A Day in the Life of a Broadband Connection

Bradley S. Rubin
University of St. Thomas
Graduate Programs in Software
St. Paul, MN
651-962-5506

bsrubin@stthomas.edu

ABSTRACT
This paper characterizes unsolicited network packets arriving at a residential-grade DSL broadband Internet connection. We summarize a simplified methodology for this characterization and the characteristics of this unwelcome Internet Background Radiation recorded during two periods of over four months each, in two consecutive years.

Categories and Subject Descriptors
C.2.0 [**Computer-Communication Networks**]: General==security and protection and C.2.3 [**Computer-Communication Networks**]: Network Operations—network monitoring

General Terms
Measurement, Security.

Keywords
Internet Background Radiation (IBR), Threat Measurement

1. INTRODUCTION
Firewalls are one of the most commonly deployed security tools used by computing consumers. Computer users of all levels of expertise have been exposed to the term firewall, and most know that firewalls help keep the bad things lurking on the Internet out of their systems. However, while the actions of other security tools such as antivirus programs are apparent from the alert dialog boxes, firewalls are a silent protector. Most firewall alerting is turned off after a brief curiosity period because of the constant popup barrage. Most firewall logging is not configured to log every incoming network packet because of the sheer volume of packets knocking on the firewall door every day. Even if these packets are logged, the logs are usually only examined in detail post intrusion. Yet we have all wondered why our modem lights are blinking when there is no apparent system activity.

This paper characterizes the data obtained by sifting through the normally unseen incoming network packets arriving at a residential-grade DSL broadband Internet connection to a Linux system that is otherwise unused. The test system never initiates activity on its own. There is no DNS entry for the IP address for this system, so in theory no packets should ever find their way to this system. This test system is a type of honeypot and provides a sample of what happens in the life of a public IP address destination. In the literature, the term IBR (Internet Background Radiation) is used to capture this notion of unsolicited and unwelcome Internet noise. This paper summarizes the characteristics of the IBR recorded during two periods of over four months each, in two consecutive years.

Section 2 compares and contrasts another IBR study, Section 3 discusses the experimental setup, Section 4 shows the results of this study, Section 5 covers more details on the major traffic types observed in this study, Section 6 makes some observations on Snort and IBR, and Section 7 offers conclusions.

2. RELATED WORK
Pang, et. al.[1] contains a good list of other IBR-related studies and discusses their own methodology and results from a study of a Class A and two sub-Class B network addresses spaces over several periods of about a week long each. Their study period occurred in March-May of 2004. The huge packet rates seen in these address spaces (30,000 packets per second in the class A network) required special filtering technology to keep the data acquisition and analysis manageable. By looking across a large address space, it is possible to identify address-scanning behavior. The study, like this one, looked passively at traffic destined for the target space. In addition, the study also shows the results of active responses to initial request packets, which in many cases allows more specific identification of the packet intent.

Although the study described in this paper covers only a single IP address, the IBR traffic characterization results are very similar to the Pang study, with much lower experimental data acquisition and analysis complexity and cost. In addition, this study covers much longer contiguous periods of time two years apart. Finally, since current systems are becoming more hardened by shutting off unused services and closing unused ports with firewalls, the statistics presented in this study more closely reflect the actual IBR traffic experienced by production systems since the target experimental system only responds to all TCP and UDP requests with an indication that the target port is closed.

3. EXPERIMENTAL SETUP
A Linux-based test system collected the IBR statistics. All ports on this system were closed and responded as closed to incoming packets (i.e. with a TCP RST-ACK for TCP or an ICMP Destination Unreachable for UDP). The system was connected to a DSL modem that was configured to statically map the external public IP address to an internal private IP address, forwarding all network traffic. All filtering on the modem was disabled. The Linux system was thus completely exposed to all network packets arriving at the public IP address. There was no distinctive DNS name (i.e. stthomas.edu) associated with this IP address, which was randomly selected from the ISP dynamic IP address pool, so it did not stand out as different from any other DSL pool IP address. All network packets arriving at the system were logged with tethereal[2]. The final packet log was then analyzed for

intrusion signatures using Snort[3] with all rules enabled, parsed by tethereal and loaded into a MySQL database [4] for analysis.

