Exposure Maps: Removing Reliance on Attribution During Scan Detection

David Whyte* P.C. van Oorschot* Evangelos Kranakis*

Abstract

Current scanning detection algorithms are based on an underlying assumption that scanning activity can be attributed to a meaningful specific source (i.e. the root cause or scan controller). Sophisticated scanning activity including the use of botnets, idle scanning, and throwaway systems violates this assumption. We propose a class of scanning detection algorithms that focus on what is being scanned for instead of who is performing the scanning. We pursue this idea, introduce the concept of exposure maps, and report on a preliminary proof-of-concept that allows one to: (1) estimate the information or exposures revealed to an adversary as a result of scanning activity; (2) detect sophisticated or targeted scanning activity with a footprint as low as a single packet or event; and (3) discover real-time changes in network exposures that may be indicative of a successful attack.

1 Introduction

Networks are constantly bombarded by backscatter packets [12], incessant probes from auto rooters, malware infected systems (e.g. worms), and Internet cartographers. Pang et al.'s analysis [13] reveals that the Internet is saturated with *nonproductive* network traffic. Yegneswaran et al. [19] estimate that there are 25 billion global intrusion attempts per day and this trend continues to increase. Unfortunately, effective security monitoring of network boundaries is seriously hampered because of a present inability to accurately discern sophisticated targeted scanning activity from unfocused background scanning activity. Exacerbating this problem is the availability of precisely such sophisticated scanning techniques and tools (see Section 2).

The majority of existing scanning detection schemes and proposals rely on observing and categorizing incoming network connection attempts. This characterization can be as simple as observing X events within a Y time window or it may contain a number of complex heuristics or behavioral patterns including statistical measures [11, 16], observing connection failures [10, 15], abnormal network behavior [17], connections to network darkspace [1, 4], or simply increased connection attempts [18]. Regardless of the characterization used, almost all current scanning detection algorithms correlate scanning activity based on the perceived last-hop origin of the scans; we call these *attribution-based* detection schemes. However, there are situations where determining *true* attribution (e.g. the actual scan controller, where this differs) is not possible. Furthermore, in some cases the use of attribution-based detection schemes is entirely ineffective as the scans may either be so slow or so broadly distributed that they exhaust finite computational state or fail to exceed some predefined alert threshold (see Section 2).

Our view is that against a growing array of attack strategies attribution (i.e. the identification of scanning systems) is becoming a quixotic approach to scan detection that overlooks an often critically important question that we suggest should be a much higher focus of scanning detection, namely, what is the adversary looking for? Although a network operator may be interested in knowing what type and amount of scanning activity is occurring, this is largely irrelevant if the proper security countermeasures are in place and software patches are up-to-date. However, the situation is different if any of the scans are a more likely precursor to a successful attack. Current scanning detection techniques do not take advantage of this observation.

Our idea is to observe both legitimate network activity and attack scans to dynamically enumerate the services currently being offered by a target network. These listening services are a normal source of information leakage from the target network to potential scanners; they can be measured and characterized in terms of what we call Host Exposure Maps and Network Exposure Maps. Once verified as permitted port/IP activity, these maps define the authorized access to the target network from external sources. Connection attempts to host-port combinations outside of these passively enumerated maps indicate a possible (sophisticated or simple) scan. In this note, we propose how to use exposure maps to allow a network operator to perform: (1) real-time verification of compliance with network and host security policies; (2) identification of both simplistic and sophisticated scanning activity regardless of scanning rate; and (3) rapid detection of changes in host and network behavior indicative of successful attack. In essence, our approach is to take advantage of adversaries by analyzing what they learn from the continual network vulnerabil-

[1] {dlwhyte, paulv, kranakis}@scs.carleton.ca. School of Computer Science, Carleton University, Ottawa, Canada. Version: 5 July 2006.

ity scanning they perform on a target network.

The paper is organized as follows. Section 2 presents brief background on sophisticated scanning techniques, highlighting limitations of attribution-based scanning methods. Section 3 outlines the basic idea of exposure maps. Section 4 describes an exemplar scanning detection technique using exposure maps. Section 5 discusses our proof-of-concept and preliminary experimental results. Section 6 reviews related research. We conclude in Section 7.

