

- for Programming Languages and Operating Systems. New York, NY, USA: ACM, 2006, 25–36.
3. Joanna Rutkowska, “Subverting Vista kernel for fun and profit”. Black Hat 2006, 2006 <<http://www.blackhat.com/presentations/bh-usa-06/BH-US-06-Rutkowska.pdf>>
 4. Dino A. Dai Zovi, “Hardware virtualization rootkits”, 2006. <<http://www.blackhat.com/presentations/bh-usa-06/BH-US-06-Zovi.pdf>>
 5. Samuel T. King, Peter M. Chen, Yi-Min Wang, Chad Verbowski, Helen J. Wang and Jacob R. Lorch, “SubVirt: implementing malware with virtual machines”. SP ’06: Proceedings of 2006 IEEE Symposium on Security and Privacy. Washington, DC, USA: IEEE Computer Society, 2006, 314–327.
 6. Jonathan M. McCune, Bryan J. Parno, Adrian Perrig, Michael K. Reiter and Hiroshi Isozaki, “Flicker: an execution infrastructure for TCB minimization”, *SIGOPS Oper. Syst. Rev.* **42** (2008) 4: 315–328.
 7. Jonathan M. McCune, Bryan J. Parno, Adrian Perrig, Michael K. Reiter and Arvind Seshadri, “How low can you go? Recommendations for hardware-supported minimal TCB code execution”. ASPLOS XIII: Proceedings of 13th International Conference on Architectural Support for Programming Languages and Operating Systems. New York, NY, USA: ACM, 2008, 14–25.
 8. Melvin J. Anderson, Michae Moffie and Chris I. Dalton, “Towards trustworthy virtualisation environments: Xen library OS security service infrastructure”. Tech. rep., HP Laboratories, Bristol, UK, 2007.
 9. Tal Garfinkel, Ben Pfaff, Jim Chow, Mendel Rosenblum and Dan Boneh, “Terra: a virtual machine-based platform for trusted computing.” SOSP ’03: Proceedings of 19th ACM Symposium on Operating Systems Principles. New York, NY, USA: ACM, 2003, 193–206.
 10. Rafal Wojtczuk, “Subverting the Xen hypervisor”, 2008. <http://invisiblethingslab.com/bh08/papers/part1-subverting_xen.pdf>
 11. David Grawrock, “The Intel safer computing initiative”. Intel Press, 2006.
 12. Intel, “Intel trusted execution technology measured launched environment developer’s guide”, 2008. <<http://download.intel.com/technology/security/downloads/315168.pdf>>
 13. Yuriy Bulygin and David Samyde. “Chipset based approach to detect virtualization malware a.k.a. DeepWatch”. Black Hat USA, 2008 <<http://www.mnm-team.org/pub/Fopras/frit08/PDF-Version/frit08.pdf>>
 14. Shawn Embleton, Sherri Sparks, and Cliff Zou, “SMM rootkits: a new breed of OS independent malware”. SecureComm 2008, Istanbul, Turkey, 2008.

Network reconnaissance

Siraj A. Shaikh, research officer, Department of Informatics and Sensors, Cranfield University, UK

Howard Chivers, professor, Department of Informatics and Sensors, Cranfield University, UK

Philip Nobles, lecturer, Department of Informatics and Sensors, Cranfield University, UK

John A. Clark, professor, Department of Computer Science, University of York, UK

Hao Chen, research associate, Department of Computer Science, University of York, UK

Along with its wider reach in society, in the form of both mobility and relatively affordable access, the internet has transformed the world we live in, serving as bedrock for electronic commerce and other digital and communication services. It has become an integral part of the personal, professional, and economic spheres of our daily life. Global organisations, whether official, commercial, or social, are relying on it ever more to function, bringing an increasing need for a secure electronic infrastructure. The pervasive nature of the internet, one major factor behind its success, is also proving to be its main threat. Once connected to this global network, no one is more than a few clicks away from servers hosting websites that transact commerce worth millions or critical state-run networks that run sensitive operations.

While such infrastructures are increasingly under threat from worm propa-

gation and script kiddies launching denial of service attacks, a more severe

threat is posed by sophisticated intruders who:

- Attempt slow and subtle attacks
- Have a presence or access on the inside
- Target specific resources
- Possibly make use of zero-day exploits.

Sabotage attempts launched by such intruders are likely to be more damaging and difficult to detect early on. An important first step that these intruders take is to perform intelligence gathering.¹ This involves engaging in reconnaissance exercises to scan networks for connectable active hosts, enumerate services, and identify potential vulnerabilities that could be exploited. The more an intruder is aware of her target (in terms of topology, address space, and services) the greater her chances are of launching an exploit and successfully evading detection.

