Attack Processes Found on the Internet

Marc Dacier, Fabien Pouget
Eurecom
2229, route des Crêtes; BP 193
06904 Sophia-Antipolis Cedex; France

{dacier, pouget}@eurecom.fr

Hervé Debar
France Télécom R&D
42, rue des Coutures; BP 6243
14066 Caen Cedex 4; France

herve.debar@francetelecom.com

ABSTRACT

In this paper, we show that simple, cheap and easily deployable honeypots can help to get a better understanding of the attack processes that machines in unclassified networks are facing. Acquiring this knowledge is a prerequisite for the sound design and implementation of efficient intrusion tolerant systems. We propose some in depth analyses carried out on data gathered during a 10 months period by several honeypots. We highlight the need for a well defined set up of honeypots, replicated in many diverse locations. Such an environment would enable the scientific community to answer the remaining open issues described here after.

1.0 INTRODUCTION

Recently, several papers have explained how so-called "Internet telescopes" can be used to get a better understanding of worms propagations ([44, 45]). As a follow up to large scale DdoS attacks, the scientific community has shown a growing interest for a pragmatic analysis of real data streams to identify the various attack processes threatening Internet users. For instance, three papers where devoted to these problems during the last SIGCOMM conference [60], and four other published at INFOCOMM 2003 [34].

In this paper, we propose to use simple honeynets instead of large "telescopes" to gather data. Based on our experimental set up we show that despite orders of magnitude of differences in the amount of collected data this approach is more efficient. Indeed, the limited, yet richer, amount of data offers a convenient way to carry out some systematic analysis that leads to interesting findings.

This paper is a follow up to an earlier publication [18] in which first results obtained over a four months period had been proposed. In this new publication, we review the conclusions drawn in [18] thanks to six more months of data and, more importantly, propose new findings. We discuss the need for a distributed infrastructure in order to confirm some of our conjectures and to answer some of the remaining open questions.

The paper is organized as follows. Section 2 proposes a general introduction to the notion of honeypots. Section 3 offers a survey of existing work. Section 4 describes the set up we have used for our experiments. Section 5 reviews the results proposed in [18] and offers new material. Section 6 concludes the paper.

Paper presented at the RTO IST Symposium on "Adaptive Defence in Unclassified Networks", held in Toulouse, France, 19 - 20 April 2004, and published in RTO-MP-IST-041.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 NOV 2004	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Attack Processes Found on the Internet	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Eurecom 2229, route des Crêtes; BP 193 06904 Sophia-Antipolis Cedex; France**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM001845, Adaptive Defence in Unclassified Networks (La defense adaptative pour les reseaux non classifies)., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**56**	

2.0 HONEYPOTS: INTRODUCTION

2.1 Definitions

Honeypots, honeytokens and honeynets have been used for more than fifteen years in computing systems even if the use of this terminology is recent. In the late 80's, Clifford Stoll [67] had the idea of placing 'interesting' data in appropriate places to lure hackers. In the 90's Cheswik implemented and deployed a real "honeypot" [12]. Bellovin discussed the very same year the advantages and problems related to its usage [6]. In 98, Grundschober and Dacier ([26, 27]) introduced the notion of "sniffer detector" (see also [1]), one of the various forms of what is called today a "honeytoken".

Lance Spitzner has the merit of having been the first one to try to define these concepts by introducing the terms "honeypot", "honeytoken" and "honeynet". Yet, we feel uncomfortable with his definitions because they describe how to use honeypots instead of defining what they really are. However, due to the lack of any better definition, the community of honeypot users/developers keeps using the following ones, taken from [65][1]:

- *Definition 1: «A honeypot is an information system resource whose value lies in unauthorized or illicit use of that resource.» [65]*
- *Definition 2: «[…] a honeytoken is a honeypot that is not a computer. Instead it is some type of digital entity. » [65]*

In order to provide a more rigorous definition of what honeypots are, we propose to take advantage of well defined concepts introduced by the dependability community. To that end, we reuse the definitions proposed by several contributors within the European MAFTIA project. In [53, page 32], the authors introduce the notions of attack, vulnerability and intrusions as follows:

- *Definition 3: «An **attack** is a malicious interaction fault, through which an attacker aims to deliberately violate one or more security properties ; an intrusion attempt»*
- *Definition 4: «A **vulnerability** is a fault created during development of the system, or during operation, that could be exploited to create an intrusion»*
- *Definition 5: «An **intrusion** is a malicious, externally-induced fault resulting from an attack that has been successful in exploiting a vulnerability»*

Based on these definitions, we derive the following new one:

- *Definition 6: A **honeypot** consists in an environment where vulnerabilities have been deliberately introduced in order to observe attacks and intrusions.*

2.2 Existing platforms

During the last 2 years, many different implementations of the concept of honeypots have been proposed. Some attempts have been made to classify them (see for instance [15, 29, 64]). For the sake of completeness, we offer a brief overview of the existing solutions in Appendix A. We report the interested reader to [52] for a more detailed presentation of these tools as well as for a discussion of the classification issues.

Besides the large number of solutions, it is worth noting that some work has also been done to propose software architectures, such as the GENI and GENII Honeynet [32], to create more sophisticated

[1] It is worth noting that a) this definition is different from the one given by the same author in his book [64], ii) this new definition has been discussed at length on the honeypot mailing list [30] but no final consensus has been reached among the participants.

environments by integrating various pieces of hardware and software. More recently, a strong trend has emerged in favor of the use of so-called virtual honeynets where a complete network is emulated by a single machine using tools such as User Mode Linux [31, 71], VMware [59, 72] or VSERVER [73].

3.0 STATE OF THE ART

3.1 Introduction

As indicated by Appendix A, most of the effort spent by honeypots developers has been devoted to implementation issues. Besides this, we identify in Section 3.2 three main trends in the usage of data collected by honeypots. In Section 3.3, we also consider the vast body of knowledge that has been accumulated by the people studying networking performance issues.

3.2 Research on data collected by honeypots

3.2.1 Post mortem analyses

In this case, honeypots machines are expected to be fully compromised by hackers [50]. Once this stage is reached, the machine is halted and analyzed by means of tools such as the « coroner toolkit » [25] or its successor, the «Sleuth kit» [9]. The main purpose of these analyses is to discover new software tools used by hackers and not yet found in the wild.

3.2.2 Identification of new threats

It has been claimed that honeypots could also be used as early warning systems. They could be designed to quickly identify new types of threats. However, as far as we can tell, no published work has investigated this problem in some depth. Some anecdotic report exists, such as the ones done with wireless honeypots [16, 39, 75] that aim at showing the risk of leaving wireless networks unattended [74].

3.2.3 A statistical data gathering tool

The rapid evolution of existing platforms might be the reason for the surprising lack of publication of data collected by honeypots over a long period of time. The most visible project, the honeynet project has published a first document in 2001 [30] but, since then, seems to have focused on implementations issues. The Irish team appears to be the only member of the Honeynet Research Alliance to offer such data on its web site [28] but this concerns their sole environment and it does not provide any kind of analysis. A noteworthy exception in the field of statistical data analysis can be found in [20] where the authors analyze, thanks to their honeypots, the propagation of the NIMDA worm. Thus, as of today, only one team seems to have investigated the possibility of using data from honeypots to model attack processes.

3.3 Research in network monitoring

As early as 1993, Bellovin [7] has shown the interest of studying real packets passing on the networks. He showed the existence of anomalous behaviors, packets that did not indicate an attempted break-in but that, nevertheless, were worthy of attention. The museum of broken packets [78] offers a survey of such weird packets. There is now a conference [37] (previously a workshop [35, 36]) where results of work dedicated to the analysis of real data streams are presented. Security issues were not really considered by that community in the past. The focus was more on performance, quality of service and optimization issues. The rise of denial of service attacks has changed things. In 2001, Moore et al. published the first quantitative analysis of these phenomena [46], followed by others [5, 44, 63]. During the last SIGCOMM conference [60], three papers were dedicated to that problem. Four others were published at the last

INFOCOM [37]. Two things are important to mention at this point: i) these studies focus on Denial of Service attacks, which represent only the tip of the iceberg in terms of attack processes to look at, ii) besides [11] none of these teams has had access to data collected by honeypots.

