Common Denial of Service Attacks

David Slee

July 10, 2007

Abstract

This paper will examine various denial of service attacks and network defense measures taken against them. A historical look at the evolution of these attacks from different flood attacks to zombie driven botnet attacks will shed light on their increasingly more sophisticated design and the networking community efforts to combat them. The current technology, motivation and future trends of distributed denial of service botnet attacks will be also be presented.

Introduction

Denial of service attacks come in two types: Denial of Service attacks (DoS) and Distributed Denial of Service attacks (DDoS). A DoS attack is "an attack in which a third party purposely floods a network or website with traffic in order to prevent legitimate access (“Denial of Service”, 2007)". A DDoS "occurs when multiple compromised systems flood the bandwidth or resources of a targeted system, usually one or more web servers (“Denial-of-service attack”, 2007)." In both cases, system vulnerabilities, hardware and/or software, are exploited to allow an intruder to compromise a system. Today, DDoS attacks carried out by “botnets”, take advantage of multiple compromised personal computers, or “zombies”, to direct a coordinated attack on a target network. Early DoS attacks are well known and can be defended against by robust networking equipment and proper security practices. DDoS botnet attacks of today present a more difficult challenge for network administrators. The perpetrators of botnet attacks have found it to be a very lucrative practice and are constantly evolving their methods as new vulnerabilities arise. A very large botnet can overwhelm the best of defenses. The motivation for botnet attacks vary from extortion to corporate warfare to nationalistic pride. The creators of botnets can be very organized and treat their endeavor as serious business. Fortunately, it appears the awareness of the information security community has reached critical mass in the last few years as many botnet detection and prevention tools have begun to appear in the market from the likes of Google, Tumbleweed and Cisco.

DoS Attacks

The most common type of DoS attack occurs when a network is flooded with data sent by an attacker. For example, a web server sent more requests than it can handle at once becomes overloaded. Once it is overwhelmed, it cannot process ordinary requests from anyone that might type a URL for a web site it is hosting. In this way, a user is denied service because he cannot access that site (McDowell, 2004). Any network device is fair game for an attack. Web, electronic mail and Domain Name System (DNS) servers and routing devices are all targets with known vulnerabilities for attackers to exploit. A DoS attack can be perpetrated in a number of ways. There are three basic types of attacks (“Denial-of-service attack”, 2007) :

1. consumption of computational resources, such as bandwidth, disk space, or CPU time;
2. disruption of configuration information, such as information;
3. disruption of physical network components.

A DoS attack may include execution of “malware” intended to:

- max out the CPU’s usage, preventing any work from occurring;
- trigger errors in the microcode of the machine;
- trigger errors in the sequencing of instructions, so as to force the computer into an unstable state or lock-up;
- exploits errors in the operating system to cause resource starvation and/or thrashing, i.e. to use up all available facilities so no real work can be accomplished;

- crash the operating system itself

Malware is software used by a hacker designed to gain access, not purposefully permitted by a user, to a computer and instruct the computer to perform a task for the hacker. Trojans, spyware, adware, keyloggers, dialers, rootkits, botnets, crimeware, badware, viruses and worms are all types of malware. The purposes for writing malware include financial gain, espionage, revenge, anger, recognition or just to see how fast it might spread (“What is Malware?”, 2007).

A Transmission Control Protocol (TCP) SYN flood is a type of DoS that falls under the category of consumption of computational resources. In a TCP SYN flood attack, a machine is overloaded with TCP connection requests coming in faster than it can respond to them. “SYN” refers to a TCP header synchronization flag (Whitman & Mattord, 2004). Each SYN packet contains a random, or “spoofed”, source IP address. The SYN requests a new connection to the target machine from the phony IP address. A machine responds to the bogus IP address and waits for a response for a few minutes. The target’s connection table is consumed with requests that are never acknowledged. New connection requests from friendly users are ignored and denied access to the machine. Fortunately, SYN floods usually do not bring down server machines and a server will most likely return to normal once the attack stops. Although they are still vulnerable, improved operating systems do make it harder for SYN attacks to flood their connection tables (“SYN Flood”, 2007).

