, 6661, 6662, 6663, 6664, 6665, 6666, 6668, 6669 and 7000. These other ports are often used so that the more commonly known Port 6667 is not shown in Netstat as a remote port that the computer is connected to. Basically this is security by obscurity. Many IRC servers used by bot masters are modified and may run on almost any port.

The picture below (figure 2.)shows a typical botnet and the steps a bot infected system will go through and the type of orders it may well be asked to carry out.
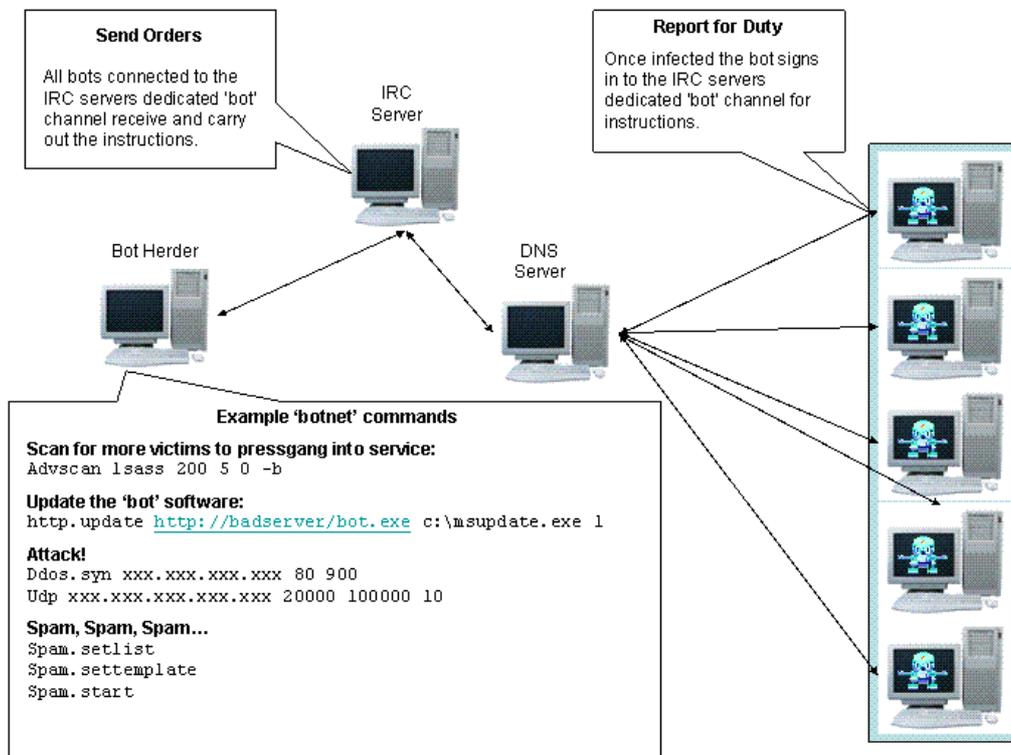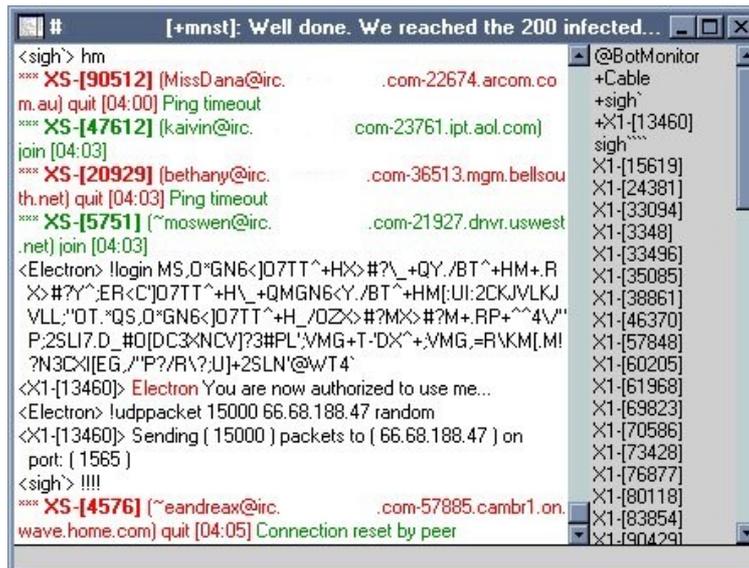

Figure 2 – Botnet Overview

When a bot signs in for duty, it does so [in the vast majority of cases] to an IRC server which is running a specific channel [room] for the bots and bot masters to log in to. Typically these 'bot' channels will hidden as much as possible to stop the IRC server owner/admin [where a public IRC server is used] from finding the botnet channel and killing it. To do this the bot master will almost certainly use the following modes for the channel at the very minimum:

- +s (Mark the channel as secret so that it cannot be seen in channels list)
- +u (Hide the userlist)
- +m (Make the channel moderated. So that a user cannot send text to that channel unless they have operator @ access or +v voice)
- +k (Make the channel password protected. This stops anyone entering the channel unless they know the correct key)

Botnets usually use dynamic DNS names from any of the providers that offer free dynamic DNS services. These when configured are setup with a very short TTL [Time-to-live], so if the botnet's current IRC server gets disconnected the botnet is headless [command and control disabled] only for a short while until a new IRC server is specified or the original comes back on-line on a new IP address.
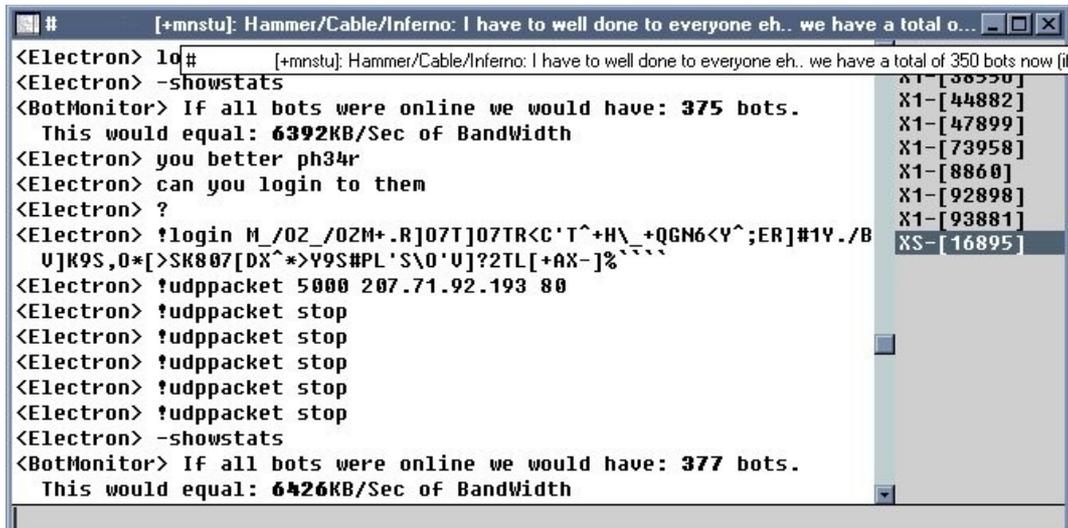
Other IRC commands may well, in the case of private IRC servers, have been removed or booby-trapped to try and discourage anyone who finds the server, or to warn the botnet owner that their server has been found.

