Detection and Characterization of Port Scan Attacks

Cynthia Bailey Lee
Chris Roedel
Elena Silenok
Department of Computer Science & Engineering
University of California, San Diego

{clbailey, croedel, esilenok}@cs.ucsd.edu

Abstract

Port scans represent a sizable portion of today's Internet traffic. However, there has been little research characterizing port scan activity. The goal of this project is to analyze sample network traces to discover and classify properties of port scans. We hope that this work will help to generate better network intrusion detection systems and increase general network security.

1 Introduction

The Internet today is a complex entity comprised of diverse networks, users, and resources. Most of the users are oblivious to the design of the Internet and its components and only use the services provided by their operating system or applications. However, there is a small minority of advanced users who use their knowledge to explore potential system vulnerabilities. Hackers can compromise the vulnerable hosts and can either take over their resources or use them as tools for future attacks. With so many different protocols and countless implementations of each for different platforms, the launch of an effective attack often begins with a separate process of identifying potential victims.

One of the popular methods for finding susceptible hosts is port scanning. Port scanning can be defined as "hostile Internet searches for open 'doors,' or ports, through which intruders gain access to computers." [7] This technique consist of sending a message to a port and listening for an answer. The received response indicates the port status and can be helpful in determining a host's operating system and other information relevant to launching a future attack. [4]

The goal of this project is to analyze and characterize port scanning traffic. By defining a set of heuristics and applying them to the network trace data, we were able to isolate suspicious packets and group them into sets of scans. These sets were further analyzed to extract properties of the port scanning traffic and to collect relevant statistics.

The remainder of this paper is organized as follows: Section 2 describes the port scanning activity and detection methods in literature. Section 3 explains our data gathering and classification methodology and its limitations. We outline the experimental platform in Section 4, and present the results of our experiments in Section 5. We describe the related and future work in Section 6 and summarize our findings in Section 7.

2 Background and Related Work

Port scanning is a technique for discovering hosts' weaknesses by sending port probes. Although sometimes used by system administrators for network exploration, port scanning generally refers to scans carried out by malicious users seeking out network vulnerabilities. The negative effects of port

scans are numerous and range from wasting resources, to congesting the network, to enabling future, more serious, attacks.

There is a plethora of tools that aim to determine a system's weaknesses and determine the best method for an attack. The best known and documented tool is nmap by Fyodor from www.insecure.org. [8] Nmap uses a variety of active probing techniques and changes the packet probe options to determine a host's operating system. Nmap offers its users the ability to randomize destination IPs and change the order of and timing between packets. This functionality can obscure the port scanning activity and thus fool intrusion detection systems. Other port scanners include queso, checkos, and SS. However, these tools do not provide all the capabilities of nmap and thus are not as popular.

Several port scan detection mechanisms have been developed and are commonly included as part of intrusion detection systems. However, many of the detectors are easy to evade since they use simple rules that classify a port scan as more than X distinct probes within Y seconds from a single source. Typically, the length of Y is severely limited, to keep the amount of state manageable. Spice, a tool developed at Silicon Defense, tries to avoid this drawback. [2] Spice maintains records of event likelihood, from which it generates an anomalousness score for each packet. Packets with high scores are stored longer, while state for unsuspicious packets is safely discarded. This heuristic allows Spice to detect stealthy port scans while still being operationally practical. Another approach is employed by Vern Paxon in Bro and emphasizes real time performance and notification, as well as clear separation between mechanism and policy. [9]

3 Classification Methodology

For the purposes of our analysis, we define a port scan as all anomalous messages sent from a single source during the trace period. We classify port scans into three basic types based on the pattern of target destinations and ports the scan explores.

3.1 Vertical Scans

The vertical scan is a port scan that targets several destination ports on a single host. Naively executed, this scan is among the easiest to detect because only local (single-host) detection mechanisms are required.

3.2 Horizontal Scans

A horizontal scan is a port scan that targets the same port on several hosts. Most often the attacker is aware of a particular vulnerability and wishes to find susceptible machines. One would expect to see many horizontal scans for a particular port immediately following the publicizing of a vulnerability on that port.