Defensive techniques include micro blocks, SYN cookies, RST cookies and stack tweaking. Using micro blocks takes up less memory by not creating complete connection objects which is what causes a memory failure. A SYN cookie cryptographic value is included in a SYN response to the sender. If the sender is for real, the acknowledgement from the sender will include the cryptographic value. A connection is only established, and memory allocated, after a proper acknowledgement. An RST cookie, or packet, should be returned to a machine by a sender, when a faulty acknowledgement from a machine is sent in response to the original SYN packet. If no RST is returned, then the sender's connection request is dropped and table resources are never used. Stack tweaking involves dropping random SYN packets and reducing the timeout of a connection request. Dropping random packets should cause the same sender to resend packets after a period of time. Shortening the timeout for a connection request frees up memory faster, mitigating the effect of a SYN attack ("SYN Flood", 2007). Decreasing the memory allocated for each connection request and checking the incoming route against the outgoing route of a new sender may offer the best solution to attempted SYN flooding (Whitman & Mattord, 2004).

A SYN flood can be used in a Man-in-the-middle TCP hijacking attack. In 1994, renowned hacker Kevin Mitnick used SYN flooding to prevent a system from transmitting. He then used TCP hijacking to assume the identity of the flooded system (Cotter, 2002). A TCP hijacking attack targets an existing connection between two machines and convinces the victim machine that a man-in-the-middle machine is the target machine. "*Sniffing* is the use of a network interface to receive data not intended for the machine in which the interface resides…Sniffers and hijacking software are the basic tools used to mount hijacking attacks (* de Vivo, de Vivo, & Isern, 1998)." It is possible to use software like HUNT to perform sniffing, MAC address collection, ARP spoofing and DNS spoofing to mount a successful TCP hijack, ("Hunt, TCP Hijacking

Tool", 2002). ARP stands for the Address Resolution Protocol responsible for translating IP addresses into MAC addresses. ARP creates a cached table of addresses. This table is updated whenever a system receives a response, whether it was requested or not. An attacker can tell system 'A' that it is system 'B' and begin receiving system B's data ("Guide to ARP Spoofing", 2004). A DNS server is responsible for translating host names to IP addresses. DNS spoofing involves impersonating a valid DNS server by sending invalid DNS responses to a client requesting a host name, or domain name, resolution. The Windows XP DNS resolver is known to have this vulnerability (* Green, 2005). It should be noted that hijacking tools like Hunt can be legitimately used for testing system vulnerabilities.

TCP hijacking attacks can be well defended by applying the latest network system security patches, implementing intrusion detection systems and properly configuring firewalls. Using encrypted connections is a best practice in mitigating TCP hijacking attacks. Even if a connection is hijacked, an attacker will not be able to make much use the encrypted connection packet data (Whitman & Mattord, 2004).

ICMP attacks include a ping of death, ping floods, and a smurf attack. A ping of death attack is very easy to administer. All that needs to be done is to issue an ICMP Echo Request, a ping command, specifying an IP packet size greater than 65536 bytes e.g. ping –l 65510 <your ip address>. Vulnerable systems will crash when attacked with this command. An IP packet greater than 65535 bytes is illegal, but is fragmented into smaller parts when sent to a victim. The victim machine will attempt to reconstruct the fragments and create a buffer overflow eventually causing some systems to come to a halt. Windows 95 and Windows NT used to allow this and were vulnerable to ping of death attacks themselves. Any protocol sending an IP datagram can exploit this problem.

TCP, UDP, NFS, Telnet or HTTP are all potential vehicles for delivering malformed IP datagrams and any open listening port may be a target (Kenney, 1997). Today most operating systems do not allow the execution of a ping command with an IP packet beyond the maximum size limit and most networking devices have been patched to minimize this threat (Whitman & Mattord, 2004).