So, once the 'zombie' system signs on for duty to the IRC control channel, it will almost certainly receive some instructions, these may well be to firstly try and find other 'victims' to press-gang into service as partof the botnet it has joined.

Figures 2a and 2b – Typical IRC botnet command and control traffic[36].

---

[36] Source: http://swatit.org/bots/gallery.html, used with permission.

In figure 3, you can see a number of ways that a bot can get installed on a new victim system.

Infection/Propagation Methods

Vulnerabilities, e.g. RPC, DCOM, LSASS, MSSQL

Dictionary attack, Open Windows Shares [SMB]

Existing Backdoor, Bagle, Mydoom, etc.

Download from website via dropper [e-mail or Instant Messaging]

Peer 2 Peer File Sharing.

E-mail, Mytob, Bagle, Mitgleider, etc.

Update Methods

Update or install new components from website or ftp server
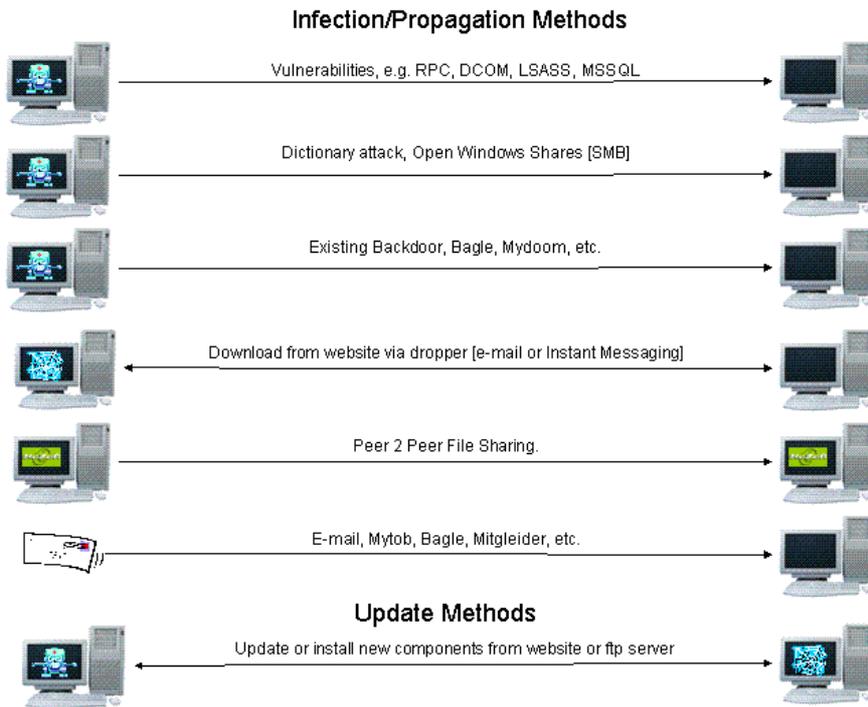
Figure 3 – Bot Spreading and Updating Methods

Other than scanning for new victims to infect, the zombie may be requested to update the bot executable or install new components, as it may have been originally infected by an old version.

The final picture in this section (figure 4) shows the typical tasks a modern botnet will be asked to carry out.

Botnet

DDoS
Victim

Spam, 419,
or Phishing
Victim

DNS   IRC
Server   Server

Bot Herder

Malware
or
Dropper

Scan and 'Sploit
Victim

Figure 4 – Typical uses for modern botnets

Any bot infected system can become the master command and control IRC server. This makes it quite difficult to 'behead' a botnet, as in reality it can 're-grow' a new head almost at will.

- 13 -

5 Solutions

We have seen how bots and botnets work, what damage they can do and what the cost can be, both financially and in some ways more importantly, damage to you/your company's reputation, brand image or credibility. Let us now look at different ways to combat them using methods which range from simple security methodologies through to technical solutions. The solutions discussed below have an effectiveness range which vary from simply blocking command and control or otherwise disrupting/disabling the botnet and detecting know bots, to proactively finding new bot variants; which are unknown to the anti-virus community at the time of capture.

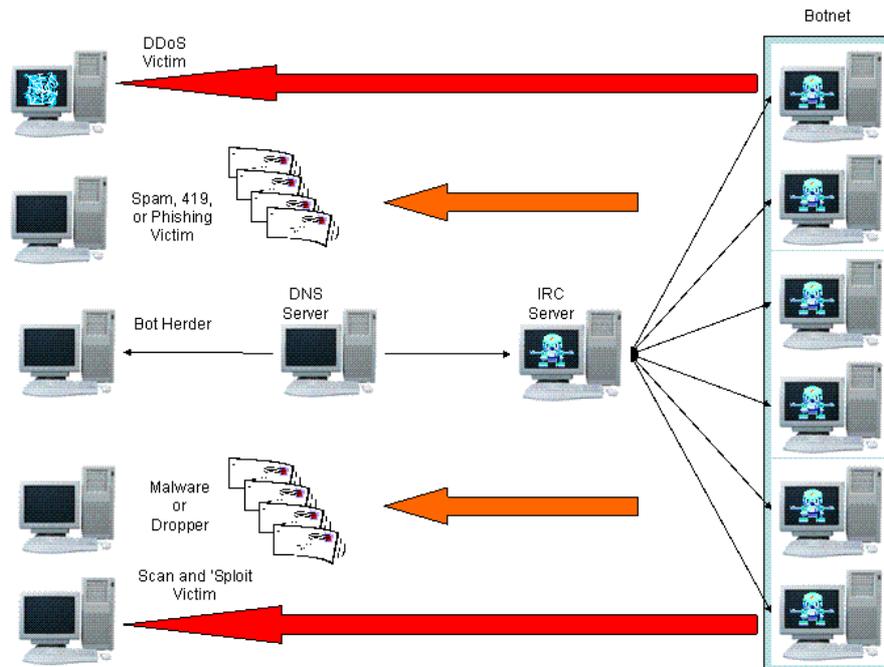## 5.1 Generic

What do I mean by generic? Simply this; techniques and/or methodologies which are not specific to bots and botnets. These may also be applied to other malware and other security issues.