4.0 TESTBED DESCRIPTION

Our experimental set up consists in a virtual network built on top of VMware [72] to which three virtual machines, or guests in the VMware terminology, called mach0, mach1 and mach2 are connected. The VMware commercial product enables us to configure them according to our specific needs. mach0 is a Windows98 workstation, mach1 is a Windows NT Server and mach2 is a Linux Redhat 7.3 server. The three virtual guests are built on non-persistent disks [72]: changes are lost when virtual machines are powered off or reset. In other words, rebooting a compromised machine is a convenient and simple backward error recovery mechanism. The three machines are attached to a virtual Ethernet switch[2]. ARP spoofing is implemented so that they can be reached from the outside world. A fourth virtual machine is created to collect data in the virtual network. It is also attached to the virtual switch and tcpdump is used as a packet gatherer [69]. This machine and the VMware host station are configured to be invisible from the outside world. Both Mach1 and Mach2 run an ftp server; in addition, Mach1 also provides a static web server. Logs are collected daily and transferred to an independent and safe place where they are enriched with some external data, as discussed below, and inserted into a database.

5.0 RESULTS

5.1 Introduction

First and foremost, we introduce two definitions which will be used in the following:

- *Attack Source:* it defines an IP address that targets our honeypot environment within one day. This time constraint is arbitrary and based on our observation only: attacks are all limited to short time periods (no more than 1 minute). Thus, if the same IP address is sending packets to one of our honeypots on the 13th of March and then on the 28th of March, we consider that they come from two distinct *attack sources.*
- *Ports sequences:* attack sources send packets to specific ports of the honeypots. A *Ports Sequence* defines the specific order according to which ports have been targeted on a given honeypot. For instance, if source A sends requests on port 80, and then on ports 8080 and 1080, the associated *ports sequence* will be {80; 8080; 1080}.

In the following, we report the results collected from March 1 until December 31, 2003. During that 10 months period, we have observed a total number of 3028316 incoming packets, sent by 18075 different sources. The attack sources rate is quite constant over the months as illustrated in Figure 1 where we have represented the number of observed sources per month. The slight increase observed in November and in December is due to the worm MBlaster, as explained in Section 5.3.2.

[2] A switch in the Vmware jargon but it actually behaves as a hub.

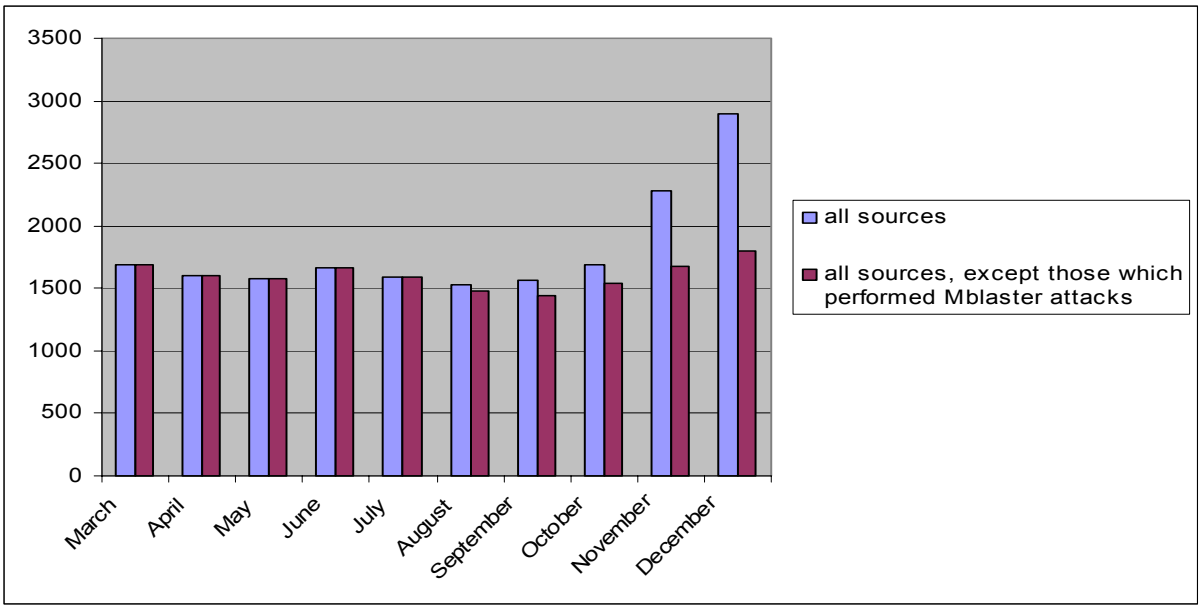


Figure 1: Number of attack sources per month.

We observe some changes with the results presented in [18]. First, the vast majority of the observed packets are TCP ones (84.7%). Others are ICMP (1.6%) and UDP packets (13.7%). In [18], almost all observed packets were TCP ones (97.9%). The difference is due to the increase of attacks to the NetBios Name Service (more precisely attacks to UDP port 137 associated with attacks on TCP port 139) at the end of 2003 (see Section 5.3.1).

However, we confirm that attacks are directed to a very specific number of ports, 188 in total. In 69% of the cases, an attacking source has sent requests to the three honeypots in a very short period of time (typically in a few seconds). In only 5.5%, attack sources contact only two out of the three machines. However, and this is something important we discuss later, in 25.5% they have focused on only one of the three honeypots. Interestingly enough also, we noted in [18] that no IP address did seem to come back: none had been observed during more than one day. This property is confirmed with a few exceptions though: 81 IP addresses have been seen many times (one of them has been observed in 14 different days) from August to December. They all seem to be taking part to the same attack process described in Section 5.3: they periodically test a known vulnerability on ports 135 (Microsoft RPC end-point mapper) and 139 (NetBIOS File and Print Sharing).

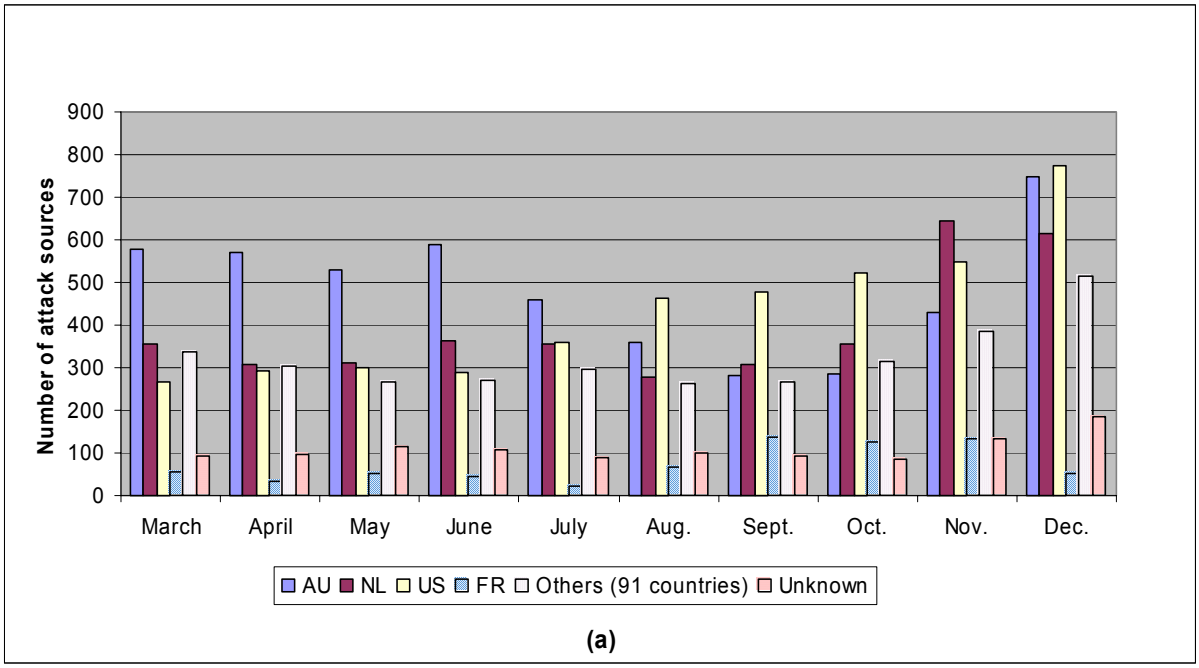
There are many things that could be said with the data we have collected so far, but in the context of this paper, we want to show two things. First, it is indeed possible to learn something about the attack processes and the threats we are facing. The data highlights the existence of some stable processes that could and should be modeled. Second, the observations we have made show the need for designing a more global set up, to answer questions that are left open.