Probes						
Activity	Host detection		Port enumeration		Vulnerability assessment	
Layer	Data link	IP	TCP	UDP	OS	Application
Desired Information	Hardware address	Network address	Service enumeration Port address		Identification Version identification Patch level information	
	Host liveness					

Table 1: A classification of probes.

These attempts at intelligence gathering need to be detected early if the defenders are to have any chance of preventing intrusions and identifying the intruders. Detecting such intruders early on is not easy however. Skilful intruders attempt to conduct reconnaissance:

- At a very low frequency¹
- Possibly using insider hosts as a launch pad
- Using a wide variety of protocols (IP/ICMP/UDP/TCP) and their characteristics to reveal information about target hosts²
- As part of distributed information gathering, using different source addresses some or all of which may be compromised or spoofed.^{1,3,4}

Added to these are other more conventional challenges, amidst which distinguishing a skilled reconnaissance attempt aimed at a specific target becomes very difficult. Notably:

- Huge volumes of traffic monitored at medium to large networks of today
- Continually growing scanning activity on public networks, at a growing scanning rate over the years, with tens of thousands of probes increasingly common⁵
- Rising proportions of non-productive traffic on public networks.
- Traffic destined for the unused part of the address space or used space that does not want to receive traffic, mostly consisting of backscatter, spam or malicious traffic.⁶ Such traffic is found to be persistent and originates

from a variety of sources including intruders, worm-infected machines, hosts responding to flooding attacks, and even misconfigured hosts.

We study reconnaissance behaviour over networks and classify it in terms of service layers and the type of information sought. We also discuss some of the challenges in detecting such behaviour and how intruders do their utmost to evade detection. Finally, we delve into various techniques used to respond to reconnaissance activity.

Reconnaissance

In traditional warfare, an enemy would attempt to gather as much information as possible before launching an attack. She would strive to know about the resources available, their weaknesses and vulnerabilities and, where it would hurt the most. She could use a variety of methods to gather this information including surveillance, eavesdropping on and intercepting communications, and launching spying probes. A similar approach is adopted by intruders over computer networks to gather information about computer systems and resources. A probe is any attempt launched to detect:

- Active hosts and networks that are reachable over a public or an accessible medium
- The services and applications they are running that could be connected to any vulnerability that these services and applications may have,

which could be exposed and taken advantage of.

Probes can be classified appropriately into three main activities: host detection, port enumeration and vulnerability assessment, as shown in table 1. Each activity is different in the type of information it seeks to establish and the service layer it operates at.

“In traditional warfare, an enemy would attempt to gather as much information as possible before launching an attack”

Host detection essentially aims to establish liveness of a host, along with its network address. Hardware addresses may also be sought by intruders having access to the same segment as the target. Port enumeration is to do with the listing of TCP/UDP services running on a host. This may be a list of all services or only those of particular interest to an intruder, along with the port address they are running on. Vulnerability assessment seeks to establish information on the type and version of the operating system and the different applications running on a machine. Version and patch level details about an operating system and applications are important to judge the possible exploits that could be used to attack the host.

A probe could be seen to be launched by an intruder in two modes: active and passive. An active probe involves some attempted interaction over the network on behalf of the intruder. This may involve sending a packet directly to a target host or a

network, or some intermediary used for the purposes of probing. A passive probe, on the other hand, would involve an intruder restricting herself to sniffing and logging traffic, originating from and destined to a potential or an identified target, and obtaining relevant information. The choice of being passive may be due to reasons of configuration or access, or it may be a deliberate act by an intruder to avoid detection. Passive probes by their nature are hard to detect. Any reconnaissance information gained using such tactics, however, is limited to the traffic visible to an intruder. Active probes are necessary if an intruder wishes to gather information both timely and of her choice.

A variety of techniques exist for active probes, including making use of mechanisms such as the TCP handshake to judge a host's liveness, fingerprinting the protocol stack (which often indicates the operating system the host is running), probing DNS servers, and grabbing service banners volunteering information on the host. Table 2 below attempts to classify active probing and gives us some idea of a wide variety of techniques used by intruders for this purpose.

Detection challenges

Detecting and preventing probes is important both to prevent intrusions and inhibit exposure of information about resources on the network.

While a variety of techniques exist for this purpose, efficient detection of such activity is not always straightforward.^{7,8} We discuss some of the challenges faced by analysts and identify potential areas that merit further investigation.

A common presumption is that intruders probe targets and follow up with exploitation. Once the services on a target host are enumerated or a particular vulnerability identified, not all intruders follow suit and send exploit packets. Often only a small percentage of detected probes are actually followed up by exploitation attempts. Panjwani et al.'s effort highlights this.⁹ Their study finds that only 4% of the port scans, for example, are actually followed up by an attack. The fact is particularly important from the perspective of detection systems, given the volume of potential probe traffic detected. Critical then are also any characteristic patterns of traffic from the point of initial detection to attempted exploitation.