5.1.1 Policies and Procedures

Policies and procedures are the foundation of your company's security stance, it will also show how seriously you take security, or not as the case may be.

Just like foundations for a building, unless they are of a good quality; built on firm principles [or ground] then they will fail, crumble or sink without trace. Leaving you or your company exposed to the elements; be they physical or technological.

The human element of security is the hardest to address, due to the general lack of interest in security which most end-users display, even to the extremes of openly flouting the rules and ignoring the security policies and procedures in place, much to the chagrin and disgust of the security staff that created them to protect their companies systems and networks[37].

Policies should not contain product names or detailed solutions, these should only be placed in procedure and technical solutions documents. All security policies and procedures [and related documents] should be reviewed at least once a year. This will enable new threats to be discussed and addressed and the relevant documentation updated to reflect this.

5.1.2 Passwords

As many bots now have the ability to perform dictionary attacks to try and gain access to your system the need for good quality passwords, or even better pass-phrases is an absolute must. A dictionary attack is where a 'list' of passwords is tried, one after another until the list is completed, or access is gained to the system being attacked.

I'm currently not aware of any bots that carry out brute forcing of passwords and/or pass-phrases, however I do expect this to happen before too much longer. A brute-force password attack would try every combination of numbers and letters [and maybe other non-standard ASCII characters too] until it gets access or runs out of combinations.

There are stand-alone tools out there that will perform these attacks, such as John the Ripper and Lophtcrack on both Windows and *NIX systems.

We may also see bots trying to steal password hashes instead of the plain-text password, as a number of tools exist that have pre-hashed [computed] thousand or millions of passwords/phrases and then it is just a matter of comparing the stolen hash against them until a match is found. These types of attacks are generally referred to as Rainbow tables[38]. As an example of the power and speed of this approach using a set of Rainbow Tables of 64GB which covers the following characters `[ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#$%^&*()-_+=~`[]{}|\:;"'<>,.?/]` and keyspace [7555858447479 possibilities which equates to 2^42.8] will break any 14 character password in a few minutes! [Based on at least a fast Pentium 4 based PC]

---

[37] Source: You are the Weakest Link, Goodbye! – Malware Social Engineering Comes of Age – Martin Overton - Virus Bulletin March 2002 pp 14-17

[38] More details on Rainbow Tables can be found here: http://www.antsight.com/zsl/rainbowcrack/

Now, imagine a bot that steals the LMHashes from Windows systems or Unix MD5 or SHA1 hashes and passes them off to such as system.

5.1.3   Education

I commented about the 'human problem' [aka Wetware] in a Virus Bulletin article back in 2002 which stated that "*the overall view of most end-users that security is an IT issue, and therefore not their problem. They seem to think that the technology will save them, what they really need to understand is that they are part of the problem, and are currently exacerbating it.*"

Education is important, but for most staff a simple security policy and acceptable use policy will be more effective than trying to educate them about all the types of risks out there on the internet. Instead focus on your support and technical staff, as they will probably be more interested and likely to retain the knowledge for a longer period. They may even end up by educating the end-users they visit and rub off some of their knowledge onto them; a bit like 'pollination' but without the mess.

5.1.4   Are you IRCed?

Do you allow your staff to use IRC clients? Is there a business need to use IRC?

If you really don't have a business case or need to run IRC then ensure that IRC traffic is blocked and that you include details in your 'security policy' or 'acceptable use policy' that IRC is not allowed.

If you use IRC on you own internal network, then block IRC at your border routers/firewalls so that any bots can't 'phone-home' to the IRC server they have been instructed to connect to.

This simple step will cripple most botnets. Especially if you use a 'deny all' policy on your perimeter routers and/or firewalls. All allowed protocols should be proxied where feasible as this adds another layer of protection to your network and makes a bot creator's job that much harder.

5.1.5   IM out to get You!

Do you allow your staff to use instant messaging clients? Is there a business need to use such a facility?

If you really don't have a business case or need to run instant messaging then ensure that this is clearly mentioned in your security policy and acceptable use policy. If possible ensure that instant messaging traffic is blocked.

This simple step will cripple most malware which uses instant messaging as one of its infection vectors.

However, if you do need to use instant messaging then use a product such as Lotus SameTime which uses a proprietary protocol and not one that uses the MSN, Yahoo or AOL one as this will help to minimise the risk of it being used by malware authors as an infection vector.

Why am I mentioning Instant Messaging in a paper about Bots and Botnets? Well guess what, there are a number of Mytob variants that can propagate via IM as one of their infection vectors.

5.1.6   Vulnerability Scanning

Do you have an in-house vulnerability scanning or ethical hacking team? If you do then use them to scan your internal servers, desktops and laptops too as this can be useful for finding un-patched systems, or those with unusual ports open or listening so they can be remediated.

A good indicator that a Windows system is a 'zombie' [or is using IRC] is that you will find TCP port 113 [Ident] open. On a clean system this is one of the few ports that is not wide-open on Windows by default.

Suitable tools for vulnerability scanning include:

- Nessus [http://www.nessus.org/]
- InternetScanner – ISS [http://www.iss.net/]
- Nmap [http://www.insecure.org/nmap/]

- SAINT [http://www.saintcorporation.com/]
- Microsoft Baseline Security Analyzer (MBSA)

5.2 Tools and Technologies

So, we have now covered a number of 'generics', let us move swiftly on to some of the tools and technologies that can be leveraged in the bot and botnet wars.