Our results are divided into three main categories that characterize i) the attacking machines ii) the attacked machines iii) the attacked ports.

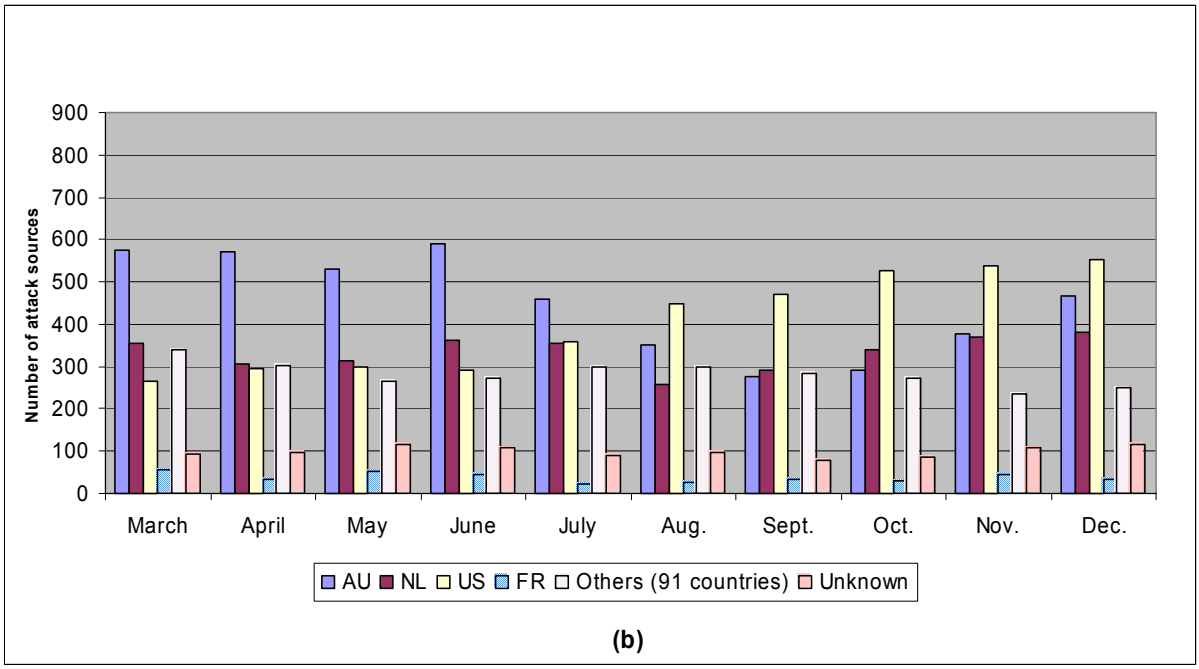
5.2 Information on attacking machines

5.2.1 Geographical location

Geographical location tells us where attacks are coming from. It is obtained thanks to the Netgeo utility [48], developed in the context of the CAIDA project[3]. This is a database and a collection of sophisticated perl scripts that map IP addresses and AS numbers to geographical locations. The result of running the Netgeo script on our logs is represented in Figures 2 a, b and c.



(a)



(b)

[3] It seems that Netgeo is hedging some results and favors the Netherlands as European default country in some undetermined cases. We intend to use MaxMind GeoIP, a commercial utility very similar to NetGeo in order to validate this observation.[42].

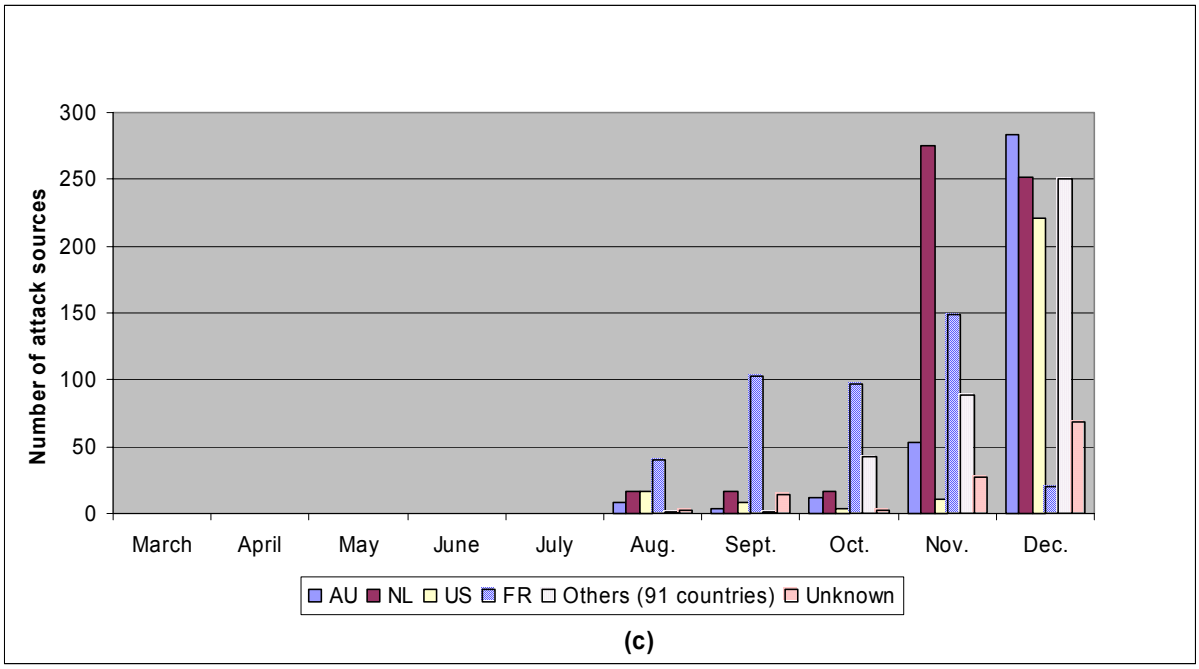


Figure 2 a) b) c): % of IP Sources per geographical location over 10 months: Figure a) for all sources – Figure b) for all sources but those associated to variants of MBlaster – Figure c) for sources associated with MBlaster only.

Figure 2 a) represents the number of attack sources from different geographical locations per month. Surprisingly enough, 71% of the attacks originate from only three countries: Australia (26.6%), the USA (22.7%) and the Netherlands (22%), whereas more than 90 other countries are also represented. These three countries are definitely not the usual suspects that would be quoted by security experts [23]. Furthermore, we would like to point out the increasing part of attacks coming from France and Netherlands since September. The main reason lies in the increasing MBlaster traffic. Figure 2 b) is similar to Figure 2 a), where we have removed sources associated to MBlaster (all variants). Figure 2c) instead, represents only those sources associated to MBlaster. These figures confirm that the majority of the MBlaster traffic that we observe is coming from France and the Netherlands. However, it does not totally explain the decrease of attacks from Australia. A closer look at the data indicates that this phenomenon is due to the slow down of the traffic to ports 80 (CodeRed/Nimda/Nachi variants [13]) and 445 (Win32.Randon [76]). Last but not least, the increase of US attacks cannot be linked to any specific ports sequence. It still requires a deeper analysis at this stage. Finally, we are really surprised by the regularity of these phenomena over the months. It seems to indicate the existence of some very stable processes, but more data from other honeypots are necessary to check if this phenomenon is, or not, a local artifact.

5.2.2 Operating System of the attacker

We have used two utilities, respectively Disco ([2, 21]) and p0f ([55]) to passively fingerprint TCP packets[4]. They both have similar characteristics but give slightly different results: all non-determined OSs from Disco are classified as Windows by p0f. Our results are presented in Figure 3. With no real surprise, we find out that the majority of attacks originate from Windows machines.