2 Shortcomings of Current Scan-Detection Approaches

Scanning activity can be broadly characterized into two categories: wide-range reconnaissance, and target-specific reconnaissance. Wide-range reconnaissance is used to rapidly scan large blocks of Internet address space to locate systems running a particular service or containing a specific vulnerability. Typically, there is little human interaction in this type of reconnaissance (e.g. worms, zombie recruitment for botnet enrollment, and auto rooters). Target-specific reconnaissance occurs when the information gathering activities are targeted or restricted to a predetermined entity. This type of reconnaissance is typically precise, deliberate, and focused. We now discuss a few of the sophisticated hard-to-detect strategies that could be used in this second category.

Current scanning detection algorithms are generally designed to: (1) classify suspicious network activity as scanning activity; and (2) attribute this activity to a particular source or sources. The following scanning techniques challenge both aspects of this traditional methodology.

Idle scanning. Idle scanning [14] allows an attacker to port-scan a target without sending a single packet from the attacker's own system.[1] The attacker first sends a SYN packet to the port of interest on the target host spoofing the source address of the packet with the IP address of an innocent system (hereafter referred to as a bot). If the port is open, the target responds to the bot with a SYN ACK. The bot does not expect this unsolicited SYN ACK packet so it responds with a RST packet to the target and increments the 16-bit identification field (IPID) it includes in its IP header. The attacker then sends a SYN packet to the bot and observes the IPID field of the RST packet the bot sends back. If the IPID has been incremented, the port on the target was open. Idle scanning utilizes side-channel communication by redirecting the scan and bouncing it off a third-party system. Most scanning detection algorithms will erroneously identify the third-party system as the scanner.

Botnet scanning. As is well known, a botnet is a collection of compromised systems (bots or zombies) used in a coordinated fashion and controlled by a single entity. A botnet can provide an attacker with, in essence, an unattributable method of reconnaissance. For instance, consider a botnet owner that has an exploit capability against a network service. A botnet of approximately 65,000 systems would be able to scan an entire Class B network for this service by sending a single packet from each bot (each with a unique IP address). In this example, even if it were possible to correlate this activity to a single scanning campaign, it still would not reveal the true adversary as the bots are simply zombie participants.

Throw-away scanning. An attacker can use a previously compromised or *throw-away* system to scan a network. The use of a different throw-away system to launch the attack essentially defeats attribution attempts by decoupling scanning and attack activities from a single system.

Low and slow scanning. An attacker may take days, weeks or months to scan a target host or network. Slow scans may blend into the network *noise* never exceeding detection thresholds and exhausting detection system state.

3 Basic Idea of Exposure Maps

Attribution-based scanning detection presupposes that identification of the root cause of scanning activity is possible. This assumption makes detection algorithms partially or completely ineffective in detecting certain classes of sophisticated scanning activity. Here we describe an example attribution-free scanning detection technique that our preliminary analysis suggests can detect sophisticated scanning using minimal resources (see Sections 4 and 5). Furthermore, although attribution is not relied on for detecting potential scans, in some instances attribution to the scanning source(s) is appropriate and can easily be determined *post-scan detection* (see Section 4). This allows our technique to detect both sophisticated and simple scanning activity.

Exposure Maps. A *host exposure map* (HEM) is constructed by passively observing a target network's traffic over a training period. During the training period, the behavior of individual systems within the network is recorded as they successfully respond to external stimuli (i.e. ICMP requests, TCP connection attempts, UDP datagrams). Over time, each host will be associated with a list of ports and protocols they will respond to, the HEM, when contacted by external systems. The HEM can be regarded as the externally visible surface of the host. As is the case with any technique that requires a training period, it is possible that malicious host activity may become part of the reference baseline for the host. Fortunately, the HEM can quickly be verified against the existing network security policy to ensure no unauthorized service is included in the HEM. The union of HEMs within a target network defines the *network exposure map* (NEM).

The NEM can be regarded as the externally visible surface (set of interfaces) of the network. Once constructed, the NEM can be compared to the network security policy to verify compliance and ensure that the hosts within a net-

[1] See also: Idle Scan and related (IPID) games, http://www.insecure.org/nmap/idlescan.html

work are providing only those services permitted by policy. At first glance, it may appear that this technique contains an inherent limitation in that scans to valid services (i.e. entries in the NEM) will not be detected. For instance, an HTTP scan to destination IP 10.0.0.1 in our network (i.e. our primary HTTP server, see Table 1) is considered valid activity and thus would not be considered a scan. In practice if using the NEM approach, this type of scan would be detected as it would almost certainly also occur against other hosts in the network not offering HTTP (i.e. not a valid IP/port tuple listed in the NEM). The scanning activity would not be detected if it were directed, although unlikely, solely at the HTTP server. However, we would consider the latter activity to be an actual attack rather than a scan; while our technique detects scans (as a precursor to attacks), we do not purport to detect actual attacks.