5.2.1 Perimeter and Network Firewalls

To help minimise the chances of infected systems 'phoning-home' once successfully infected by a bot you should ensure that you operate a 'deny-all' policy on your firewalls; both at the perimeter and also on other firewalls used to partition your network. As mentioned earlier in this paper, if your company/institution does not allow IRC then ensure that this traffic cannot traverse your firewalls and network by using suitable filtering. For IRC start by ensuring that the default range of ports is not open, these being: 6600 – 7000/TCP.

The same goes for all other network aware applications that need [or want] to connect to the internet or across your own network, use a denial all firewall setup. Only open up ports that need to be open for internet access. This will help not just in tackling bots but malicious software in general.

Firewall logs [and DNS, Proxy, SMTP, etc.] should be reviewed regularly to ensure that any bot and botnet traffic can be analysed, infected systems remediated and further defences can be considered or existing ones fortified by tightening configurations, etc.

5.2.2 Application Firewalls (Proxies)

Where possible proxy all traffic destined for the Internet, this includes IRC, HTTP, FTP and any other protocol or application that can be setup to use a proxy server. All traffic for these protocols that do not use the proxies should be blocked.

Be aware that there are ways to make any application [even a bot] able to use a proxy, these include using Netcat, SocksCap, and HTTP-Tunnel.

If you do use a proxy, ensure that it is secured and you enable logging so that you can review the logs to look for any IRC traffic which has passed through the proxy server.

5.2.3 DNS

Setup local DNS records for known botnet control sites, so that the command and control for these botnets are disabled. This is commonly called "nullrouting" or a "sink hole", because the DNS entries direct the offending domain or subdomains to an inaccessible IP address. Below you will see examples of IRC botnet names[39] that can be neutralised in this way:

- `bleh.darkacidonline.us`
- `blackcarder.net`
- `pod2004.dyndns.dk`
- `metalhead2005.info`
- `d66.myleftnut.info`
- `m3t4lh34d.info`
- `diablo.corsforcors.com`
- `all.evilpacket.org`
- `18.xxor.biz`
- `hellbot.magic-guy.org`
- `hellbot.nasrat.net`
- `hellmagicbot.no-ip.org`
- `nasrat.org`

---

[39] The ones listed here are mainly used by the many Mytob variants.

If you take this route you must ensure that you set the TTL for the record to a sufficiently long period of time, say 2592000 [30 days] or longer.

Here is an example DNS record for one of the entries listed above:

Zone entry in named.conf

<pre>Zone "blackcarder.net" {
      Type master;
      File "blackcarder.zone";
};</pre>

Contents of blackcarder.zone file

<pre>$TTL 2592000
@   IN SOA blackcarder.net. root (
                                46
                                3H
                                15M
                                1W
                                1D)
IN   NS   your.dns.server.name
IN   A   10.109.37.123</pre>

You can easily setup zone files for any domain that you want to effectively block access to, botnets or not. As with any filtering/blocking technique care must be taken to ensure that you don't block access to 'business' resources.

5.2.4 SMTP

Ensure that only your 'official' SMTP servers are allowed to route mail to the internet, all other SMTP traffic that does not use the 'official' SMTP servers should be logged and/or dropped as it is almost certainly the result of malware, either trying to spread itself or sending SPAM, Phishing or Scam e-mails.

Setup attachment filters [both for inbound and outbound traffic], if you don't already have them in place, to block e-mails that can be used as malcode infection vectors. These include file extension such as:

ade	Access Project Extension
adp	Access Project file
bas	BASIC program
bat	DOS batch file script
chm	Compiled HTML file
cmd	1st Reader External Command Menu
com	Command file (program)
cpl	Control Panel Module
crt	Certificate file
eml	Outlook Express message
exe	Executable file (program)
hlp	Windows help file
hta	HTML file
inf	Package information file
ins	Install script
js	Javascript
lnk	Shortcut file (Windows)
mdb	Access database
mde	Access file
msc	Common console document (Windows 2000)

msi	Installer program
msp	Windows Installer patch file
mst	Windows Installer transform
pif	Program information file (Win 3.1)
rar	RAR compressed file format
reg	Registration file
scr	Screen saver
sct	FoxPro forms
shs	Shell scrap file
url	Internet shortcut file (Universal Resource Locator)
vbs	Visual Basic program
vbe	Visual Basic related
wsh	Windows Shell
zip	ZIP file

The above list should not be seen as being complete; there are almost certainly a number of other extensions/file types that should be blocked and a number of those on the list have caveats associated with their use.

5.2.5   Patch Management

Over the last few years we have seen the window between a vulnerability being announced and malware exploiting it shrink from years to months and now weeks and days. So, this area needs to be addressed in the fight against bots and botnets as many bots use known [which have patches available] vulnerabilities to gain access to vulnerable systems.

At the very least you should ensure that all Windows systems are set to automatically check the Windows Update website at least once a week. If your systems run Windows 2000, 2003 or XP make sure you enable the Windows update service via Automatic Updates . This will ensure that updates are automatically downloaded and installed on those systems.

If you prefer to control when windows updates are deployed across your network then you can use the Microsoft Software Update Server [SUS].

Here is some data on SUS from the Microsoft site:

SUS is a version of Windows Update designed for organizations that want to approve each software update before installing them. SUS allows administrators to quickly and easily deploy Windows–related security updates and critical updates to any computer running Windows 2000, Windows XP Professional, or Windows Server 2003 systems. SUS includes the following capabilities:

- Software updates can be approved on each SUS server, enabling testing in a separate environment as well as phased deployments across an enterprise.
- SUS clients, which are the same as the Automatic Update component described earlier, can be configured to download software updates from the SUS server (saving bandwidth on shared Internet connections), or directly from Windows Update.
- Software updates can also be copied onto a CD-ROM from an SUS server connected to the Internet, and then transferred to SUS server in a protected network no Internet access.