[4] Active fingerprinting techniques such as Nmap [51], Quezo [56], or Xprobe [77] have not been considered to minimize the risk of alerting the attacker of our investigations.

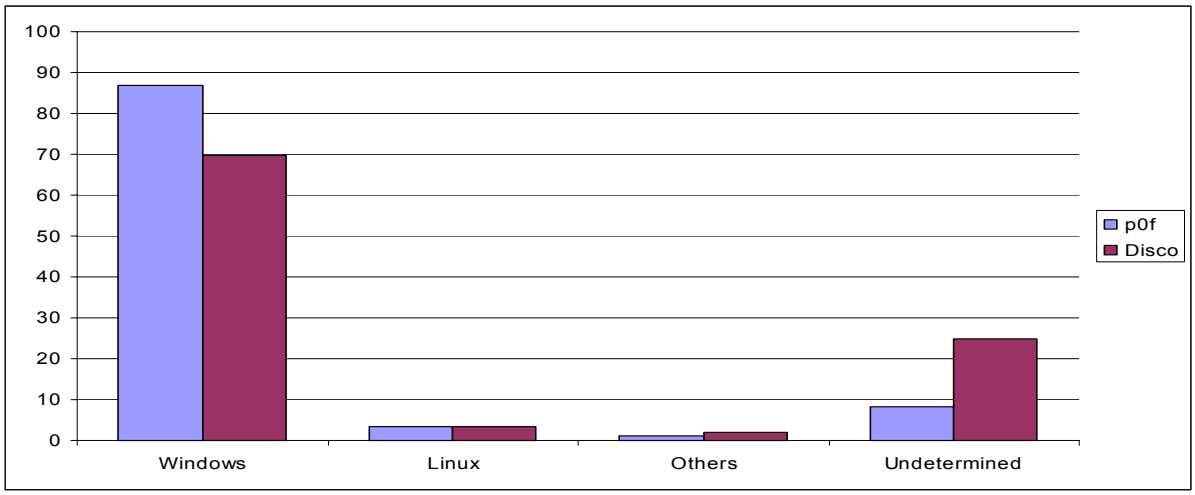


Figure 3: % of diff. IP per OS type with Disco and p0f

5.2.3 Timing of the attacks

We have shown so far that the majority of the attackers were coming from three countries and that this phenomenon was constant over the ten month observation period. Moreover, passive fingerprinting confirms that attacks originate mostly from Windows machines. Additionally, we explain in Section 5.3.1 that attack sequences are limited and very repetitive. Thus, it seems to confirm the classical assumption which claims that the surge in attacks is due to compromised personal computers running automated robots. To validate this claim, we have represented in Figure 4 the percentage of IP addresses observed per hour per country (time is the local time of the attacking source) For instance, point A in that Figure indicates that 4.5% of the attacks coming from Australia occurred between 4 a.m. and 5 a.m. (Australian local time). We have not represented all attacks from the USA, but, instead, only those coming from Virginia and California. These two states account for almost 70% of the American observed attack sources. Each of them is within a single time zone.

There are three important things that are worth being noted in this Figure:

- Attacks are launched on 24x7 basis. This confirms the idea that robots are attacking continuously
- All curves indicate a slight increase of the attack during the late afternoon and in the evening. This is very clear for the attacks originating from France. This indicates that, on top of the attacks running during the whole day, there is another set of attacks that is somehow linked with the activity of human beings.
- The curves of the attacks originating from Virginia and California have a very weird, but similar, shape. They follow the same trends described here after but they are less regular. More interestingly, they seem to move from high to low values at the same rate. So far, we have no good explanation for this unexpected similarity between these curves. It might just be a coincidence or, at the contrary, the expression of a specific attack process running at well defined time intervals.

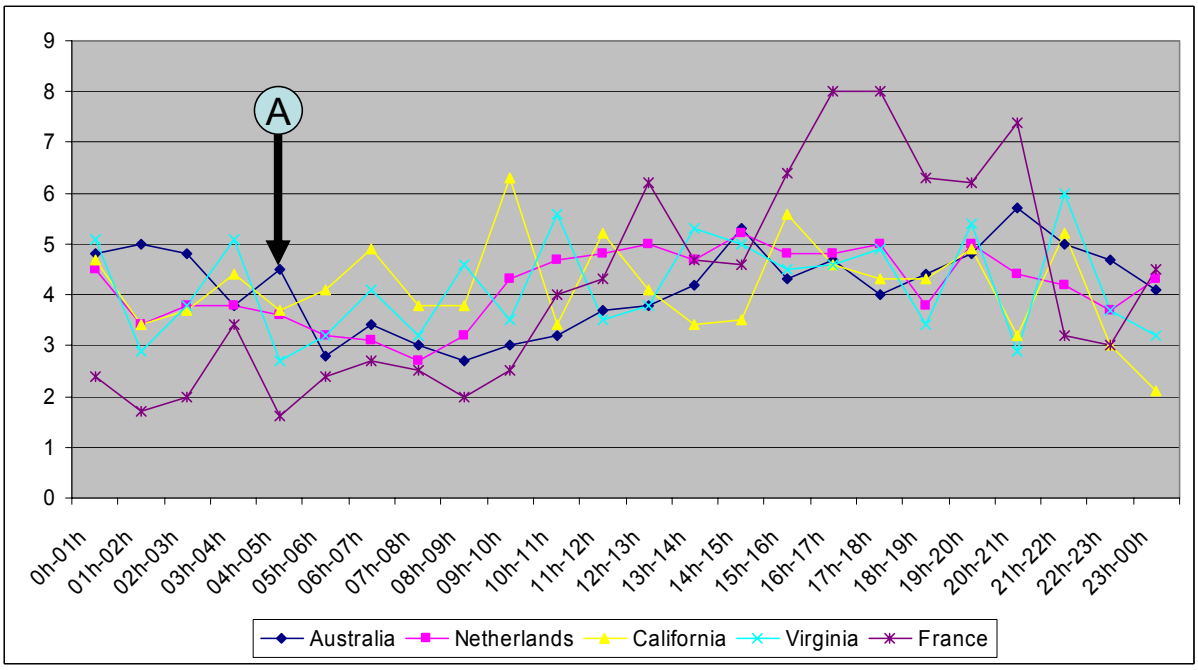


Figure 4: % of IP sources geography distribution per hour and per location

5.3 Details on targeted ports

5.3.1 Generalities

188 different ports have been probed. Each attacking machine probes one or more targeted ports following a given *ports sequence* (see 5.1). The number of different observed sequences is limited to 470 distinct sequences. Furthermore, we note that i) each sequence is often limited to one port and ii) a given set of ports almost always uniquely identifies a *ports sequence*, as we have very rarely observed two sequences differing only by the order of ports being probed. Of course, two attackers can scan same ports for different purposes. A closer inspection of the data payload indicates that the same *ports sequences* often also correspond to similar sequences of packets: in transport level (flags fields) and in the application layer (similar requests/data). In other words, each sequence could potentially be linked to a small number of available attack tools/robots. Some examples are described in Section 5.4, concerning sequences {135, 4444} and {554}.