The data were collected over two consecutive years. In each year, the packets were collected for periods of over four months. The first period, subsequently referred to as Run 1, covered 8/15/2004 to 1/2/2005 (141 days). The second period, subsequently referred to as Run 2, covered 9/24/2005 to 3/3/2006 (161 days).

4. Results

Table 1summarizes the major metrics from this study.

For Run 1, over the period of 141 days, 81,627 IP datagrams and ICMP packets were sent to the target system, an average of 579 packets per day. These packets came from 25,845 (183/day) distinct source addresses and targeted 2102 (15/day) distinct destination ports and distinct ICMP message types. These packets triggered 3,292 Snort alerts (23/day). Snort reported 166 port scans in Run 1 (1.2/day).

For Run 2, over the period of 161 days, 193,242 IP datagrams and ICMP packets were sent to the target system, an average of 1,200 packets per day. These packets came from 20,062 (125/day) distinct source addresses and targeted 481 (3/day) distinct destination ports and distinct ICMP message types. These packets triggered 10,931 Snort alerts (68/day). Snort reported 936 port scans in Run 2 (5.8/day).

Table 1. Results Summary

	Run 1 Total	Run 1 Daily	Run 2 Total	Run 2 Daily
Days	141		161	
Dates	8/15/2004 - 1/2/2005		9/24/2005 - 3/3/2006	
Packets	81,627	579	193,242	1200
Packet Range		278 – 1,548		295 – 1,973
Unique Source Addresses	25,845	183	20,062	125
Unique Destination Ports	2102	15	481	3
Snort Alerts	3,292	23	10,931	68
Port Scans	166	1.2	936	5.8

4.1 Traffic Volume

Does IBR take an appreciable amount of bandwidth? Run 2 had the greatest data volume per day. It consisted of 71,830,523 bytes of UDP traffic, 3,031,100 bytes of TCP traffic, and 106,927 bytes of ICMP traffic for a total of 74,968,550 bytes over the 161-day period. This yields about 5.4 bytes/second, which is a very small portion of most residential broadband capacities. This figure only characterizes the incoming traffic, and does not include any responses to that traffic. Firewalls can be configured to not give any response to a port connection attempt, as opposed to the "port closed" responses sent in this study, so this IBR bandwidth number reflects that stealthy approach.

4.2 Traffic Characteristics Over Time

Figure 1shows the number of network packets logged each day for Run 1. Although the average packets received per day was 579, the range was from 278 to 1,548 packets/day. The two most prevalent types of traffic were associated with TCP destination port 445, which is used by the Windows SMB/CIFS protocol, and TCP destination port 135, which is used by the Windows NETBIOS protocol. The next most prevalent type of traffic corresponded to Windows Messenger spam. Also, next highest on the list but not shown, are traffic associated with UDP port 137 and TCP port 139, two other Windows NETBIOS-related ports. All together, these five types of traffic accounted for 85% of the packets seen over the entire period. Interestingly, all of the five most prevalent types of traffic are associated with Windows exploits, even though the test system is Linux-based. We discuss these attack types in greater detail later in this paper.

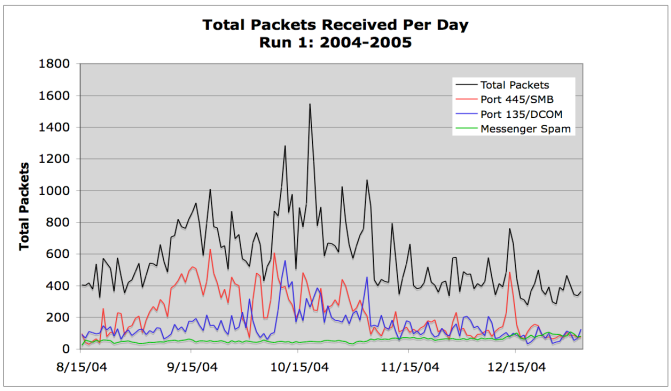

Figure 1. Run 1 Total Packets/Day and Major Types

Figure 2 shows the number of network packets logged each day for Run 2, with very different characteristics from the previous year. The number of packets seen per day ranged from 295 to 1973. Here, the dominant traffic is Windows Messenger spam. Of the 193,242 packets received, 140,386 were Windows Messenger spam (73%). This amounted to an average of 872 packets per day, up from only 58 per day during Run 1. The next most prevalent traffic type is associated with TCP port 445 (11%). Together, these two traffic types account for 84% of the packets.