SUS servers require Windows 2000 Server or Windows Server 2003, IIS, and port 80 communications with SUS clients. SUS servers can be configured to synchronize software update packages and approvals either manually or automatically from a parent SUS server (or from Windows Update),
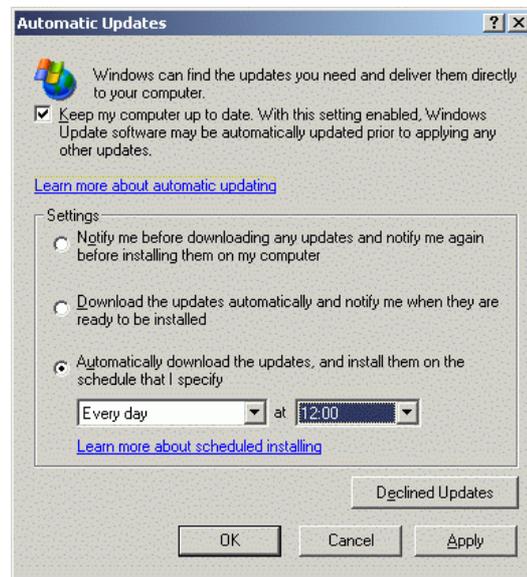
enabling flexibility in how the environment is maintained.

Below are links to articles covering other solutions:

- http://www.networkworld.com/reviews/2003/0303patchrev.html
- http://www.serverwatch.com/tutorials/article.php/3414841
- http://www.serverwatch.com/tutorials/article.php/3381211
- http://www.serverwatch.com/tutorials/article.php/3424551

5.2.6   IDS and IPS

IDS Definition: "A system that tries to identify attempts to hack or break into a computer system or to misuse it. IDSs may monitor packets passing over the network, monitor system files, monitor log files, or set up deception systems that attempt to trap hackers[40]".

IDS comes in two main flavours; NIDS [Network based Intrusion Detection Systems] and HIDS [Host based Intrusion Detection Systems] and they both have a place in the fight against bots and botnets. Then there is the offspring of IDS, known as IPS.

Back in 2003 the Gartner Group, caused something of a stir by pronouncing that Intrusion Detection Systems (IDS) and their Intrusion Prevention Systems (IPS) offspring were a market failure -- and in fact will be obsolete by the middle of the decade.

The problem is not with IDS and IPS technologies; the problem is managing these tools and technologies and the massive amount of data they produce. Too many companies treated IDS and IPS as they did Anti-Virus; they installed them and left them to update themselves. Very few took the time to look at what the IDS/IPS was finding; fine-tuning them to get the best out of them or even [more worryingly] doing anything about the alerts that were being generated.

Host based Intrusion Detection Systems:
Most HIDS do one or more of the following to detect that a system may have been compromised:

1. Integrity checking
2. System Log monitoring
3. Policy driven behaviour blocking
4. Kernel wrapping
5. Buffer overflow detection

Network based Intrusion Detection Systems:

NIDS Definition[41]: Monitors all network traffic passing on the segment where the agent is installed, reacting to any anomaly or signature based activity. Basically this is a packet sniffer with attitude. They analyse every packet for suspected nefarious activity, most will also look for anomalies within the protocol.

There are many NIDS products on the market, probably the best known are:

- Snort
- RealSecure

What is SNORT
For the uninitiated, SNORT is a lightweight Network IDS [NIDS] which works on Windows and *NIX systems and is free (apart from the hardware and manpower costs), very flexible and widely used and respected.

Snort contains a number of rules/signatures that can be used to identify bot traffic. This will allow botnets found on your own networks to be easily identified and the infected systems remediated to

---

[40] http://myphliputil.pearsoncmg.com/student/bp_hoffer_moderndbmgmt_6/glossary.html
[41] Source: http://www.networkintrusion.co.uk/ids.htm Also has other useful definitions, such as IPS, HIDS, etc.

remove the threat.