3.3 Block Scans

Some attackers combine vertical and horizontal scanning styles into large sweeps of the address-port space. This method can yield a hit-list for future exploitation as described in [10].

3.4 Scan Detection

One way to avoid detection is to increase the time between consecutive probes. This technique works since most intrusion detection systems look for X events in a Y-sized time window and can only keep a limited amount of state. [6] We did not have real-time constraints and thus were able to use a time window large enough to detect such stealthy scans.

An attacker can also conceal her IP address by using IP decoys, or "zombie" computers under an attacker's control. Such a scan will appear in our analysis as different scans originating from several IPs. We attempt to quantify this error by combining scans that appear coordinated. If several source IPs are seen targeting the same set of hosts and ports and these source IPs are in the same /24 network, they are classified as decoys. This is

only an approximate solution, which we use for comparison.

3.5 *Classification Rules*

To separate port scanning traffic from other traffic, we looked for probes of two or more {IP address, port number} pairs from a given source within 120 seconds. By using this heuristic we detect the majority of all scans since most port scanning tools set the time between the packets to be much less than 120 seconds. We also keep the state of the destinations by maintaining information about 5000 targets that are not part of currently known scans and keep a 60-second window between the consecutive packets.

4 Experimental Platform

The network trace data was obtained from CAIDA [5] and was gathered on a very lightly utilized /8 network. Two traces were used, each spanning about a week: February 1st-8th, 2001, and February 11th-17th, 2001. More information about the data can be found in [1]. The data was filtered to exclude all legitimate, outgoing and backscatter traffic. The remainder was mostly port scans with a small percentage of misconfigured traffic.

The analysis was conducted on the UCSD ActiveWeb machines. We used Snort to simplify scan detection and logging. Snort is a freeware traffic analyzer much like tcpdump, with the addition of preprocessors that allow for packet sorting based on a set of pre-defined rules. We then used a series of Perl scripts to further analyze the results and generate scan statistics.

5 Results

Using the previously defined rules, we observed 9927 vertical scans, 5623 horizontal scans, and 2008 block scans.

5.1 *Packet Types and Distribution*

Most of the packets were sent over TCP, with some UDP traffic. The distribution of packet types and protocols can be seen in Figure 1. As shown, most of the packets are TCP SYN packets, with ACK FIN packets a distant second, followed by UDP and TCP ACK RST. All the remaining types combined are only a minute fraction.

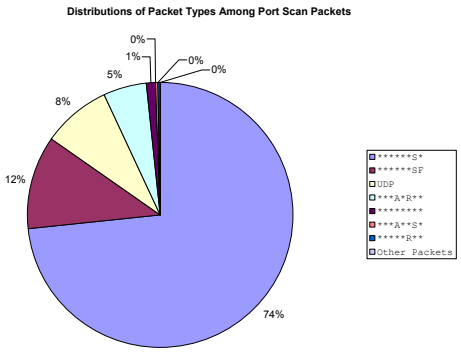

Figure 1 - Packet Types

Another property we wanted to observe was the time distribution of port scan traffic. Figure 2 does not show a correlation between traffic and the time of the day. We can thus assume that port scanning is a constant activity. This might be due to time zone differences between the attack sources.

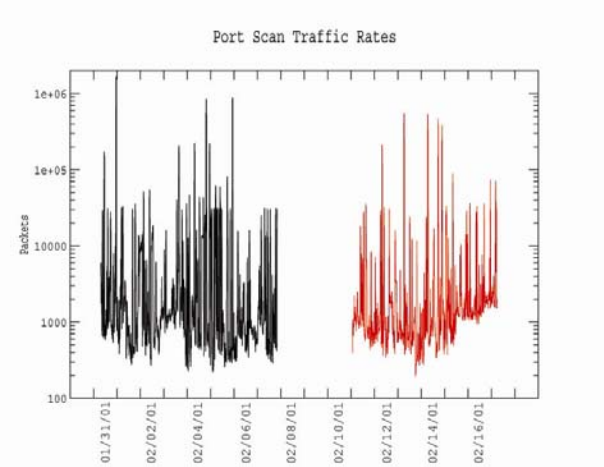

Figure 2 - Port Scan Traffic Rates

5.2 *Vertical and Horizontal Scans*

An interesting metric is a scan size, which gives an indication of the amount of information gathered by the attacker. For vertical scans, we define a scan size as the number of distinct ports scanned. For