Table 1 represents, per month, the top 8 *ports sequences* performed by the attack sources observed during that month. For instance, one can see that in March, 45.5% of the attack sources have sent packets to the sole port 445, while 2.7% have scanned ports 80, 57 and 21 in that precise order. We notice that there is a slow evolution over time. These sequences characterize the activity of around 75% of the attack sources every month. Even if in every month we observe the same set of traditional scanned ports (i.e. 21, 80, 135, 139, 445, and 1433), there are a few other cases, such as ports 443, 554, 4444, 17300 and 27374 that are only present in some specific months. These ports correspond to:

- port 443: https, http protocol over TLS/SSL
- port 554: Real-time streaming Protocol
- port 4444: Kerberos authentication (see Section 5.3.2)
- port 17300: Kuang2 Trojan
- port 27374: Ramen linux Worm - SubSeven Trojan

March	April	May	June
445 (45.5%) 80 (15.9%) 1433 (8.8%) 139 (5.8%) 21 (3.5%) 135 (2.9%) 80, 57, 21 (2.7%) 443 (2.1%) Others (12.8%)	445 (43.5%) 80 (15.4%) 1433 (7.6%) 139 (7.5%) 21 (3.3%) 135 (2.2%) 80, 57, 21 (1.4%) 443 (1.4%) Others (17.7%)	445 (40.1%) 80 (15%) 1433 (8.8%) 139 (7.1%) 21 (3.3%) 135 (3.2%) 80, 57, 21 (2.2%) 139, 445 (2%) Others (18.3%)	445 (30.6%) 80 (15%) 139 (9.9%) 1433 (8.2%) 445,139 (4.4%) 139, 445 (3.7%) 21 (3.3%) 135 (3.2%) Others (21.7%)
July	**August**	**September**	**October**
445 (29.5%) 80 (20.5%) 1433 (9.6%) 139 (8.1%) 21 (3.1%) 139, 445 (2.9%) 135 (2.5%) 443 (1.6%) Others (22.2%)	445 (23.6%) 80 (17%) 139 (11%) 1433 (9.5%) 135 (5.8%) 139, 445 (4.1%) 17300 (3.4%) 21 (3.2%) Others (22.4%)	80 (15.8%) 1433 (12.6%) 445 (12.6%) 139 (11.6%) 135 (7.8%) 554 (5.1%) 139,445 (4.2%) 21 (4.2%) 135, 4444 (3.8%) Others (22.3%)	80 (15.6%) 1433 (11.9%) 139 (10.6%) 135 (9.8%) 445 (8.4%) 27374 (6%) 139,445 (5.2%) 135, 4444 (5.2%) 21 (4.4%) Others (22.9%)
November	**December**		
135 (28.9%) 135, 4444 (13.6%) 80 (9.2%) 1433 (8.7%) 139 (8%) 445 (6.4%) 21 (3.5%) 554 (2.9%) Others (20.8%)	135 (28.4%) 135, 4444 (17.3%) 80 (9.4%) 1433 (8.5%) 139 (8.4%) 445 (5.4%) 139, 445 (2.6%) 21 (2.1%) Others	**Others ≈** 185 distinct sequences each month	

Table 1: % of *ports sequences* per month

Port 17300 was mainly scanned during June, July and August, while scanning phases on ports 554 and 4444 have been virulent since September only. This phenomenon is illustrated in figures 5 and 6 and explained in Section 5.3.2 and 5.3.3.

Moreover, a deeper look at attacks on port 139 reveals that there is an increasing number of attacks to TCP port 139 associated with UDP port 137 (in terms of packet numbers). They are due to an increasing number of scans to netbios services. We are convinced they are linked to the recent published vulnerabilities on those two ports: one from March 2003 (CERT CA-2003-08), and one from September 2003 (CERT CA-2003-23) [10].

To conclude, we were expecting to find peaks of activities against specific ports as a result of the publication of a new attack tool is published in underground mailing lists. As explained here above, this is the case. However, we note from Table 1 that the phenomena can be lost in the noise and take some time to become very visible. Figure 5 shows the evolution of the sequence {135, 4444} over the 10 months. There is a similar evolution with the simple sequence {554}, as illustrated in Figure 6. Both sequences are discussed in the two following sections.

5.3.2 Worm observation

This sequence {135, 4444} can be attributed to the proliferation of the Blaster worm.

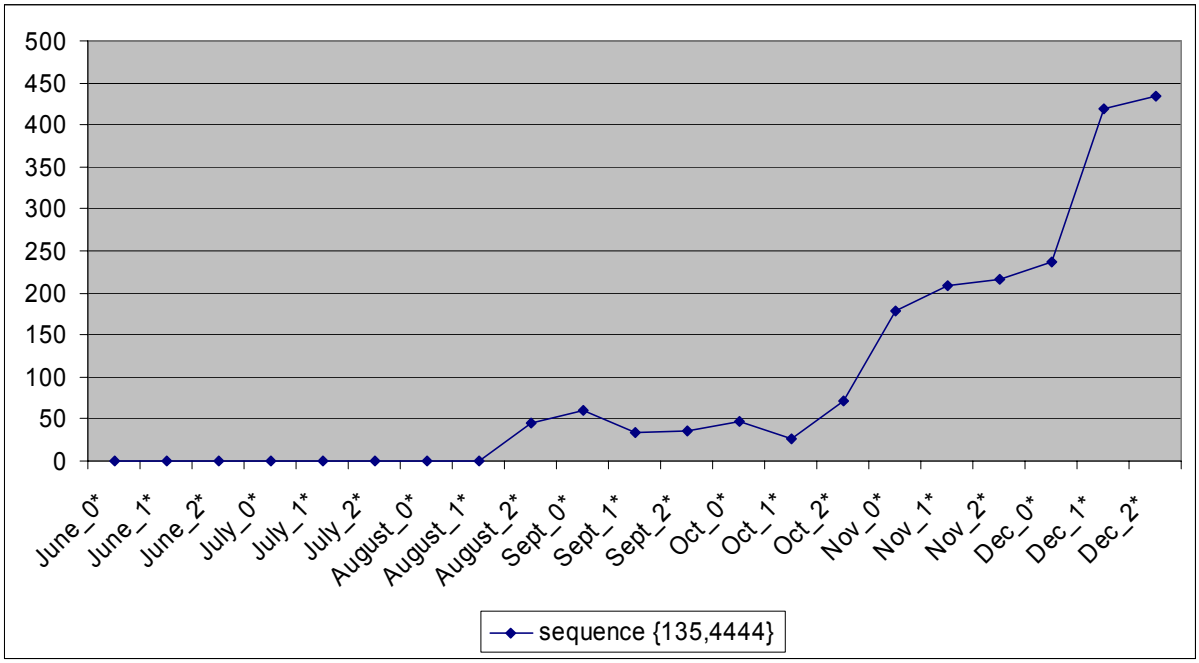


Figure 5: Number of sources scanning ports sequence {135, 4444}

This, by now, famous worm, is named W32.Blaster worm but also "msblast.exe", "Lovscan", "Poza", "Exploit-DcomRpc", etc. TCP port 4444 is normally used for Kerberos authentication and oracle9i communication. A host infected with the W32.Blaster worm opens a command shell on this port, allowing the machine to be remotely controlled. This worm exploits a vulnerability previously disclosed by Microsoft, details of which can be found at [43]. Figure 5 indicates that this worm has been launched on the early days of August 2003 [68]. In our environment, this sequence has only been observed on Mach1. Other machines were never scanned on port 4444, but on port 135 only. This is due to the fact that the worm first scans ports 135 and never goes further if the port is closed. Mach0 is a windows 98 workstation where port 135 is closed. Mach3 is a Linux RedHat server that has by default its port 135 closed. In a more general way, the major part of observations which were published about MBlaster are easily checked and validated in our environment [22, 68]. Finally, we want to point out the gap that exists between public statements on some attacks on one side and concrete network activity on the other side. For instance, Symantec has downgraded the MBlaster worm threat on the 8th of October 2003, *due to a decreased rate of submissions* [68]. However, Figure 5 clearly shows that the epidemic is still expending.

5.3.3 Scanning tools

There is no well-identified worm that produces traffic on port 554. However, a new RTSP scanner has been released on August 2003: this utility, freely available at [57] looks for "Real Server" vulnerabilities (version 7, 8 and 9 on Windows and Linux) to obtain remote shell with root privileges. The tool was publicly released on the 28th of August, while the first scan appeared on the 26th of the same month. This indicates that it has been tested in the wild by its authors before being published.