Here is an example of a Snort bot rule/signature:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "BLEEDING-EDGE
VIRUS Agobot/Phatbot Infection Successful"; flow: established;
dsize: 40; content:"221 Goodbye, have a good infection |3a 29 2e
0d 0a|"; reference: url,www.lurhq.com/phatbot.html; classtype:
trojan-activity; sid: 2000014; rev:2; )
```

As you can see this signature is for one of the Agobot variants and was created by Joe Stewart from Lurhq.

I have also created rules/signatures for Snort which can be used to identify systems that have been sent the Bot as a binary, or even, in the case of Mytob as it arrives via e-mail. There are also a number of other signatures/rules for Snort which I have created to detect the bot 'binary' travelling across the network being monitored by Snort.

Furthermore you could simply use an IDS to look for bot IRC traffic such as: " 332 ", " TOPIC ", " PRIVMSG " or " NOTICE " strings. Any such signatures/rules would have to be carefully crafted to minimise the possibilities of false-positives.

Intrusion Prevention Systems:

IPS Definition: An intrusion prevention system (a computer security term) is any device which exercises access control to protect computers from exploitation. "Intrusion prevention" technology is considered by some to be an extension of intrusion detection (IDS) technology, but it is actually another form of access control, like an application layer firewall.

Intrusion prevention systems were invented by vendors who decided to make access control decisions based on application content, rather than IP address or ports as traditional firewalls had done. This ability to inspect network traffic at a deeper level confused them with intrusion detection systems, which also inspect network traffic for signs of intrusions.[42]

Intrusion prevention systems may also act at the host level to deny potentially malicious activity.

According to some researchers, IDS is dead[43] and has been replaced by IPS [Intrusion Prevention Systems]. Examples of IPS products include: IntruShield from McAfee, Proventia from Internet Security Systems and Attack Mitigator from Top Layer. Just like with IDS there are both Network and Host based solutions available.

The beauty of IPS is that it can stop malicious traffic it recognises in its tracks, thereby stopping an infected system infecting others on the network.

5.2.7   Anti-Virus
The use of anti-virus technologies as a detection method for bot infected systems is self-evident, as many of the bots are detected by anti-virus products. This is why we are seeing the inclusion of techniques in many of the modern bots to allow them to disable as many security and anti-virus products as possible. In some cases this functionality may well be the first to be deployed, as a dropper being spammed out. Once run the dropper lowers or neutralises any local defences and then opens up the backdoor, or just downloads more components as required to complete the infiltration.

The thing to remember with anti-virus tools is that they can only [normally] detect malware they know about. New malware variants may well be detected by heuristics; however they are still far from perfect.

---

[42] Source: http://en.wikipedia.org/wiki/Intrusion_prevention_system
[43] Source: http://www.esecurityplanet.com/views/article.php/2228631

5.2.8   Anti-Rootkit Tools

Rootkits have been around for *NIX systems for many years, however they are now a growing problem for Windows systems.

What is a rootkit?
A rootkit is a collection of tools an intruder brings along to a victim computer after gaining initial access, usually via hacking into the box manually or by getting the a user to execute a Trojan or Worm which will install a backdoor for them to slither onto the system in the first place. A rootkit generally contains network sniffers, log-cleaning scripts, and trojaned replacements of core system utilities. There is however another type which does not tend to replace system files, these are: Kernel [LKM] rootkits which subvert the system by attaching themselves to, or by otherwise modifying the kernel of the targeted operating system.

Some examples of such kernel rootkits on Linux include: Knark, Adore, and Rtkit.

Although *NIX rootkits have been around for many years and are generally considered the major threat to *NIX security, there are also a growing number of Windows rootkits. This 'rootkit' scenario is a complete about-face when compared to other classes of malware, where DOS/Windows is the most targeted and *NIX is little more than a drop in the malware ocean.

Some examples of Wintel rootkits include: Hacker Defender, FU and Vanquish.

Why do you need to consider rootkit detection tools? Well, a number of bots are starting to include rootkit techniques[44] to allow them to hide from the OS and many security tools as they bind in directly to the kernel. A number of bots have used a recompiled version of the FU rootkit driver to remove their process entry from Windows Task Manager, others have used the JiurlPortHide driver for hiding network connections. It does look like we will be seeing increasingly sophisticated and 'invisible' bots as rootkit technologies and techniques get added to the codebase of the major bot families.

There are a number of tools available that claim to be able to detect and remove rootkits, these are listed below, along with the OS that they are suitable for:

- ChkRootkit [*NIX - http://chkrootkit.org/]
- Rootkit Hunter [*NIX - http://www.rootkit.nl/projects/rootkit_hunter.html]
- RootkitRevealer [Wintel - http://www.sysinternals.com/ntw2k/freeware/rootkitreveal.shtml]
- UnHackme [Wintel - http://greatis.com/unhackme/]
- Blacklight [Wintel - http://www.f-secure.com/blacklight/]

5.2.9   Personal Firewalls

These can be used to block unwanted applications from being able to connect to the network, effectively, in the case of a bot, stopping it from connecting to the command and control network. This means that the bot can't join the botnet, it won't get the orders that the bot-herder is issuing and therefore the risks are reduced.

5.2.10 Anti-DDoS Products

All of the following vendors offer products/services which can be used to filter [drop] DDoS traffic on your network perimeter. This is achieved by dropping traffic based on source IP addresses and protocols. However, these types of solutions can still be defeated by large botnets [1,000+ in size] or by botnets that generate genuine requests rather than just firing off lots of UDP/TCP packets which are all the same.

Many of these products/services work by looking for anomalous traffic, they achieve this by monitoring individual or aggregate traffic flows.

Network level defences (used to detect and filter/stop floods)
- Arbor Networks [http://www.arbornetworks.com/]

---

[44] More details can be found here: http://www.f-secure.com/weblog/archives/archive-052005.html#00000559

- CS3 [http://www.cs3-inc.com/]
- Captus Networks [http://www.captusnetworks.com/]
- Cisco Systems [http://www.cisco.com
- Lanscope [http://www.lancope.com/]
- Mazu Networks [http://www.mazunetworks.com/]
- Riverhead Networks [http://www.riverhead.com/]
- Reactive Network Solutions [http://www.reactivenetwork.com/]
- Top Layer [http://www.toplayer.com/]
- IntruShield [http://www.mcafee.com/]

Host level defences (detect, stop handler/agent installation)
- Entercept [http://www.mcafee.com/]
- Tripwire [http://www.tripwire.com/]
- AIDE [http://sourceforge.net/projects/aide]

Other products/services aim to keep your network running by using packet shaping or quality of service [QoS], effectively dividing your network into separate mini data-pipes and setting thresholds for a specific protocol. This enables other protocol pipes [such as SMTP] to keep running even if one or more [such as HTTP and FTP] are saturated.

5.2.11 Anti-DDoS Strategies
Akamai

According to the Akamai site:

"Akamai is the global leader in distributed computing solutions and services, helping organizations grow their online businesses without growing their IT infrastructures. The company created the world's largest and most widely used on-demand distributed computing platform, with more than 14,000 servers in 1,100 networks in 65+ countries."