horizontal scans, it is the number of distinct destination IPs. We observed many scans originating from the same /24 networks that exhibited the same behavior. We believed these scans were a coordinated effort. To roughly quantify the number of scans in this category, we grouped the vertical scans by the source IP if the destination IP and the scanned ports were the same and the source IPs were in the same /24 network. Thus, we assume that most of the source IPs were decoys since the scanning patterns of all of them are so similar. Figure 2 shows the distribution of the vertical scans, both before and after the grouping. We can see that there are a handful of large scans, with the size distribution being dominated by small scans.

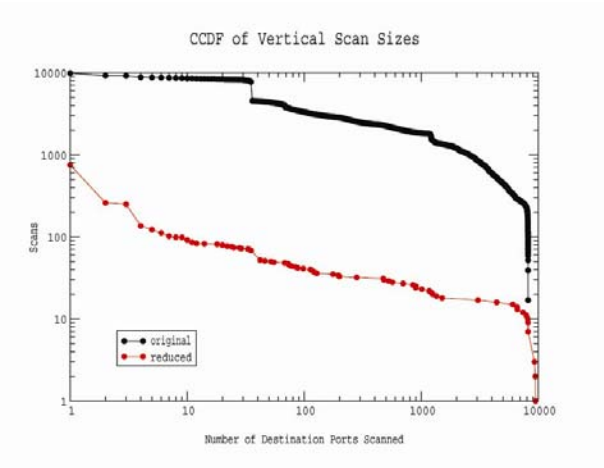

Figure 3 - Complementary Cumulative Distribution Function of the Vertical Scan Sizes

We perform the same grouping on the horizontal scans. The distribution of the horizontal scan sizes is also dominated by the small scans. However, in this case the change between the original and the reduced scan sets is hardly noticeable. It is unclear why vertical scans appear to make more widespread use of decoys; we leave this as an open question.

The typical block scan we observed examined the same 2 or 3 ports across a large set of machines. In other words, we did not observe block scans that appeared to be comprehensively covering the address-port space.

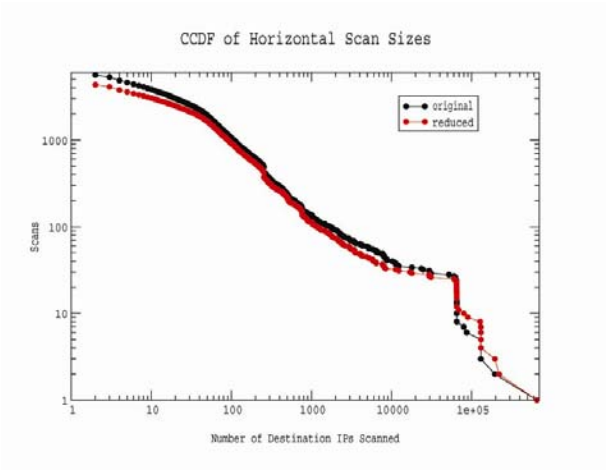

Figure 4 - Complementary Cumulative Distribution Function of Horizontal Scan Sizes

5.3 Target Ports

The popularity of target ports was another metric we evaluated. All 2^{16} port numbers were scanned at least once on some host, mostly due to the vertical scans that looked at all the ports on a host. Although some ports were scanned many more times than others, the amount of these scans as a percentage of overall probes is still negligible. Table 1 lists the target ports, the corresponding services and some related statistics.

5.4 Geographic Distribution of Scan Sources

The port scans are a global phenomena. They originate from a multitude of locations across the world and seem to be correlated only with accessibility of the Internet. To map the location of scan sources, we created a list of unique source IPs and then used CAIDA's IPGeo tool to map IPs to corresponding latitude/longitude pairs. Figure 5 shows the resulting map.