A ping flood swamps a victim with ping packets consuming all of the victim’s combined inbound and outbound bandwidth. It is a simple attack to administer e.g. ping –f . An attacker must have more bandwidth than the victim for the attack to succeed. Outbound bandwidth is exhausted by ping replies. Two possible firewall solutions to ping floods are 1) to only filter large ping packets and 2) delay the passing of ping request packets (“Ping Flood”, 2007).

DDoS Attacks

A smurf attack is a variety of DDoS attack called an amplification attack. Network traffic is amplified through compromised systems before it reaches a victim computer. A smurf attack accomplishes this by flooding a victim computer with ICMP echo and reply messages sent to one or more unprotected intermediary network broadcast addresses. The ping messages contain the spoofed IP source address of the victim’s machine and computers in the broadcast address domain will receive and reply to the ICMP echo messages. All of the replies, which may be amplified by more than one broadcast domain and scores of machines, will be sent to the victim’s machine exhausting its bandwidth and bringing it to a halt (* Kumar, Azad, Gomez & Valdez, 2006).

To defend against smurf attacks, all routers in a network must be configured to drop broadcast address ICMP echo requests. It is not good enough to configure a firewall router alone (Chau, 2004).

A UDP flood or "fraggle" attack is another type of DDoS amplification attack. It relies on the UDP chargen and UDP echo services. These services are usually not used. Connecting to a UDP port running the chargen service produces a constant stream of data. A UDP port running the echo service simply bounces the packet back to the sender's source address. Creating a UDP connection between chargen and echo ports will create a massive amount of traffic between two machines. If a UDP packet is sent with a spoofed source broadcast address to victim's chargen port, multiple connections might be created between echo ports on machines in the spoofed broadcast domain and the victim machine amplifying the amount of traffic targeting the victim (Wooding, 1998).

These are the recommended steps to protect a network from UDP flooding fraggle attack ("UDP Port Denial-of-Service Attack", 1996):

1) Disable and filter chargen and echo services.

2) Disable and filter other unused UDP services.

3) If you must provide external access to some UDP services, consider using a proxy mechanism to protect that service from misuse.

4) Monitor your network.

5) Take steps against IP spoofing.

Botnets

When it comes to DDoS attacks, the current scourge of the Internet are DDoS attacks carried out by botnets. These networks are comprised of malware infected zombie personal computers and can number in the millions (Keizer, 2005). Types of botnet attack occur when a “zombie or bot source opens multiple TCP connections, and sometimes issues repetitive HTTP requests…[and]…low-rate zombie/botnet attacks, which are similar to bandwidth consumption attacks except that each attack source sends multiple requests at a low rate (“DDoS Mitigation Services Based on Cisco Systems Clean Pipes”, 2007) ”. There a many types of malware that exploit operating system and browser vulnerabilities to turn a personal computer into a zombie.

Early DDoS attacks were done manually. “The attacker scanned remote machines for vulnerabilities, broke into them and installed the attack code, and then commanded the onset of the attack (* Mirkovic, Martin & Reiher, 2005).” This process has evolved into something quite a bit more sophisticated. An example of this is Mpack, “a malware distribution and attack kit…sold by a Russian gang…and…comes complete with a collection of exploit modules to be used out of the box (Lau, 2007)”. The largest botnet known to date consisting of about 1.5 million zombies was also linked to the “Russian Internet mafia” and was built to compromise and pillage online banking accounts. In this case, a Trojan horse called “Wayphisher” was created by a Dutch group of programmers (the leader was only 19 yrs old) and sold to the Russian mafia (Keizer, 2005).