4 Scanning Detection Approach based on Exposure Maps

Once the training period has concluded and a NEM constructed, scanning detection is performed by simply recording any connection attempt (i.e. TCP connection *attempt*, UDP or ICMP datagram) to a host and port combination not found within the NEM; we call these *outside-NEM* scans. Each scan attempt is reported using a 6-tuple (source IP, source port, destination IP, destination port, protocol, timestamp). The approach does not require maintaining any state information other than the NEM and thus can detect very slow and distributed scan activities (recall Section 2).

Once these outside-NEM scans are recorded, a number of *post-scan* detection analysis activities are possible. For one, changes to the NEM itself could be monitored to detect potential malicious activity. If a high-order port number opens on one or more hosts simultaneously this could indicate either new legitimate services offered (i.e. the NEM has to be updated) or evidence of unauthorized software installation (e.g. a backdoor). As a second example analysis activity, monitoring for sudden increases in scanning activity could be used to identify bursts or unusual scanning activity against the NEM (see Figure 1(a)). This may prompt a network operator to check deployed network service patch versions and undertake research for any applicable newly released exploits or vulnerability information to gauge the current threat to the network.

Although Section 2 lists a number of scenarios where attribution is neither possible nor helpful, the exposure map approach does not preclude source attribution, in appropriate cases, *once a scan has been detected*. In fact, our preliminary analysis suggests that this approach can easily detect the source of both unsophisticated and some forms of sophisticated scanning activities (see Section 5). However, an important distinction must be made: here we are suggesting that in some cases, some form of attribution can occur *post-scan* detection, not as part of the scanning detection algorithm itself. Even in this case, attribution is not *relied on* in order to *detect* scans.

5 Proof-of-concept

As an initial test of our idea, we carried out a small proof-of-concept. The data set for our analysis consists of a two-week training and monitoring period of network traffic collected in pcap files from one of our university research labs containing 62 Internet-addressable systems. Our proof-of-concept recorded successful TCP connection attempts and UDP/ICMP responses to generate the NEM presented in Table 1. The NEM conformed to the network's existing security policy.

A standard Internet host has a total of $2 * 2^{16}$ UDP and TCP ports. Responses from any one of these ports indicates that a service is listening. Thus, to fully enumerate a host, an adversary would need to scan a total of 2^{17} ports. To fully enumerate the number of ports on a network, the total is $E = N * 2^{17}$ (where N is the number of systems in the network). E is the upper bound on the potential number of unique scan combinations a network could expect (e.g. in our network, E = 2^{23}). Our exposure map technique is very efficient as the NEM need only record and maintain in state the port/IP pairs that respond to connection attempts in order to perform scanning detection; the NEM need not record per-port information for each port in E. In our network, the NEM consisted of 11 entries (i.e. unique IP/port pairs).

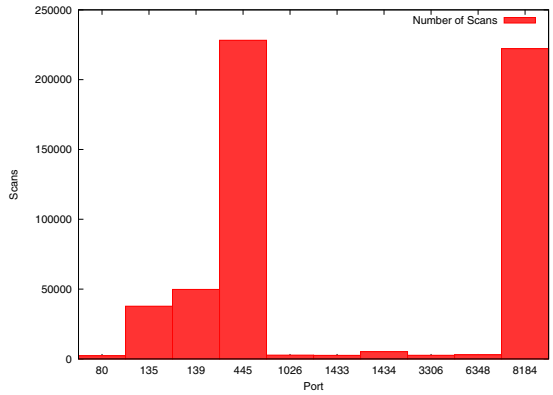
In practice, attackers typically port scan only a subset of available ports. The port scans that we have detected (in this and previous network data sets) have been either directed at well-known services in the reserved port range (i.e. 0-1023) or trojan backdoor ports. For example, using our technique we detected scanning activity against only 338 unique TCP and UDP ports over the entire two week period. The top ten ports scanned outside our target NEM are presented in Figure 1(a). Furthermore, although there were 776,074 scans detected by our technique, the actual scan footprint (*A*) consisted of only 6,131 unique IP/port combinations. In most networks, only a few hosts offer publicly available services. These host and port combinations will comprise the NEM: in our network 11 unique IP/port pairs (see Table 1). Figure 1(b) (not drawn to scale), shows the general relationship between potential service ports scanned (*E*), actual scans (*A*) and the NEM for a network.