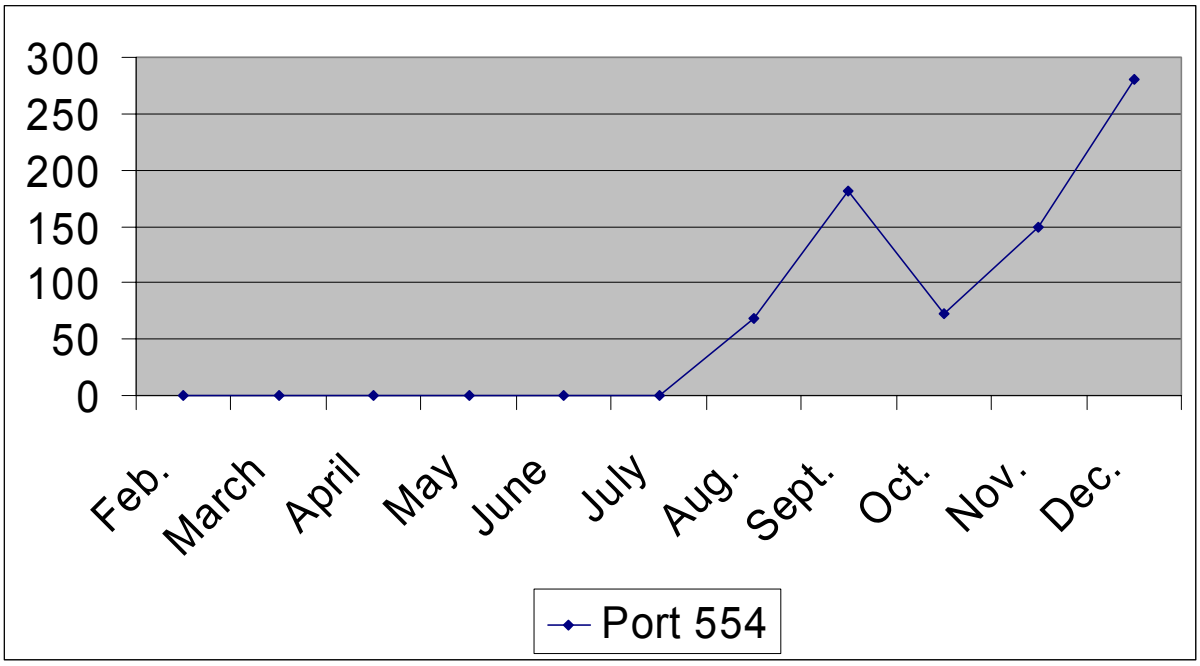


Figure 6: Number of sources scanning ports sequence {554}

5.4 Analysis of targeted machines

5.4.1 Targeted machines: the attacks distribution

We distinguish two kinds of attacking sources: those that have sent packets to all our honeypots, and the others that have only been observed by a subset of our honeypots. We focus in this part on the first set (called '*attacks of type I'* in the following). The next section is dedicated to the other set. First and foremost, whenever a source attacks our three honeypots, we observe that the ports sequence is always the same, and so is the honeypots sequence, namely mach0, mach1 and then mach2. We have never observed any other honeypots sequence during the 10 month period. Furthermore, port sequences are also very limited. To make things shorter, the three honeypots are targeted similarly, independently of their Operating System or offered services. A deeper look at the packets shows that most of sequences have several common characteristic patterns. For instance almost all requests on port 80 observed from July to December 2003 consist in the following request: 'GET /scripts/nsiislog.dll' (98% of http requests). This corresponds to a Microsoft Remote Buffer overflow vulnerability published in June 25th 2003 (CVE CAN-2003-0349) [17]. In the same way, we note that a very few different commands have been issued after a successful login to our ftp server. Those given in Table 2 represent 99% of the ftp connection attempts. Other observed attempts (1% of all ftp attempts) are either hand-written ones (assumption based on the inter-request time interval) or dictionary attacks.

USER	PASS	Other commands
anonymous anonymous@ftp.adobe.com anonymous@ftp.microsoft.com:21 anonymous@ftp.microsoft.com	_gpuser@home.com abc@126.com guest@here.com me@here.com ano@ano.com	TYPE I PORT 80, 180, 24, 202, 7, _ RETR /products/mediaplayer/unix/netshow_su

Table 2: Frequent observed FTP commands

Searches on the underground sites of blackhats communities provide the name of associated tools (Grim's Ping public ftp scanning tool, Roadkil's FTP Probe, and so on). Some specific Unix attacks are applied against all machines, no matter what OS they run. Other tools are issuing ftp commands while login connections have failed. Thus, many characteristics show that we are facing simplistic robots and that their number is rather limited. People using them do not even take the time to change the tool fingerprinting characteristics.

5.4.2 Attacks focusing on one target only

These attacks happen less frequently (25.5%) than attacks of type I. We distinguish two types of attacks: those which are similar to the ones observed on all machines (type I), and those which seem specific to one machine only (type II). Type I attacks are similar to those described in Section 5.4.1 but the difference lies in the scanning phase. In Section 5.4.1, all machines are scanned sequentially while we observe here a different scanning method. Either one or two machines are scanned. Thus, the involved tools have different scanning options or versions but they can be recognized as belonging to type I because of the ports sequence. On the other hand, when looking at attack sources sending new sequences against a single of our honeypots, two different situations coexist:

- The address of the honeypot has been spoofed and is used in an ongoing DDoS attack against a third party.
- A focused attack is being tried on one of our honeypot.

In the first case, the observed packets are so-called "backscatter" packets. They are responses to packets from Denial of Service attacks, in which our IP addresses have been used. These attacks are well-analyzed by Moore et al. in [46]. Figure 7 summarizes the various types of responses (column 'response from victim') that can be sent "against" our honeypots. These packets hit a large variety of ports that are traditionally unused, such as 27374 (TCP RST), 11224 (TCP SYN ACK), 9026 (RST ACK), etc...

Packet sent	Response from victim
TCP SYN (to open port)	TCP SYN/ACK
TCP SYN (to closed port)	TCP RST (ACK)
TCP ACK	TCP RST (ACK)
TCP DATA	TCP RST (ACK)
TCP RST	no response
TCP NULL	TCP RST (ACK)
ICMP ECHO Request	ICMP Echo Reply
ICMP TS Request	ICMP TS Reply
UDP pkt (to open port)	protocol dependent
UDP pkt (to closed port)	ICMP Port Unreach
...	...

Figure 7: Some victim responses to flooding attacks [46]

Type II attacks are quite well identifiable: they correspond to 36% of the sources that send packets to only one of our honeypots. We are more concerned about the other category: attacks which target specific services on our machines. In these situations, the machines have never been seen doing a port scan. On the contrary, and much to our surprise, they are always, systematically, sending requests to ports that are open on the machine they are communicating with. Statistically speaking, it is very unlikely to see this phenomenon. It shows that those machines benefit from the results of the scans performed by other machines. In order to highlight this problem, we have opened four new ports on Mach2 on mid-October: one is a well-known Microsoft service, named MS SQL (port 1433), and others are old Trojans (ports 8998, 28934 and 54321). The results we obtain for port sequence {1433} are given in Figure 8. It represents the number of sources that targeted only port 1433 on that sole machine. None of these IP addresses had been observed before. Other ports give similar results.