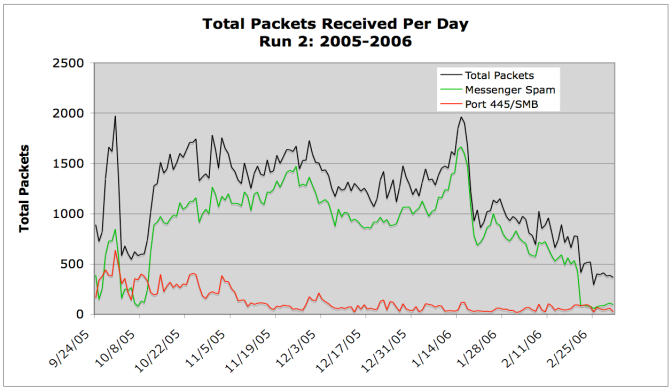

Figure 2. Run 2 Total Packets/day and Major Types

4.3 Traffic Origin

Since IP source addresses can be forged, the true source of the packets captured in this study must always be suspect. For protocols that do not require a response or a handshake from packets sent to a destination, such as the Windows Messenger spam packets, the source address can easily be forged without impacting the protocol and this can hide the true packet source. For protocols that do require a response or a handshake, the source address of the packets cannot be forged without disrupting the protocol, so these source addresses are probably reliable. However, it is possible that in this latter case the traffic source might be a zombied system with associated malicious code planted into this system by the true source. In this case, the source address would accurately reflect the system sending the traffic, but not accurately reflect the origin of the ultimate originator of the event.

A project by Beverly and Bauer[5], with results updated as of 7/24/06[6], attempts measure the prevalence of forged source addresses on the Internet. It shows that 22.7% of IP addresses are capable of being spoofed, but it does not show the actual spoofing contribution of these addresses.

In this study, we looked at the packet source without regard to possible source address forgery. The packets source addresses used as the key of a lookup table of the countries that own ranges of IP address blocks[7]. An analysis of this traffic origin shows that the majority of the traffic potentially originated from the US, with China a distant second. An analysis of the Run 2 traffic origin shows a very different picture, with China as the most prevalent potential packet source, followed by the US. The Run 2 results mostly reflect the Windows Messenger spam explosion in this period, since most of these had a Chinese source address.

4.4 Backscatter

Backscattered packets occur when a TCP packet with a forged source address arrives at its target. If the target responds with a TCP RST or RST-ACK (if the destination port is closed) or a TCP SYN-ACK (if the destination port is open), the response will be sent to the forged source address instead of the true packet sender. These packets are known as backscatter. For UDP, ICMP Destination Unreachable messages also indicate backscatter. In Run 1, 394 (0.48%) packets consisted of TCP RST, RST-ACK, SYN-ACK or ICMP Destination Unreachable packets sent to the target system. In Run 2, 1,737 (0.90%) packets were associated with backscatter. This metric can be used as a lower bound of source address forging, since only targets that respond to incoming packets and protocols that have response packets provide backscatter response packets.

4.5 Traffic Uniqueness

Another interesting question surrounds the uniqueness of the packet sources. Do sources continue to show up, day after day, or do they only appear once and never again? The latter might indicated that the packet was a random accidental occurrence. An analysis of the Run 1 data shows that 44% of source addresses only appeared once, and never again in Run 1. In fact, 92% of the traffic sources appeared in 1, 2, or 3 packets and then never again. This tends to support the position that most of the IBR comes from random, accidental, scenarios. However, not all traffic is so shy. One source appeared in 10,158 packets! An analysis of the Run 2 data shows the results from Run 2, showing a very similar 90% of traffic sources appearing in 1, 2, or 3 packets only. Once again, a persistent source appeared in 2,216 packets.