Once scanning activity has been detected by using the NEM as in Section 4, a number of heuristics can be developed to classify the type of scans detected. We briefly discuss two example heuristics we created using simple scripts that reveal evidence of sophisticated scanning activity:[2]

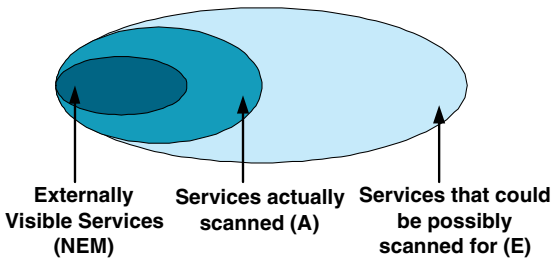
[2]The example heuristics we describe in Section 5 are performed post-detection and therefore the source address feature is used simply as a means to help classify the type of (not detect) scans performed against the target network.

Table 1: Network Exposure Map from small proof-of-concept trial

Host	Description	TCP Ports	UDP Ports
10.0.0.1	Mail/DNS/HTTP Server	22, 25, 80, 993, 631	53
10.0.0.2	DNS/HTTP Server	443, 80, 22	53
10.0.0.3	SSH Server	22	



(a) Top 10 Ports Scanned outside our target Network Exposure Map



(b) Scanning Potentials versus Exposed Network Services

Figure 1: Using Network Exposure Maps.

- *Number of scan packets sent to target destination ports*: All outside-NEM scans are first sorted by the number of scan packets sent into the network from a unique source address over a configurable time interval. A similar amount of scanning activity from individual sources comprise a cluster. These individual clusters are then sorted by the target destination ports. This final comparison can reveal coordinated scanning activity by identifying scanners that exhibit the same scanning frequency and targets of interest (e.g. services). Using this heuristic we detected a co-ordinated scan consisting of six systems registered to a single class C network directed to the same eight ports on every system in our network over the entire two-week period. Average network scanning rate from the group was 1 scan every 40 seconds. This activity is ongoing.
- *Target service and scanning interval*: All detected scanning activity is first sorted by unique source address. Using the time-stamp as a reference, scans from a fixed source address that have a scanning interval of less than 5 minutes are discarded. The remaining scans were then sorted by destination port. This heuristic detected a slow scan for the *pcanywhere* port (i.e. TCP port 5631) that occurred with an average scan interval of 15 minutes.

These two example heuristics detected two forms of sophisticated scanning activity (i.e. a co-ordinated and a slow scan) that would not be detected by most existing scanning detection schemes.

6 Related Work

Our work is related in part to that of Gates [9], which explores detection of co-ordinated scanning and includes an evaluation structure to predict scanning detection algorithm performance. A number of scanning detection techniques use evidence of connection failures as an indicator of scanning activity (e.g. [6, 10, 15]). Other scanning detection techniques consider external system connections to network dark space (i.e. no host at scan destination IP address) as a scan [1, 4]. The term *extrusion detection* has been used to describe the activity of monitoring for suspicious internal network connections to the Internet [7]. In contrast, exposure maps dynamically identify externally accessible host services in the network as a result of incoming network activity. Once the training period is complete, we do not require the observation of any responses from the internal network to determine if scanning activity has occurred. Furthermore, exposure maps provide the ability to detect real-time changes in host behavior (e.g. a host begins to respond on a port not listed in the HEM) that may indicate a successful compromise. A host-based extrusion detection system developed by Cui et al. , called BINDER[8], correlates outgoing connections with user input to detect outgoing activity not triggered by user activity. Finally, a few commercial products provide network behavior analysis and traffic profiling to detect malware, insider breaches, and security policy violations [2, 3, 5]. We plan to explore the capabilities of these products as we evolve the concept of exposure maps to gain a better understanding of the differences, advantages, and disadvantages.