Typically botnets number in size from 10 to 100 zombies. A malicious botnet attack is carried out by the following steps (* Geer, 2005):

1) a personal computer is turned into a zombie by an e-mail attachment (with attached Trojan for example), infected web site or some other avenue

2) the zombie pc will typically connect to a rogue IRC command and control server

3) the server will then send instructions and commands to the zombie

4) the zombie will awaken as a component in a DDoS attack, a e-mail spammer or keystroke logger/password stealer

Another type of botnet will use peer-to-peer software to facilitate communication between zombies. No central server is used which makes it much harder to detect. One zombie can initiate an attack on its own and pass the attack command on to the next zombie which will do the same until the entire botnet has been activated. If one zombie is discovered, it can alert other zombies to start infecting new machines to maintain their numbers (* Geer, 2005).

The reasons for carrying out botnet attacks include extortion, corporate warfare, and nationalistic pride among others (including spam, pay per click abuse, keystroke logger and password theft) ("What is a Botnet ?", 2007). It is unclear how many botnets exist and how many website owners have been extorted. For example, a very profitable online gambling website located in the Caribbean or Central America might be attacked. The website owner will then be sent a payoff demand in the amount of thousands of dollars with the threat of repeating the attack. The website owner cannot afford much downtime before losing its clients to other gambling sites, so he pays off the attacker. This is an embarrassing predicament and one that is underreported by many such victims ("DDoS Mitigation Services Based on Cisco Systems Clean Pipes", 2007).

Attacks on business rivals can be another motive for botnet attacks. One such attack was launched by the owner of an athletic apparel business on the website of a competing business. In that case, the attacker was caught and sent to prison. In another case, an attack against security company Blue Security, which offered free spam prevention software and a do-not-spam registry, resulted in Blue Security being ultimately overhelmed by a very large botnet. Homes with personal computers with always on broadband connection haves accelerated the spread of botnets. The greater bandwidth of these connections allow for faster infection and more forceful DDoS attacks and spamming. The largest botnets out there of a million plus zombies can harness and direct an attack with a bandwidth of 22 to 24 Gbps. The largest of websites, perhaps even one like Google, may be vulnerable to such an attack (Berinato, 2006). Attempts at taking down the Internet itself, for boastful purposes one would presume, have occurred by attacking the Internet's root servers (McMillan, 2007). However, an attack on Google would not be in the best interest of botnet operators as automated botnet pay per click abuse is a common and profitable practice for them. Botnets can be programmed to automatically click on Google AdSense banner ads creating revenue for website publishers and raising costs for advertisers (Leyden, 2006).

Politics may also be the reason behind some botnet attacks. In Estonia, governmental websites were recently attacked because a Soviet era monument was moved to another location (Vamosi, 2007). Political party and new media websites in Russia were attacked as elections drew near ("Cyber attacks engulf Kremlin's critics", 2007). On the day of the Democratic primary in Connecticut, the website of candidate Joe Lieberman was attacked, most likely by a botnet (Sullivan, 2006).

Defensive measures against botnet attacks fall into the preventive and reactive realms. One preventive measure includes these three steps (* Freiling, Holz & Wicherski, 2005):

1) Infiltrating a remote control network

2) Analyzing the network in detail

3) Shutting down the remote control network

This approach involves setting up a vulnerable machine as bait. Once it becomes a zombie, it is possible to monitor and identify the extent of the botnet using specialized software. The remote control network can then be shutdown by containing and extinguishing the command and control server at the heart of the botnet.

Also in the area of prevention Google has made a push by offered a free anti-malware tool (Naraine, 2007) and includes warnings about potential malware websites in their search results (“Google warns users over malware sites in search results”, 2007).

Reactive steps include white botnets like the do-not-spam registry bot tool offered by Blue Security. This bot tool would automatically send Blue Security unwanted spam information that would help Blue Security identify the location of the spammer. Blue Security would then send out a do-not-spam-me message back to the spammer. The reverse flood of these messages aimed at spammers consumed resources otherwise used to send spam depriving them of income (Berinato, 2006).