Number of Hits	Percentage of Total	Port Number	Port Services
6588	0.081%	137	NetBIOS name service (UDP)
5127	0.063%	21	FTP
5103	0.063%	25	SMTP
4960	0.061%	53	DNS
4943	0.061%	17	QOTD
4940	0.061%	113	IDENTD/AUTH
4935	0.061%	105	CSO
4934	0.061%	33	DSP
4932	0.061%	129	PWDGEN – not used for anything, so most likely a port scan
4932	0.061%	29	MSG-ICP
4931	0.061%	1	TCPMUX – test if machine is running SGI Irix
4928	0.060%	13	daytime - Not clearly specified format => used for fingerprinting machines
4928	0.060%	93	DCP
4925	0.060%	41	RAT: Deep Throat - Puts an FTP Service at Port 41
4925	0.060%	85	MIT ML Device
4924	0.060%	97	Swift Remote Virtual File Protocol
4922	0.060%	77	Private Remote Job Execution Services
4920	0.060%	73	Remote Job Services
4919	0.060%	121	Jammerkilla - Encore Expedited Remote Procedure Call
4918	0.060%	37	Time

Table 1 - Top 20: Most Actively Scanned Ports and their Functions

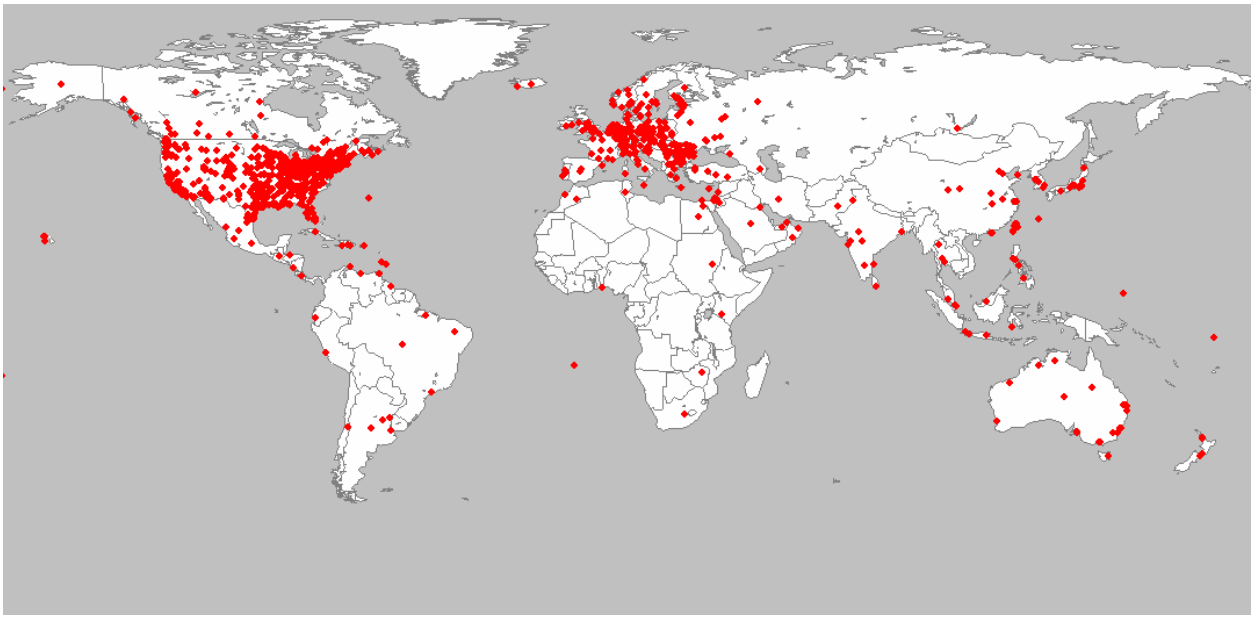

Figure 5 - Geographic Distribution of Port Scan Source IP Addresses

5.5 Scan Patterns

Attackers might try to hide their port scanning activity from naïve detection mechanisms by randomizing the order of destination IP and port probes. From our analysis we saw that most scans did not employ this strategy. Of vertical scans, 58% probed port numbers sequentially, and 91% of horizontal scans traversed the destination IPs sequentially.