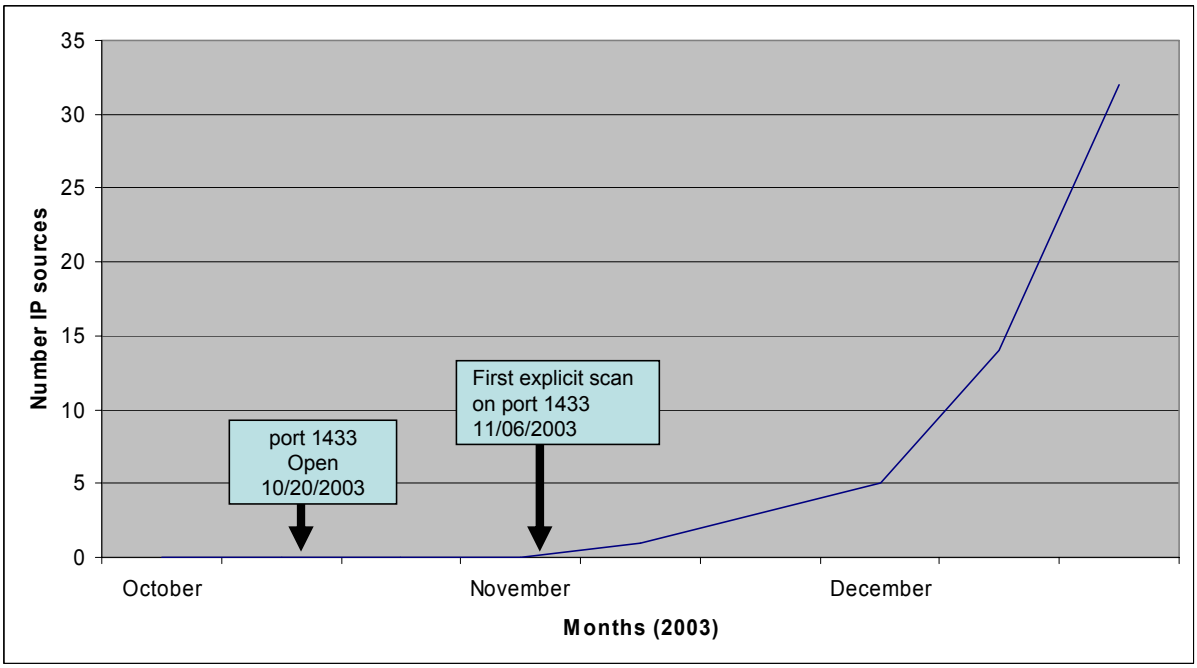


Figure 8: Number of IP sources targeting mach2 only and on port 1433

Figure 8 reveals that it takes 17 days for the first precise attack to happen. Also, it is worth pointing out that port 1433 has been opened on a Linux machine whereas an attack against the service running on that port only exists for Windows machines. Thus, this indicates that scanning machines provide some basic information regarding the opened ports but fail in fingerprinting the OS of the machine they have probed.

6.0 CONCLUSION

In this paper, we have presented data obtained by means of honeypots being attacked over a period of ten months. We have shown the interest a honeypot environment can bring to the network analyst. Due to space limitations, we did not detail to much our analyses but, instead, have focused on the variety of information which can be obtained this way.

We have described the information gathered regarding the attackers and their attack processes. Some results are quite surprising, such as the regularity of the attacking behaviors. We postulate that this information is complementary to what can be obtained through 'Internet telescopes' and vulnerability lists [44, 58].

Our current results have opened issues for further research. Some questions have been left unanswered. The unexpected behavior of machines knowing our environment setup without having probed it and the increase of attacks coming from the USA are two examples of issues that must be clarified. In order to compare local results, new dynamic environments must be designed, not only to find out how long it takes for new information to be collected and shared, but also to figure out if we are facing one or several populations of collaborating attackers.

7.0 BIBLIOGRAPHY

[1] H. AbdelallahElhadj, H. M. Khelalfa and H. M. Kortebi, "An experimental sniffer detector: SnifferWall", *SEcurité des Communications sur Internet Workshop (SECI'02)*, Tunisia, Sept. 2002. http://www.lsv.ens-cachan.fr/~goubault/SECI-02/Final/actes-seci02/pdf/008-Abdelallahelhadj.pdf

[2] O. Arkin, "*ICMP Usage in Scanning - The complete Know-How*", The Sys-Security Group. Version 3.0. June 2001. http://www.sys-security.com

[3] *Back Officer Friendly* home page: http://www.nfr.com/resource/backOfficer.php

[4] *Bait N Switch Honeypot* home page: http://violating.us/projects/baitnswitch/

[5] P. Barford, J. Kline, D. Plonka, A. Ron, "A Signal Analysis of Network Traffic Anomalies", *Internet Measurement Workshop 2002*.; www.icir.org/vern/imw-2002/imw2002-papers/173.pdf

[6] S. Bellovin, "There Be Dragons", *Proc. of the Third Usenix Security Symposium*, Baltimore MD. Sept. 1992. Available on line: http://www.research.att.com/~smb/papers/dragon.pdf

[7] S. M. Bellovin, "Packets Found on an Internet", *Computer Communications Review* 23:3, pp. 26-31, July 1993. Available on line: http://www.research.att.com/~smb/papers/packets.pdf

[8] *Bigeye* home page: http://violating.us/projects/bigeye/

[9] *The sleuth kit V1.62* (previously known as TASK), Brian Carrier, 2003, www.sleuthkit.org/

[10] CERT @Vulnerabilities advisories home page: http://www.cert.org/advisories/

[11] Z. Chen, L. Gao, K. Kwiat, "Modeling the Spread of Active Worms", *IEEE- INFOCOM 2003*, April 2003, San Francisco. Available on line: http://www.ieee-infocom.org/2003/papers/46_03.PDF

[12] B. Cheswick, "An evening with Berferd in which a cracker is lured, endured and studied", *Proc Winter USENIX Conference*, San Francisco, Jan 20, 1992.

[13] Some CodeRed statistics available at: http://kropf.net/coderedstats.html

[14] *Deception Tool Kit*, DTK, Fred Cohen & Associates. http://www.all.net/dtk/dtk.html

[15] F. Cohen, D. Lambert, C. Preston, N. Berry, C. Stewart and E. Thomas, "*A Framework for Deception*", Tech. Report, July 2001, http://all.net/journal/deception/Framework/Framework.html

[16] P. Cracknell, "*The wireless honeypot project: A brief look at how wireless networks are used and misused in the City of London*", RSA Security UK Limited (RSA SUL), CISSP, Tech. Report. http://www.rsasecurity.com/worldwide/downloads/honeypot_report2003.pdf

[17] Common Vulnerabilities and Exposures (CVE) home page: http://www.cve.mitre.org/

[18] M. Dacier, F. Pouget and H. Debar, "Honeypots: Practical Means to Validate Malicious Fault Assumptions", *Proc. Of the 10th Pacific Ream Dependable Computing Conference (PRDC04)*, February 2004.

[19] *Symantec Decoy Server*, enterprisesecurity.symantec.com/products/products.cfm?ProductID=157

[20] H. Debar and D. Lefranc, "Observations on the Internet traffic reaching broadband-connected users", *EICAR Conference*, Copenhagen, Mai 2003.

[21] *The Disco tool* home page: http:// www.altmode.com/disco

[22] eEye Digital Security Research home page: http://www.eeye.com/html/Research/

[23] J. Evers, "*Experts: Most Code Red attacks coming from Asia*", IDG News Service, 2001. Available on line: www.computerworld.com/securitytopics/security/story/0,10801,62730,00.html

[24] FakeAP, par Black Alchemy, home page: http://www.blackalchemy.to/project/fakeap/

[25] Coroner toolkit, D. Farmer, W. Venema, home page: http://www.porcupine.org/forensics/tct.html

[26] S. Grundschober, "*Sniffer Detector Report*", ,Internship Report from IBM Zurich for the Eurecom Institute , June 1998, 50 pages, ref. Eurecom : CE-98/IBM/GRUN - Document number: 1914. Available on line: http://www.eurecom.fr/~nsteam/Papers/grundschober98.ps

[27] S. Grundschober, M. Dacier, "Design and Implementation of a Sniffer Detector", *Recent Advances on Intrusion Detection Workshop (RAID98)*, 1998. www.raid-symposium.org/raid98/

[28] Irish Honeynet Alliance members. Collected data available on line: www.honeynet.ie/results.htm

[29] *Honeypot mailing list*, available at: www.securityfocus.com/popups/forums/honeypots/

[30] "*Know Your Enemy: Statistics Analyzing the past ... predicting the future*", Honeynet Project, July 2001. Available on line: http://project.honeynet.org/papers/stats

[31] "*Know Your Enemy: Learning with User-Mode Linux Building Virtual Honeynets using UML*", Honeynet Project, December 2002. Available on line: http://www.honeynet.org/papers/uml/

[32] "*Know Your Enemy: GenII Honeynets Easier to deploy, harder to detect, safer to maintain*", by the honeynet Project members, June 2003. Available on line: http://project.honeynet.org/papers/gen2/

[33] Honeyweb download page, http://www.var-log.com/files/

[34] *IEEE INFOCOM, April 2003*, San Francisco, www.ieee-infocom.org/2003/technical_programs.htm

[35] *Internet Management Workshop 2001*, home page: http://www.icir.org/vern/imw-2001/