Other reactive measures are offered in the form of infrastructure networking devices offered by different vendors designed to mitigate botnet and other types of DDoS attacks. Cisco offers a tool with the following characteristics (“DDoS Mitigation Services Based on Cisco Systems Clean Pipes”, 2007):

1) attack analysis, identification, and mitigation services required to block attack traffic

2) use [of] behavioral analysis and attack recognition technology to proactively detect and identify assaults. They compile detailed profiles that indicate how individual devices normally behave. When deviations are detected, the detector responds.

3) network traffic flow analysis technology for IP networks

4) gathers data to determine a traffic baseline and then compares traffic against the baseline for flagging and anomaly detection

Another vendor, Tumbleweed, offers a reactive tool with the following features ("Tumbleweed Press Releases", 2007):

1) combines a global network of over 100 million IP address, real-time updates, and pattern detection from more than 50,000 end-points

2) can correlate distributed attacks and drop connections from newly hijacked Ips

3) defense capabilities blocking directory harvest attacks, email denial of service attacks, and invalid recipients

4) often eliminates 90 percent or more of raw email traffic before it overwhelms the network

All of the above are sophisticated defensive measures taken against a difficult and evolving threat.

The Future

The trend in the size of botnets has been toward smaller ones. The likely reason is that they are harder to detect and when used with broadband connections are just as effective as larger botnets using slower connections. Moreover, these smaller botnets could serve as the foundation for a coordinated *super-botnet* attack (* Vogt & Aycock, 2006) .

Future botnets will likely make better use of encryption to avoid detection. Modifying open source malware programs to include encryption techniques is very likely (* Thigpen, 2007). Encypting sessions will make it very hard to analyze botnet behavior. For example, an encrypted attack command to a super-botnet would make detecting the owner nearly impossible.

More peer-to-peer botnet architectures are likely to appear in the future (* Grizzard, Sharma, Nunnery, Kang, & Dagon, 2007). Imagine a botnet piggybacking over a peer-to-peer network like Gnutella or BitTorrent.

Fortunately, in recent years the information security industry has finally started to take botnets seriously. Vendors like Cisco and Tumbleweed will continue to improve their products. Google is making a major effort to identify malware infected websites. Interestingly enough, the current trend actually shows a decrease in botnets and a significant increase in phishing websites (Leyden, 2007). The appearance of do-it-yourself phishing kits has made the creation of these web sites very easy for someone who might have difficulty creating and maintaining a botnet (Leyden, 2007). It is also likely that these websites are generating more revenue for their perpetrators than botnets do currently.

It is doubtful botnets will go away any time soon. A coordinated and sustained effort in the form of better consumer awareness and education efforts, improved browser design, zombie detection tools and enhanced network infrastructure defense appliances may be our best defense in mitigating a problem that may be here to stay.

References

Denial of Service. (2007). Retrieved June 29, 2007, from the Information Technology Toolbox Web site: http://security.ittoolbox.com/topics/t.asp?t=380&p=380&h1=380

Denial-of-service attack. (2007, July 6). In *Wikipedia, The Free Encyclopedia*. Retrieved, July 10, 2007, from http://en.wikipedia.org/w/index.php?title=Denial-of-service_attack&oldid=142946274

McDowell, M. (2004). Understanding Denial of Service Attacks. Retrieved June 30, 2007, US-Cert.gov Web site: http://www.us-cert.gov/cas/tips/ST04-015.html

What is Malware? (2007). Retrieved July 1, 2007, from the iS3.com Web site: http://www.is3.com/docs/pdf/learning-center/introductory/iS3-malware.pdf

Whitman, M.E. & Mattord, H.J. (2004). *Management of Information Security.* Canada: Course Technology.