"Intruders are increasingly making use of compromised hosts to launch reconnaissance against target networks"

Once an intruder spots a vulnerability of interest, she is disposed to take advantage of it immediately. The more she delays the exploit, the more she runs the risk of the reconnaissance

information becoming useless. This may not always be the case, however, as for reasons of stealth an intruder may attempt to probe the target low and slow.¹ Given such motivation it is not always clear as to how long a detection system should wait for an intruder to send an exploit once an initial probe is detected. Such systems are often constrained in their usage of memory, which serves to affect both their performance and scalability. Some estimation of how long intruders keep reconnaissance information before they make use of it would therefore be very beneficial. What is of particular interest here is if there are any specific exploits that arrive immediately (or with a notable delay). Such patterns could provide an insight into the various strategies adopted by the intruder for this purpose.

Intruders are increasingly making use of compromised hosts to launch reconnaissance against target networks.³ This not only gives them the advantage of hiding their own identity, but also of using a variety of source addresses to perform reconnaissance. The more source addresses they can use for this endeavour, the better their chances of evading detection. A similar strategy is often used for sending exploits, in which the exploit payload may not arrive from the same source address that the probe came from. This makes it very challenging to form any correlation

Active probes						
Activity	Host detection		Port enumeration		Vulnerability assessment	
Layer	Data link	IP	TCP	UDP	OS	Application
Techniques	Echo	ICMP echo ICMP non-echo RESET scan Invalid Protocol Response	SYN scan (Full open) SYN scan (Half open) FIN scan XMAS scan NULL scan SYN/ACK scan ACK scan	UDP scan	TCP/IP stack fingerprinting Obtaining DNS host information (HINFO) Patch level Assessment	Exploitation Banner Grabbing Password Auditing

Table 2: A classification of active probes.

between the different source addresses used to launch probes and to send exploit packets. A variety of factors do influence this tactic however: the amount of time elapsed between the initial probe and the exploit sent, the number of hosts targeted, and the type of attack planned. A worm infection, for example, is likely to probe and exploit from the same source and immediately, as opposed to a more targeted attack that is likely to perform a vertical probe against a single or few targets originating from multiple sources. Such factors could inform the deployment of detection systems by identifying patterns of source addresses used for launching attacks.

Responding to probes

Most active probes make use of techniques that use the core protocols of the modern day communications, namely IP, ICMP, TCP and UDP.¹⁰ Common approaches to counter-probing activity at this level include:¹⁰

- Filtering inbound ICMP probes (responses to which are used to determine what machine is alive)
- Filtering outbound ICMP responses to UDP port scanning attempts (where a lack of response allows an intruder to determine a live host)
- Filtering inbound TCP probes with different combinations of flags set, (response, or lack of it, to which (depending on the flags set and the operating system probed) may indicate to an intruder whether a host is live or not)
- Using a variety of firewalling techniques that allow throttling of probes and stateful mechanisms that disallow unsolicited packets aimed at generating responses from target hosts.

A somewhat more proactive approach is suggested by Kang et al. who propose to generate false positive responses to any probes attempting to detect hosts or enumerate ports targeting an unused address space or closed ports on active hosts.¹¹ Their approach,

referred to as all-positive response (APR), is designed to make it difficult for an intruder to distinguish active hosts from inactive ones, and open ports from closed ones. To an intruder, all machines appear active and all ports appear open. Such an approach could also help in detecting any packets that follow up after initial probes, which attempt to probe the host further, enumerating ports or assessing some vulnerability.

Using false responses is useful in hiding any information about the network that an intruder may try to gather. But an all-positive approach will certainly indicate to an intruder that false responses are being generated to all probing. Another important issue is that generating false responses for a very large network may require untenably large resources, and may therefore not be scalable. Some factors to consider here are the size of the entire (used and unused) address space that the false response needs to be generated for, the rate at which the network is probed, the various types of probes launched (that need to be responded to) and memory state required to detect any attempts at intrusion that follow up a false response.

Generating a false positive response to probes targeting a closed port on an active host could also result in a conflict: an active host may have a port closed at the time of the probe, but the port may open (upon the host initiating a connection or starting a service, for instance) some time after the false response is generated. Some alternatives to APR could be designed so that such responses are generated:

- Randomly where some probes are replied to and some are not (chosen entirely at random)
- To a specified subset of the unused address space. This subset could be chosen randomly (from a given chunk of addresses) or strategically (from an address space used non-contiguously)
- For all probes destined for the unused address space. This is similar to APR,

except that only probes destined for the unused parts of the address space are replied to and one or a few services depicted.