There is a related longer-term question about address uniqueness. How many of the source addresses appeared in both runs, about one year apart? Of the 45,866 unique source addresses logged in this study, only 41 (0.089%) appeared in both years. Note that the test system IP address changed between the two runs.

4.6 Traffic Targets

Table 2 lists the top ten ports (for UDP and TCP-based protocols) referenced by packets received in Run 1. Table 3 lists the same type of data for Run 2. In both cases, the data show that the IBR is aimed at a few dominant target ports and protocols.

Table 2. Run 1 Top 10 Most Common Target Ports

Protocol	Port No.	Port/Protocol Name	Number of Packets	Packet Percentage
TCP	445	microsoft-ds	31328	38.4%
TCP	135	epmap	20897	25.6%
UDP	1026-1027	NetrSendMsg	8194	10.0%
UDP	137	netbios-ns	4517	5.5%
TCP	139	netbios-ssn	4088	5.0%
TCP	80	http	1368	1.7%
TCP	1433	ms-sql-s	1186	1.5%
UDP	1434	ms-sql-m	1013	1.2%
TCP	4899	radmin-port	764	0.9%
ICMP	-	Echo Request	553	0.7%
TOTALS			**73,908**	**90.5%**

Table 3. Run 2 Top 10 Most Common Target Ports

Protocol	Port No.	Port/Protocol Name	Number of Packets	Packet Percentage
UDP	1026-1027	NetrSendMsg	140386	72.6%
TCP	445	microsoft-ds	21784	11.3%
TCP	139	netbios-ssn	9883	5.1%
TCP	135	epmap	6542	3.4%
TCP	80	http	3033	1.6%
TCP	4899	radmin-port	2144	1.1%
UDP	1434	ms-sql-m	1794	0.9%
ICMP	-	Echo Request	931	0.5%
TCP	1433	ms-sql-s	930	0.5%
UDP	137	netbios-ns	854	0.4%
TOTALS			**188,281**	**97.4%**

5. MAJOR TRAFFIC TYPES

UDP Ports 1026-1027 - Windows Messenger Spam – The Windows Messenger Service (which is different from the Windows MSN instant messaging client) is the square box that slides into view or pops up on target systems notifying users of certain events. Legitimate events may include administrators sending messages to the systems they manage, print job completion, anti-virus status, and new mail notification. This functionality is built into Windows NT, 2000, and XP and does not require any other application software. The protocol for these messages begins with a UDP request to the Windows portmapper service on port 135, which in turn knows the port where the messenger service itself listens. But since many firewalls now block UDP port 135, it is common for the messenger spam to just guess at the messenger service port (usually 1026-1028). This guessed port message blast is the version of this spam captured in this study. Since no handshake is needed, the source addresses can be forged and although the vast majority of source IP addresses are owned by China, these may have been forged to make them appear that way, hiding the true source. This spam first started to appear in 3Q2003, and Windows XP SP2 disabled the messenger service by default in response to this problem. Once again, this is a case of a Windows exploit targeting a Linux system.

UDP Port 1434 Slammer Worm – The Microsoft SQL Slammer worm[8] first appeared in January, 2003. This worm exploited the vulnerability Microsoft SQL 2000 Resolution Service Stack Overflow Vulnerability (CAN-2002-0649) first discovered in July 2002. The Slammer worm targets Microsoft SQL Server 2000 by sending a single packet to UDP port 1434, which SQL Server uses as its Resolution Service Port. This results in denial of service and arbitrary code execution consequences. This worm picks its target IP addresses randomly, and makes no attempt to forge the source address in the packet. Once again, although this worm was built to exploit a Windows vulnerability, it makes no attempt to determine the fingerprint of our Linux system target.

TCP Port 1433 – This port is often associated with the Microsoft SQL Server User Authentication Remote Buffer Overflow Vulnerability (CAN-2002-1123) first made public in August 2002. The SQLSnake worm or the Spike tool most commonly exploits this vulnerability.