One service offered by Akamai specifically mentions anti-DDoS , this is the **EdgeSuite for Business Continuity** managed service, it claims:

"Unmatched Performance - DDoS attacks affect the performance of the entire Internet by creating congestion and failed routers. EdgeSuite for Business Continuity bypasses bottlenecks to deliver content and applications with greater speed and reliability."[45]

Distributed services such as those offered by Akamai can help a company ride out a concerted DDoS attack, however Akamai itself was the target of a DDoS attack on June the 15th 2004[46], instead of attacking web servers, Akamai's DNS servers were targeted. This apparently resulted in sites such as Yahoo, Google, Microsoft and Apple being almost unreachable for some time. The attack is suspected to have been carried out by a Russian hacker who had allegedly offered to "pull any website, say Microsoft" for a minimum cost of $80,000. This offer was made a mere four days before the attack took place.

The 'Quick and Dirty' Option
If the attack is a simple persistent request for the same web page on a site then a suitable short-term solution is to replace the affected page, which may be content rich, with a low-bandwidth version. It is going to be less of an issue serving 100,000 requests for a page under 1KB in size than a page [and all it's graphics, flash, scripting, etc.] that may well be between 20 and 150KB normally.

ISPs
You should check to see if your ISP has any experience in handling and mitigating DDoS attacks. This could include the following:

1. Network address flexibility; such as being able to switch address blocks to thwart attacks.

---

[45] Source: http://www.akamai.com/en/html/about/press/press319.html
[46] Source: http://loosewire.typepad.com/blog/2004/06/behind_the_akam.html

2. Incident Response
3. Ability to preserve connectivity to 'key' network segments.
4. Traffic capture and analysis; to trace attacks
5. Traffic filtering
6. IDS and/or IPS
7. Firewalls

Final Thoughts

I think that this quote which was made in regard to the Akamai attack sums the problem of DDoS attacks up very neatly:

"*On any day there are in excess of a million compromised systems just waiting to be used in DDoS like this, any business is vulnerable, and there is no 100% protection.*" - ***Johannes Ullrich, CTO - SANS Institute's Internet Storm Center***.

Other suggestions on solutions as well as data on the problem of DDoS attacks can be found on the following web pages:

- http://www.cert.org/tech_tips/denial_of_service.html
- http://www.sans.org/dosstep/roadmap.php
- http://staff.washington.edu/dittrich/misc/ddos/

5.2.12 Industry Initiatives

While I was carrying out my research for this paper, I found out that there are a number of initiatives to share data and combat DDoS attacks and botnets.

I am not going to cover them all here, but this one seems the most promising, so deserves some verbiage:

Fingerprint Sharing Alliance
A community for coordinated, rapid attack resolution.

The Fingerprint Sharing Alliance is a coalition of telecommunications companies around the globe that are stamping out cyber attacks that cross company boundaries, continents and oceans.

Distributed denial-of-service (DDoS) attacks, worms, and other cyber attacks can paralyze even the most well structured network for days, costing millions of dollars in lost sales, freezing online services and crippling a company's reputation. Hacker-controlled bot nets can be used to attack a Web site or network on command, requiring little effort to knock a company off-line. According to a recent report published in Cisco Packet Magazine, more than 30,000 computers are "recruited" into botnets everyday…[47]

Companies involved in this initiative include:

- Asia Netcom
- British Telecom
- Broadwing
- Cisco Systems
- Earthlink
- Energis
- Internet2
- ITC^DeltaCom
- MCI
- Merit Network
- NTT Communications

---

[47] Source: http://www.arbornetworks.com/fingerprint-sharing-alliance.php

- University of Pennsylvania
- The Planet
- Rackspace Managed Hosting
- Utah Educational Networks
- Verizon Dominicana
- WilTel Communications
- XO Communications

5.2.13 SMB-Lure and WormCharmer

A quick introduction to SMB-Lure for those who are not aware of it:

"The SMB-Lure is a network security sensor that remotely detects computers infected with file-share worms on a corporate network. After entering a corporate network, worms such as Funlove, Elkern, Klez, Nimda, Sircam, and Qaz exploit shared folders as a stealthy means of infecting more computers. The SMB-Lure actively attracts file-share worms to itself so that it can detect and identify the infected computers.

The SMB-Lure has been successfully used to detect hundreds of infected computers on both corporate and educational networks.

The SMB-Lure is built using a Samba file server specially configured to appear as a large number of computers in the Windows Network Neighbourhood. These virtual computers are positioned in strategic places, in the spread pattern of file-share worms, within the Network Neighbourhood.

The Samba server is configured to run in debug mode to provide extensive logging of each worm visit. The Samba file-share is baited as a honey-pot, containing a variety of interesting files and directories for the worms to interact with.[48]"

A quick introduction to WormCharmer for those who are not aware of it:

WormCharmer builds on the original design from John Morris, but instead of using the SMB-Lure on an internal network, mine was pointed at the Internet. Furthermore, I added the ability to capture samples, and automatically process known samples and archive them by year, month and name [malware name, e.g. W32.Sdbot.GBM, W32.Mytob.FL, WORM_AGOBOT.AUV, etc.]

Other modules of the WormCharmer[49] system create near-real-time statistics and write these out to a web page on my personal website[50].

You could consider this [WormCharmer] to be an early-warning system, in some ways similar to Intrusion Detection Systems which are placed in-strategic areas of your network . This will pick up threats earlier, and may give you and your company more time to prepare your internal/gateway defences against these new threats.

Both SMB-Lure and WormCharmer [and other similar systems] are ideal for gauging penetration of new bots and other malware on an internal network. In the event of an outbreak; be it a new or known bot [or other malware] variant, it will allow you to identify 'problem' areas/systems with greater speed than just relying on anti-virus tools; which may be unable to detect the new variant at that time, or relying on end-users to report the problem.

In the time that I have been using WormCharmer I have captured numerous new bots trying to spread. Many of these are so new that at the time of capture none of the anti-virus companies [and products] could detect them.

The real beauty of SMB-Lure and WormCharmer is that they are not signature based and therefore catch numerous new variants that would otherwise be missed by conventional malware detection tools.

---

[48] Taken from John Morris's web page on SMB-Lure.
[49] Full details on SMB-Lure and WormCharmer can be found in my VB2003 paper, entitled: Worm Charming: Taking SMB-Lure to the Next Level
[50] htpp://arachnid.homeip.net

6 Conclusions

Bots have grown from simple tools to automate tasks on IRC to a major threat to the Internet and the companies and institutions that use the internet for commercial, educational and scientific benefit. This threat also attacks the home users, denying them access to site they want to use, or using their own systems to perform these attacks; steal their data, their identities, use their systems to store stolen or illegal material, to route spam, malware or scams through…There is no doubt that we will see other uses for bot infected systems in the near future.