7 Concluding Remarks

We suggest that reliance on attribution in scanning detection algorithms can be misleading, or fail entirely, for identifying some forms of scanning activity, as many sophisticated scanning techniques will easily evade attribution-based detection. Our proposed scanning detection approach shifts its focus to a characteristic of the activity that can be considered a *ground truth*, namely, the services or vulnerabilities that are being scanned. We expect that this should allow a network operator to more quickly determine potential targets and perform directed risk and security posture assessments accordingly.

We have developed and discussed one example of an attribution-free detection technique that our preliminary analysis reveals can detect both sophisticated and unsophisticated forms of scanning activity. Although attribution (i.e. source address correlation) is not required for scan *detection* in our algorithm, attribution can be easily performed to classify both unsophisticated and sophisticated scanning campaigns *post detection*. Exposure maps utilize both legitimate and malicious network activity to dynamically identify the responding hosts and services in the network. Our ongoing work includes developing a full prototype; further refining additional scan detection heuristics once a scan (i.e. atomic connection attempt) has been detected; and analyzing much larger network data sets to determine both the stability of Host Exposure Maps and further test the scanning heuristics we produce.

Acknowledgments

We thank anonymous reviewers, Anil Somayaji, and members of Carleton University's Digital Security Group for comments which significantly improved this paper. This research is supported in part by MITACS (Mathematics of Information Technology and Complex Systems). The second author is Canada Research Chair in Network and Software Security and is supported in part by an NSERC Discovery Grant and the Canada Research Chairs Program. The third author is supported in part by NSERC (Natural Sciences and Engineering Research Council of Canada).

References

[1] Forescout. Wormscout. http://www.forescout.com/wormscout.html.

[2] Lancope. StealthWatch. http://www.lancope .com.

[3] Mazu Networks. Mazu Profiler. http://www.mazu networks.com.

[4] Mirage Networks. Mirage NAC. http://www.mirage networks.com.

[5] Q1Labs. QRadar. http://www. q1labs.com.

[6] G. Bakos and V. Berk. Early detection of Internet worm activity by metering ICMP destination unreachable activity. In *SPIE Conference on Sensors, and Command, Control, Communications and Intelligence*, April 2002.

[7] R. Bejtlich. *Extrusion Detection, Security Monitoring for Internal Intrusions*. Addison Wesley, first edition, 2006.

[8] W. Cui, R. Katz, and W. Tan. BINDER: An Extrusion-based Break-in Detector for Personal Computers. In *Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC)*, December 2005.

[9] C. Gates. *Co-ordinated Port Scans: A Model, A Detector and an Evaluation Methodology*. PhD thesis, Dalhousie University, 2006.

[10] J. Jung, V. Paxson, A. Berger, and H. Balakrishman. Fast portscan detection using sequential hypothesis testing. In *IEEE Symposium on Security and Privacy*, pages 211–225, 2004.

[11] C. Leckie and R. Kotagiri. A probalistic approach to detecting network scans. In *Eighth IEEE Network Operations and Management Symposium (NOMS 2002)*, pages 359–372, 2002.

[12] D. Moore, G. Voelker, and S. Savage. Inferring Internet denial of service activity. In *10th USENIX Security Symposium*, 2001.

[13] R. Pang, V. Yegneswaran, P. Barford, V. Paxson, and L. Peterson. Characteristics of Internet background radiation. In *The Internet Measurement Conference ICM*, 2004.

[14] S. Sanfilippo. Bugtraq: new tcp scan method. December 1998. http://seclists.org/lists/bugtraq/1998/Dec/0079.html.

[15] S. Schechter, J. Jung, and A. Berger. Fast detection of scanning worm infections. In *7th International Symposium on Recent Advances in Intrusion Detection (RAID 2004)*, September 2004.

[16] S. Staniford, J. Hoagland, and J. McAlerney. Practical automated detection of stealthy portscans. In *7th ACM Conference on Computer and Communications Security*, 2000.

[17] D. Whyte, E. Kranakis, and P. van Oorschot. DNS-based detection of scanning worms in an enterprise network. In *Proc. of the 12th Annual Network and Distributed System Security Symposium*, Feb. 2005.

[18] M. Williamson. Throttling viruses: Restricting propagation to defeat malicious mobile code. In *18th Annual Computer Security Applications Conference (ACSAC)*, 2002.

[19] V. Yegneswaran, P. Barford, and J. Ullrich. Intrusions: Global characteristics and prevalence. In *SIGMETRICS*, 2003.