TCP Port 135, UDP Port 137, TCP Port 139, and TCP Port 445 – These four related ports showed up as the top packet target in both Runs 1 and 2. Port 445 is the modern replacement for the other three, and is associated with the Microsoft Directory Service in Windows 2000, XP, and Server 2003. It is used by the SMB (Server Message Block) protocol for file and printer sharing over TCP/IP. This port is a popular target because it might lead to read and/or write access to file shares if this service is erroneously exposed to the Internet. The other three ports provide a similar service, NetBIOS over TCP/IP on Windows NT. In addition to the file share exposure, the code for these services has had numerous vulnerabilities that could be exploited by engaging these ports. There are two popular exploits. One is the Blaster worm, which exploits the DCOM RPC Vulnerability associated with TCP port 135 (CAN-2003-0352) made public in August 2003. The other is the Sasser worm, which exploits the Windows Local Security Authority Subsystem Service associated with TCP port 445 (CAN-2003-0533) made public in April 2004.

Port 4899 Radmin – This port is associated with a remote administrator service called Radmin written by Famatech[9]. It was exploited by a virus known as RAHack which exploited weak passwords and first appeared in December 2004. It is interesting to note that this port was probed in this study as early as 8/15/2004, the first date a packet was captured, so there was interest in this port well before the virus appeared. The activity tripled from Run 1 to Run 2, potentially correlating with the virus impact.

6. SNORT RESULTS

Snort picked up some, but not all, of the IBR activity. This is expected, since Snort could only deal with the very first IBR packet since no other protocol packets follow that would allow enough of a signature for identification in all cases. Snort would be unacceptably noisy if it triggered alerts for any activity on any port. It did flag the Slammer worm activity on UDP Port 1434, but did not flag the activity on TCP port 1433. It did not flag any activity on Ports 135, 137, 139, or 445. It did flag numerous port scans and pings. It flagged no other traffic. Snort is a useful, but not sufficient, tool for studying IBR.

7. CONCLUSIONS

This study provides an in-depth look into the hundreds of uninvited packet visitors a residential DSL IP address received every day for two extended periods of time over the course of two years. While this sample is not large enough to characterize the IBR of the broader Internet, it does provide some insight into the nature of the traffic that a user might see. What we see is traffic that is fairly high in volume, yet still small when compared to typical individual residential broadband capacities. Windows attacks are routinely seen on the Linux target system. Perhaps most interesting, this traffic is composed of probes and attacks from worms and scans for vulnerabilities long ago identified and supposedly remediated. How many of the world's computing systems are plugged into the Internet, but otherwise ignored while spewing packets trying to spread ancient infections?

Often in computer security, users deploy technology to thwart threats without knowing about the true nature of the threat. This study shows that the threat against residential broadband connections is real. Users must continue to deploy firewall technology and keep their software updated to close vulnerabilities. And, in doing so, they are reacting to real, quantifiable threats and not just responding to generic security fear, uncertainty, and doubt.

8. ACKNOWLEDGMENTS

Thanks to the reviewers for their helpful comments.

9. REFERENCES

[1] Pang, R., et. al., "Characteristics of Internet Background Radiation", *Internet Measurement Conference, Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, Taormina, Sicily, Italy, 2004, pp. 27-40.

[2] Ethereal: A Network Protocol Analyzer. http://www.ethereal.com

[3] Snort – the de facto standard for intrusion detection/prevention. http://www.snort.org

[4] MySQL: The World's Most Popular Open Source Database. http://dev.mysql.com

[5] Beverly, Robert and Bauer, Steven, The Spoofer Project: Inferring the Extent of Source Address Filtering on the Internet, USENIX SRUTI '05: Steps to Reducing Unwanted Traffic on the Internet Workshop, Cambridge, MA, July 7, 2005.

[6] ANA Spoofer Project, http://spoofer.csail.mit.edu.

[7] IP-to-Country.com | 'Cause every I.P. has a Home…, http://ip-to-country.webhosting.info

[8] Moore, David, et. al., "Inside the Slammer Worm", *IEEE Security and Privacy*, August, 2003.

[9] Radmin – PC Remote Control Software – fast and secure remote access from anywhere. http://www.famatech.com.