Scan duration is another metric that helps us to evaluate port scans and design better intrusion detection systems. Figure 6 shows complementary cumulative distribution functions for vertical, horizontal, and block scans. We see a significant variance in the data although short scans tend to be more widespread than the long ones.

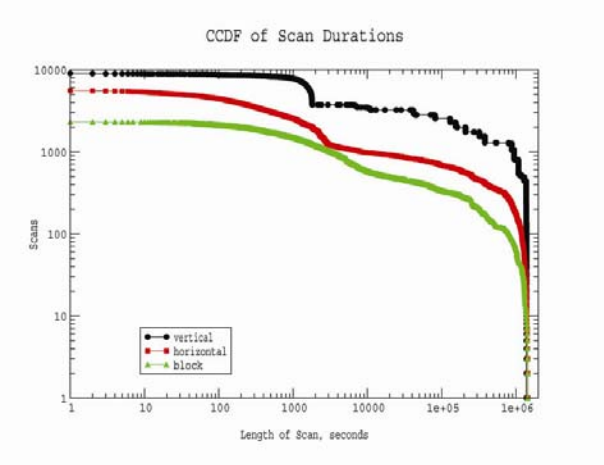

Figure 6 - Complementary Cumulative Distribution of Scan Durations

6 Related and Future Work

At the time of this writing, we are unaware of any port scanning characterization research. We serve as pioneers in this exciting field! However, much remains to be done. More advanced heuristics and classification would allow for more accurate identification of port scanning traffic. Our data set contained little or no legitimate traffic from which the port scans had to be separated. Future research could examine more difficult, and more typical, datasets. Port scan traffic could be evaluated as a percentage of overall traffic as well as categorized on the basis of its contribution to network congestion and resource consumption. Another limitation of our dataset is that we only observed port scanning where hosts or addresses on our network were the victims. It is possible, even likely, that an actively used network would see more port scan activity, as attackers would have more specific interest in those systems.

7 Conclusions

The majority of the scans were carried over TCP, with TCP SYNs dominating the traffic. UDP was another protocol that we saw, although it was not very prevalent. Most of the scans were simple vertical or horizontal scans, with vertical scans prevailing by a factor of nearly 2. All the ports were scanned at least once, although even the most frequently scanned ports did not account for a large percentage of the probes. The scan sources originated from a multitude of locations and favored densely populated areas of Europe and North America. Most of the horizontal scans were sequential whereas the vertical scans varied. We observed great diversity in the scan duration.

8 Acknowledgments

We would like to express our gratitude to a number of people who helped on this project. We want to thank the people of CAIDA and in particular David Moore for providing the traces, as well as advice and assistance. We would also like to thank David Hutches who provided amazing help with the ActiveWeb machines. We also acknowledge the ActiveWeb project for allowing us to run the scan detection software for several days, and for providing us with unlimited disk quota on their machines, which we put to abundant use. Finally we would like to thank Stefan Savage for making logistical arrangements, giving advice and guiding our investigation.

References

[1] D. Moore, G. Voelker, S. Savage, Inferring Internet Denial-of-Service Activity, USENIX Security Symposium, August 2001.

[2] S. Staniford, J. Hoagland, J. McAlerney, Practical Automated Detection of Stealthy Portscans, http://www.silicondefense.com/pptntext/Spice-JCS.pdf

[3] Snort, www.snort.org

[4] Search Security Definitions, http://searchsecurity.techtarget.com

[5] CAIDA, www.caida.org

[6] S. Northcutt, Network Intrusion Detection Analyst's Handbook. New Riders, Indianapolis, 1999. p.125.

[7] Agenda and Work Plan. Computer Security Incident Response Team (CSIRT), Florida State University, http://www.security.fsu.edu/csirt_mtg

[8] Fyodor. http://www.insecure.org/nmap

[9] Vern Paxon. Bro: A System for Detecting Network Intruders in Real-Time. ftp://ftp.ee.lbl.gov/papers/bro-CN99.ps.gz

[10] S Staniford, V. Paxson and N. Weaver, How to 0wn the Internet in Your Spare Time, USENIX Security Symposium, August 2002.