DDoS attacks are a growing threat to businesses. Many cyber-extortionist price their 'protection money' rates under the cost of deploying anti-DDoS solutions, this is not a coincidence. Businesses that are threatened with these attacks should never pay, as otherwise they:

1. Will get repeat 'protection money' requests, and they will become more frequent.
2. They play into the hands of the cyber-criminal, which shows them that this is an easy way to make 'good' money.
3. Increase the overall threat to other businesses.

According to recent research from Forrester, one in three large businesses has been the victim of a successful DDoS attack, with more than forty percent of those attacked facing losses or more than £54,000.

We seem to be currently witnessing the birth of a new threat; the super-bot. This is a multi-component and multi-stage creation. As we have seen from the recent Bagle and Mytob variants, these are usually spammed out as a small downloader Trojan, which disables anti-virus software, personal firewalls, other security software and tools. They often go onto modifying the HOSTS file, to stop an infected host from getting assistance from the vendors.

We will see bots using other command and control networks other than IRC. There have been a few other techniques used so far, such as P2P. I suspect that we will start to see encrypted command and control in the not too distant future and may even see bots that are 'proxy' aware and communicate via port 80 [HTTP], 53 [DNS] or other ports that are known to be open on firewalls to bypass security solutions.

The war for control of your PC has started and so far the 'enemy' has infiltrated and captured many systems; using them as bases for attack, storage, mis-information, mis-direction, theft and spying… ..are you going to let them win?

This year is the two-hundredth anniversary of the Battle of Trafalgar. Let us take and modify the communication that Admiral Lord Nelson gave before the great sea battle against the forces of Napoleon which cost him his life, and use it as a security mantra; "<insert company or institution name here> expects that every person will do their duty[51]"….in other words, if you are not part of the solution, there is a very good chance that you are part of the problem. "Prepare for battle."

7 Thanks and Feedback

I would like to thank Dr. Igor Muttik of NAI for his help in supplying some of the Bot family statistics. Thank also go to all AVIEN (and AVIEWS) members for their support and members of the IBM MSSD and IBM Virus Emergency Response Teams.

All constructive feedback on this paper will be warmly received.

---

[51] The original communication was "England confides that every man will do his duty", however the word confides would have had to be spelt out letter by letter, so the flag for the word 'expects' was used instead. This changed the communication to: "England expects that every man will do his duty".

Further Reading/Resources

- Know Your Enemy: Tracking Botnets - http://www.honeynet.org/papers/bots/
- Profiles - Automated Credit Card Fraud - http://www.honeynet.org/papers/profiles/cc-fraud.pdf
- Know Your Enemy: Phishing - http://www.honeynet.org/papers/phishing/
- Defeating DDOS Attacks   - http://www.cisco.com/en/US/products/ps5888/products_white_paper0900aecd8011e927.shtml
- Rise of zombie PCs 'threatens UK' - http://news.bbc.co.uk/1/hi/technology/4369891.stm
- Phishers use zombie nets to automate attacks - http://www.vnunet.com/2126242
- Phishing Activity Trends Report - October, 2004 - http://www.antiphishing.org/APWG_Phishing_Activity_Report-Oct2004.pdf
- Worm Charming: Taking SMB Lure to the Next Level - Proceedings of the 13th International Virus Bulletin Conference 2003
- Invasion of the "Bots" You Could Be A "Zombie" and Don't Know It!! - http://www.cscic.state.ny.us/msisac/webcasts/05_05/info/5_18_05presenter.htm
- Anti-Malware Tools: Intrusion Detection Systems – Proceedings of the 14th EICAR Conference 2005 pp286 – 307
- Malware in a Pig Pen – Part 2 – Virus Bulletin October 2004 pp 11-13
- Canning More Than SPAM with Bayesian Filtering - Proceedings of the 14th International Virus Bulletin Conference 2004
- Malware in a Pig Pen – Part 1 – Virus Bulletin October 2004 pp 11-13
- Botnets and Botherds - http://www.sfbay-infragard.org/SUMMER2004/Botnets_Botherds-1.pdf
- Agobot & the Kit-chen Sink - http://www.infectionvectors.com/vectors/Agobot_&_the_Kit-chen_Sink.pdf
- Denial of Service on the Internet - http://vayner.net/dos/dos.html
- Distributed Denial of Service Attacks - http://newsite.prolexic.com/downloads/whitepapers/Prolexic_WhitePaper-DDoS-Q4-2004.pdf

8   Appendix A

Phatbot has quite an extensive command list which is mainly derived from Agobot.

Command	Description
bot.command	runs a command with system()
bot.unsecure	enable shares / enable dcom
bot.secure	delete shares / disable dcom
bot.flushdns	flushes the bots dns cache
bot.quit	quits the bot
bot.longuptime	If uptime > 7 days then bot will respond
bot.sysinfo	displays the system info
bot.status	gives status
bot.rndnick	makes the bot generate a new random nick
bot.removeallbut	removes the bot if id does not match
bot.remove	removes the bot
bot.open	opens a file (whatever)
bot.nick	changes the nickname of the bot
bot.id	displays the id of the current code
bot.execute	makes the bot execute a .exe
bot.dns	resolves ip/hostname by dns
bot.die	terminates the bot
bot.about	displays the info the author wants you to see
shell.disable	Disable shell handler
shell.enable	Enable shell handler
shell.handler	FallBack handler for shell
commands.list	Lists all available commands
plugin.unload	unloads a plugin (not supported yet)
plugin.load	loads a plugin
cvar.saveconfig	saves config to a file
cvar.loadconfig	loads config from a file
cvar.set	sets the content of a cvar
cvar.get	gets the content of a cvar
cvar.list	prints a list of all cvars
inst.svcdel	deletes a service from scm
inst.svcadd	adds a service to scm
inst.asdel	deletes an autostart entry
inst.asadd	adds an autostart entry
logic.ifuptime	exec command if uptime is bigger than specified
mac.login	logs the user in
mac.logout	logs the user out
ftp.update	executes a file from a ftp url
ftp.execute	updates the bot from a ftp url
ftp.download	downloads a file from ftp
http.visit	visits an url with a specified referrer
http.update	executes a file from a http url
http.execute	updates the bot from a http url
http.download	downloads a file from http
rsl.logoff	logs the user off
rsl.shutdown	shuts the computer down
rsl.reboot	reboots the computer
pctrl.kill	kills a process
pctrl.list	lists all processes
scan.stop	signal stop to child threads
scan.start	signal start to child threads
scan.disable	disables a scanner module
scan.enable	enables a scanner module

Command	Description
scan.clearnetranges	clears all netranges registered with the scanner
scan.resetnetranges	resets netranges to the localhost
scan.listnetranges	lists all netranges registered with the scanner
scan.delnetrange	deletes a netrange from the scanner
scan.addnetrange	adds a netrange to the scanner
ddos.phatwonk	starts phatwonk flood
ddos.phaticmp	starts phaticmp flood
ddos.phatsyn	starts phatsyn flood
ddos.stop	stops all floods
ddos.httpflood	starts a HTTP flood
ddos.synflood	starts an SYN flood
ddos.udpflood	starts a UDP flood
redirect.stop	stops all redirects running
redirect.socks	starts a socks4 proxy
redirect.https	starts a https proxy
redirect.http	starts a http proxy
redirect.gre	starts a gre redirect
redirect.tcp	starts a tcp port redirect
harvest.aol	makes the bot get aol data
harvest.cdkeys	makes the bot get a list of cdkeys
harvest.emailshttp	makes the bot get a list of emails via http
harvest.emails	makes the bot get a list of emails
waste.server	changes the server the bot connects to
waste.reconnect	reconnects to the server
waste.raw	sends a raw message to the waste server
waste.quit	
waste.privmsg	sends a privmsg
waste.part	makes the bot part a channel
waste.netinfo	prints netinfo
waste.mode	lets the bot perform a mode change
waste.join	makes the bot join a channel
waste.gethost	prints netinfo when host matches
waste.getedu	prints netinfo when the bot is .edu
waste.action	lets the bot perform an action
waste.disconnect	disconnects the bot from waste