SYN Flood. (2007). Retrieved July 1, 2007 from the Internet Security Systems Web site: http://www.iss.net/security_center/advice/Exploits/TCP/SYN_flood/default.htm

Cotter, S. (2002, October 17). Book Review [Review of the book *Network Intrusion Detection, An Analyst's Handbook*]. Retrieved July 2, 2007 from the Univerisity of

Michigan Web site: http://www.si.umich.edu/Classes/540/Readings/Cotter-review-Network%20Intrusion%20Detection.doc

* de Vivo, M., de Vivo, G. O., and Isern, G. (1998). Internet security attacks at the basic levels. SIGOPS Oper. Syst. Rev. 32, 2 (Apr. 1998), 4-15. Retrieved July 2, 2007, from ACM Digital Library: DOI= http://doi.acm.org/10.1145/506133.506136

Hunt, TCP Hijacking Tool. (2002, February 17). Retrieved July 2, 2007, from the Securiteam.com Web site: http://www.securiteam.com/tools/3X5QFQUNFG.html

Guide to ARP Spoofing. (2004, April 6). Retrieved July 3, 2007, from the HackintheBox.org Web site:http://www.hackinthebox.org/modules.php?op=modload&name=News&file=article&sid=12868&mode=thread&order=0&thold=0

* Green, I. (2005, January 10). DNS Spoofing by The Man In The Middle. Retrieved July 3, 2007, from the SANS.org Web site: http://www.sans.org/reading_room/whitepapers/dns/1567.php

Kenney, M. (1997, January 22). Ping of Death. Retrieved July 3, 2007 from the Insecure.org Web site: http://insecure.org/sploits/ping-o-death.html

Ping flood. (2007, June 21). In *Wikipedia*, *The Free Encyclopedia*. Retrieved, July 3, 2007, from http://en.wikipedia.org/w/index.php?title=Ping_flood&oldid=139655430

* Kumar, S.; Azad, M.; Gomez, O.; Valdez, R. (2006, February 25). Can Microsoft's Service Pack2 (SP2) Security Software Prevent SMURF Attacks?. Retrieved, July 3, 2007 from the IEEE electronic database: http://ieeexplore.ieee.org.jproxy.lib.ecu.edu/iel5/10670/33674/01602221.pdf?tp=&arnumber=1602221&isnumber=33674

Chau, H. (2004, September 17). Defense Against the DoS/DDoS Attacks on Cisco Routers. Retrieved July 5, 2007, from the SecurityDocs.com Web site: http://www.securitydocs.com/library/2553

Wooding, K. (1998, June 12). Magnification Attacks: Smurf, Fraggle, and Others. Retrieved July 5, 2007, from the Pintday.org Web site: http://pintday.org/whitepapers/dos-smurf.shtml

UDP Port Denial-of-Service Attack. (1996, February 8). Retrieved July 5, 2007, from the Cert.org Web site: http://www.cert.org/advisories/CA-1996-01.html

Keizer, G. (2005, October 21). Dutch Botnet Suspects Ran 1.5 Million Machines. Retrieved July 5, 2007, from the TechWeb.com Web site: http://www.techweb.com/wire/security/172303160

DDoS Mitigation Services Based on Cisco Systems Clean Pipes. (2007). Retrieved from the Cisco.com Web site: http://www.cisco.com/en/US/netsol/ns341/ns121/ns310/net_value_proposition0900aecd80511f1e.html

* Mirkovic, J., Martin, J., & Reiher, P. (2005, April 17).A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms. Retrieved July 6, 2007, from the Enhyper.com Web site: http://www.enhyper.com/content/ddostaxonomy.pdf

Lau, H. (2007, May 27). MPack, Packed Full of Badness. Retrieved July 6, 2007, from the Symantec.com Web site: http://www.symantec.com/enterprise/security_response/weblog/2007/05/mpack_packed_full_of_badness.html

Keizer, G. (2005, November 8). Dutch Botnet Suspects Ran 1.5 Million Machines. Retrieved July 6, 2007, from the TechWeb.com Web site: http://www.techweb.com/showArticle.jhtml;jsessionid=RTJ1GEM23ZACCQSNDLPSKH0CJUNN2JVN?articleID=173600331

* Geer, D. (2005, January). Malicious bots threaten network security. Retrieved, July 6, 2007 from the IEEE electronic database: http://ieeexplore.ieee.org/iel5/2/30112/01381249.pdf?tp=&isnumber=&arnumber=1381249