Acknowledgment

This effort is a result of a collaborative project between Cranfield University and the University of York funded by the EPSRC (EP/E028268/1) to study system level approaches to intrusion detection.

References

1. Hybrid, "Distributed information gathering", *Phrack Magazine*, 1999, Vol. 9, Issue 55, Article 9, 11 October 2008 <<http://www.phrack.org/issues.html?issue=55&id=9#article>>
2. M.D. Vivo, E. Carrasco, G. Isern and G.O. Vivo, "A review of port scanning techniques", *ACM SIGCOMM Computer Communication Review*, 29.2 (April 1999): 41-48. New York: ACM.
3. M.A. Rajab, J. Zarfoss, F. Monrose and A. Terzis, "A multifaceted approach to understanding the botnet phenomenon". Proceedings of 6th ACM SIGCOMM Conference on Internet Measurement (IMC'06), 25-27 October 2006, Rio de Janeiro, Brazil, New York: ACM, 41-52.
4. CERT IN-98-05, "Probes with spoofed IP addresses". CERT Incident Note, 24 November 1998 <http://www.cert.org/incident_notes/IN-98-05.html>
5. M. Allman, V. Paxson and J. Terrell, "A brief history of scanning". Proceedings of 7th ACM SIGCOMM Conference on Internet Measurement (IMC'07), 24-26 October 2007, San Diego, California, USA, New York: ACM, 77-82.
6. R. Pang, V. Yegneswaran, P. Barford, V. Paxson and L. Peterson, "Characteristics of internet background radiation". Proceedings of 4th ACM SIGCOMM Conference on Internet Measurement (IMC'04), 25-27 October 2004, Taormina, Sicily, Italy, New York: ACM, 27-40.
7. S. Staniford, J.A. Hoagland and

- J.M. McAlerney, "Practical automated detection of stealthy portscans", *Journal of Computer Security* **10(1-2)** 105-136.
8. C. Gates, "Co-ordinated port scans: a model, a detector and an evaluation methodology". PhD thesis. Dalhousie University, Canada, February 2006.
 9. S. Panjwani, S. Tan and K.M. Jarrin, "An experimental evaluation to determine if port scans are precursors to an attack". Proceedings of 2005 International Conference on Dependable Systems and Networks (DSN'05), 28 June-1 July 2005, Yokohama, Japan. Washington: IEEE Computer Society, 602-11.
 10. C. McNab, "Network security assessment: know your network". 1st edn., March 2004, Chapter 4. O'Reilly Media.
 11. M.G. Kang, J. Caballero and D. Song, "Distributed evasive scan techniques and countermeasures". Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA 2007, Lucerne, Switzerland, 12-13 July 2007. Proceedings, volume 4579 of LNCS, Berlin: Springer-Verlag, 157-174.

Technological alternatives in incident response

Dario Forte, CISM CFE, founder and CEO of DFLabs

We have discussed incident response many times in *Network Security*. The technical tasks (which include many pertinent to computer forensics) are handled mainly with the use of technology based on scientifically established methodologies.



Incident response and computer forensics can be applied using technologies that perform analyses either locally or remotely. In the former case, the requisite operating conditions include media available for analysis.

The media, such as hard disks and various types of mass storage device, must be available directly at the site where acquisition and analysis will be carried out. This assumes that the machine or devices in question can be excused from company IT operations for the time it takes to make forensic copies. This time is directly proportional to such factors as the type and size of the hard disk, or the type of hardware and software used to make the copy and verify the integrity of the data using techniques such as file hashes.

In my own experience, it is not always possible to have direct access to the machine requiring analysis. This is especially true for portable devices or those used by mobile operators who work away from company offices. Additionally, in post mortem analyses,

it is not always possible to access compromised servers since they may be in a different physical location from that of the operator.

"It is not always possible to have direct access to the machine requiring analysis. This is especially true for portable devices or those used by mobile operators who work away from company offices"

The question of media availability also has repercussions on evidence preservation. The potential sources of digital evidence must be acquired and preserved in an integral state for their eventual presentation in the legal arena. Proper evidence preservation is more difficult if the media to be investigated are not directly available to investigators.

Just as the media must be available, the availability of human resources necessary to carry out the forensic analysis is crucial. The more investigators a team

has, the faster a forensic analysis of a collection of physical machines can be carried out.

Distributed vs. remote forensics

To resolve the above problems, some years ago technology vendors began to release interesting agent-based architectural solutions allowing multiple access to target machines from a single workstation. Machines can be accessed on different levels, from the most volatile data to the most static (the file system). This architecture allows for analysis of a very high number of machines simultaneously (we are talking about thousands of target RAMs analysed in just a few hours) by a single operator, with obvious savings in terms of time and human resources.

Unfortunately, these systems are really only an option for large organisations, whose operational distribution demands paring response times