What is a Botnet ? (2007). Retrieved July 6, 2007, from the Tech-Faq.com Web site: http://www.tech-faq.com/botnet.shtml

Berinato, S. (2007, November). Attack of the Bots. Retrieved, July, 6, 2007, from the Wired.com Web site: http://www.wired.com/wired/archive/14.11/botnet.html

McMillan, R. (2007, February, 6). Hackers slow Internet root servers with attack. Retrieved, July 7, 2007 from the Computerworld.com Web site: http://www.networkworld.com/news/2007/020707-hackers-slow-internet-root-servers.html

Leyden, J. (2006, May 15). Botnet implicated in click fraud scam. Retrieved July 7, 2007, from the TheRegister.co.uk Web site: http://www.theregister.co.uk/2006/05/15/google_adword_scam/

Vamosi, R. (2007, May 29). Cyberattack in Estonia--what it really means. Retrieved July 7, 2007, from the Cnet New.com Web site: http://news.com.com/Cyberattack+in+Estonia-what+it+really+means/2008-7349_3-6186751.html?tag=st.ref.goo

Cyber attacks engulf Kremlin's critics. (2007, July 2). Retrieved July 7, 2007, from the Cnn.com Web site: http://www.cnn.com/2007/TECH/07/02/russia.cyberwar.ap/index.html

Sullivan, B. (2006, August 8). Lieberman campaign site, e-mail hacked. Retrieved July 7, 2007, from the MSNBC.com Web site: http://www.msnbc.msn.com/id/14245779/

* Freiling, F. C., Holz, T., & Wicherski, G. (2005, April). Retrieved July 7, 2007, from the Google Scholar Web site: http://66.102.1.104/scholar?hl=en&lr=&q=cache:DfSyDZdaGtkJ:ftp.informatik.rwth-aachen.de/ftp/pub/reports/2005/2005-07.ps.gz+

Naraine, R. (2007, June 11). Googler updates open-source anti-malware tool. Retrieved July 7, 2007, from the Blogs.ZDNet.com Web site: http://blogs.zdnet.com/security/?p=278

Google warns users over malware sites in search results. (2006, August 7). Retrieved from the NewsTarget.com Web site: http://www.newstarget.com/019897.html

Tumbleweed Press Releases. (2007, June 19). Tumbleweed Unveils Intelligent Data Leak Prevention and Real-Time Botnet Defense. Retrieved July 8, 2007, from the Tumbleweed.com Web site: http://www.tumbleweed.com/news/press_releases/2007/2007-06-19.html

* Vogt, R. & Aycock, J. (2006, August). Attack of the 50 Foot Botnet. Retrieved July, 8, 2007, from the University of Calgary Web site: http://pages.cpsc.ucalgary.ca/~aycock/papers/50foot.pdf

Thigpen, S. (2007). Investigating Botnets, Zombies, and IRC Security. Retrieved July 8, 2007, from the InfosecWriters.com Web site: http://www.infosecwriters.com/text_resources/pdf/InvestigatingBotnetsZombiesandIRCSecurity.pdf).

* Grizzard, J.B., Sharma, V., Nunnery, C., Kang, B.B., & Dagon, D. (2007). Peer-to-Peer Botnets: Overview and Case Study. Retrieved July 8, 2007, from the Usenix.org Web site:http://www.usenix.org/events/hotbots07/tech/full_papers/grizzard/grizzard_html/

Leyden, J. (2007, June 20). Phishermen, not zombies, causing biggest security woes. Retrieved July 8, 2007, from the ChannelRegister.co.uk Web site: http://www.channelregister.co.uk/2007/06/20/mcafee_security_trends/

Leyden, J. (2007, June 8). DIY kits dumb down phishing. Retrieved July 8, 2007, from the ChannelRegister.co.uk Web site: http://www.channelregister.co.uk/2007/06/08/phishing_kit_survey_ibm/