[36] *Internet Management Workshop 2002*, home page: http://www.icir.org/vern/imw-2002/

[37] *Internet Management Conference 2003*, home page: http://www.icir.org/vern/imc-2003/

[38] *Internet Storm Center*, home page: http://isc.incidents.org/

[39] E. Jacksch, "*Tenebris Wireless Honeypot Project: Assessing the threat against wireless access points.1.0*", CISSP, Tenebris Technologies Inc, 2002. www.tenebris.ca/docs/TWHP20021119.pdf

[40] KFSensor, by Keyfocus, home page http://www.keyfocus.net/kfsensor/

[41] Labrea Tarpit Project, http://labrea.sourceforge.net/

[42] MaxMind GeoIP utility home page: http://www.maxmind.com/app/home

[43] Microsoft vulnerability announcement: http://www.microsoft.com/technet/security/bulletin/MS03-026.asp

[44] D. Moore, C. Shannon, K. Claffy, "*Code-Red: a case study on the spread and victims of an Internet worm*", Internet Measurement Workshop 2002, www.icir.org/vern/imw-2002/imw2002-papers/209.ps.gz

[45] D. Moore, G. Voelker et S. Savage. "*Inferring Internet Denial-of-Service Activity*", 2001 USENIX Sec. Symp. www.caida.org/outreach/papers/2001/BackScatter/usenixsecurity01.pdf

[46] *Netbait* home page http://www.netbaitinc.com/

[47] *NetFacade intrusion detection service,* Verizon utility homepage: www22.verizon.com/fns/netsec/fns_netsecurity_netfacade.html

[48] *Netgeo Utility*, available online at http://netgeo.caida.org/perl/netgeo.cgi

[49] *Specter 7.0*, by Netsec, home page: http://www.netsec.ch/

[50] A. Neville, "*IDS Logs in Forensics Investigations: An Analysis of a Compromised Honeypot*", March 2003. Available on line: http://www.securityfocus.com/infocus/1676

[51] "The Art of Port Scanning". *Phrack Magazine* Volume 7, 1997, article 11 (www.nmap.org)

[52] F. Pouget, M. Dacier, H. Debar. "*Honeypots: a comparative survey*", Eurecom Report, RR-03-81, July 2003.

[53] "*Conceptual Model and Architecture of MAFTIA"*, D. Powell et R. Stroud (Editors), MAFTIA Project (IST-1999-11583), Deliverable D21, January 2003; available on line at www.maftia.org.

[54] *Honeyd* Home page, Niels Provos, http://www.citi.umich.edu/u/provos/honeyd/

[55] p0f passive fingerprinting tool home page: http://lcamtuf.coredump.cx/p0f-beta.tgz

[56] *The Quezo tool*, home page: http://www.apostols.org/projectz/queso/

[57] RTSP Scanner available at: http://iperl.homelinux.org/haxor/scanner.pl]

[58] The SANS Security Institute, home page: http://www.sans.org

[59] K. Seifried, "*Honeypotting with VMware – basics*". www.seifried.org/security/ids/20020107-honeypot-vmware-basics.html

[60] *SIGCOMM 2003 Conference*, August 2003, http://www.acm.org/sigcomm/sigcomm2003/

[61] *Single Honeypot*, home page http://sourceforge.net/projects/single-honeypot/

[62] *Smoke Detector*, product home page http://palisadesys.com/products/smokedetector/index.shtml

[63] D. Song, R. Malan and R. Stone, "*a global snapshot of internet worm activity*", November 2001. Technical Report, http://research.arbor.net/downloads/snapshot_worm_activity.pdf.

[64] L. Spitzner, "*Honeypots: Tracking Hackers*", Addislon-Wesley, ISBN from-321-10895-7, 2002.

[65] L. Spitzner, "*Honeytokens: The Other Honeypot*", 2003. www.securityfocus.com/infocus/1713

[66] L. Spitzner, "*Specter: a Commercial Honeypot Solution for Windows*", 2003, http://www.securityfocus.com/infocus/1683

[67] C. Stoll, "Stalking the Wiley Hacker", *Communications of the ACM*, Vol. 31 No 5. May 1988.

[68] Symantec MBlaster worm updates, available at : http://securityresponse.symantec.com/avcenter/venc/data/w32.blaster.c.worm.html

[69] *Tcpdump* home page: http://www.tcpdump.org/

[70] *Tiny Honeypot* home page: http://www.alpinista.org/thp/

[71] *User Mode Linux*, UML, home page: http://user-mode-linux.sourceforge.net/

[72] *VMWARE*, User's manual. Version 3.1, home page: http://www.vmware.com

[73] *VSERVER*, home page: http://freshmeat.net/projects/vserver/

[74] "*Warchalking: Collaboratively creating a hobo-language for free wireless networking*", www.warchalking.org/

[75] *WISE*, Wireless Information Security Experiment, http://www.incident-response.org/WISE.htm

[76] W32.randon virus information, available at: http://www.viruslibrary.com/virusinfo/Worm.Win32.Randon.htm

[77] *The Xprobe project*, version 0.0.1 July 13, 2001, http://sourceforge.net/projects/xprobe

[78] "*The museum of broken packet*", M. Zalewski, http://lcamtuf.coredump.cx/mobp/

8.0 ANNEX A

	Interaction Level	Freeware	Emulated Services	Emulated OS	Host OS	Regularly maintained
Bait N Switch Honeypot [4]	Medium	Yes	Malicious traffic is redirected towards non critical targets	None	Linux (on the switch)	Yes
BigEye [8]	Medium	Yes	2 (ftp, http)		Unix	
BOF [3]	Low	No	7 (telnet, ftp, smtp, http, pop3, imap2)		Win 32, Unix	Apparently yes
Decoy Server [19]	High	No	No limit	several	Windows (9x, 2000, NT) Solaris	yes
DTK [14]	Low to Medium	Yes	No limit		Unix	Apparently no
FakeAp [24]	Low	Yes	802.11b	none	Linux	yes
HoneyD [54]	Medium	Yes	No limit	No limit	Unix	yes
HoneyWeb [33]	Medium	Yes	1 (web server)	none	Win32, Unix	Apparently no
KFSensor [40]	Medium	No	No limit	none	Win32	yes
Labrea [41]	Low to Medium	Yes	None	none	Win32s, Linux	no
Netbait [46]	Medium	No	No limit	several		yes
NetFacade [47]	High	No	13	8	Solaris	yes
Single Honeypot [61]	Low	Yes	Smtp, pop3		Linux, Freebsd, Solaris	no
Smoke Detector [62]	High	No	22	9	Windows 2000	
Specter [49], [66]	High	No	14	13	Windows NT, 2000, XP	yes
Tiny HoneyPot [70]	Medium	No			Linux	

Attack Processes found on the Internet

M. Dacier, F. Pouget
{dacier, pouget}@eurecom.fr

H. Debar
herve.debar@francetelecom.com

EURECOM
Sophia Antipolis

Overview

- Introduction :
  honeypots and related work
- Experimental set up
- Results:
  - The attacking machines
  - The attacked machines
  - The attacked ports
- Conclusions:
  - Proposal for a collaborative international set up

Honeypots: definition

"A honeypot is an information system resource whose value lies in unauthorized or illicit use of that resource."

L. Spitzner, "*Honeytokens: The Other Honeypot*", 2003.
www.securityfocus.com/infocus/1713

Honeypots: Historical view

- 1988: Clifford Stoll
  - Cuckoo's Egg: first passive honeytoken
- 1992: Cheswick, Bellovin
  - « An evening with Berferd… »: first honeypot
- 1998: Dacier, Grundschober
  - Sniffer detector: first active honeytoken

Honeypots: usage

- Post mortem Analysis
  - Coroner Toolkit, Sleuth Kit
- Identification of new threats
- Statistical data gathering tool

Honeypots: current focus

- Implementation and architectural issues
  - Mostly done within the Honeynet Research Alliance
- A few anecdotical post mortem analysis
  - Most of that work is done ‘secretly’
- Almost no publication on data analysis

Motivations of our work

- Lack of unbiased information dealing with attacks on the Internet

- Need of such data to justify fault assumptions made when building intrusion tolerant systems.

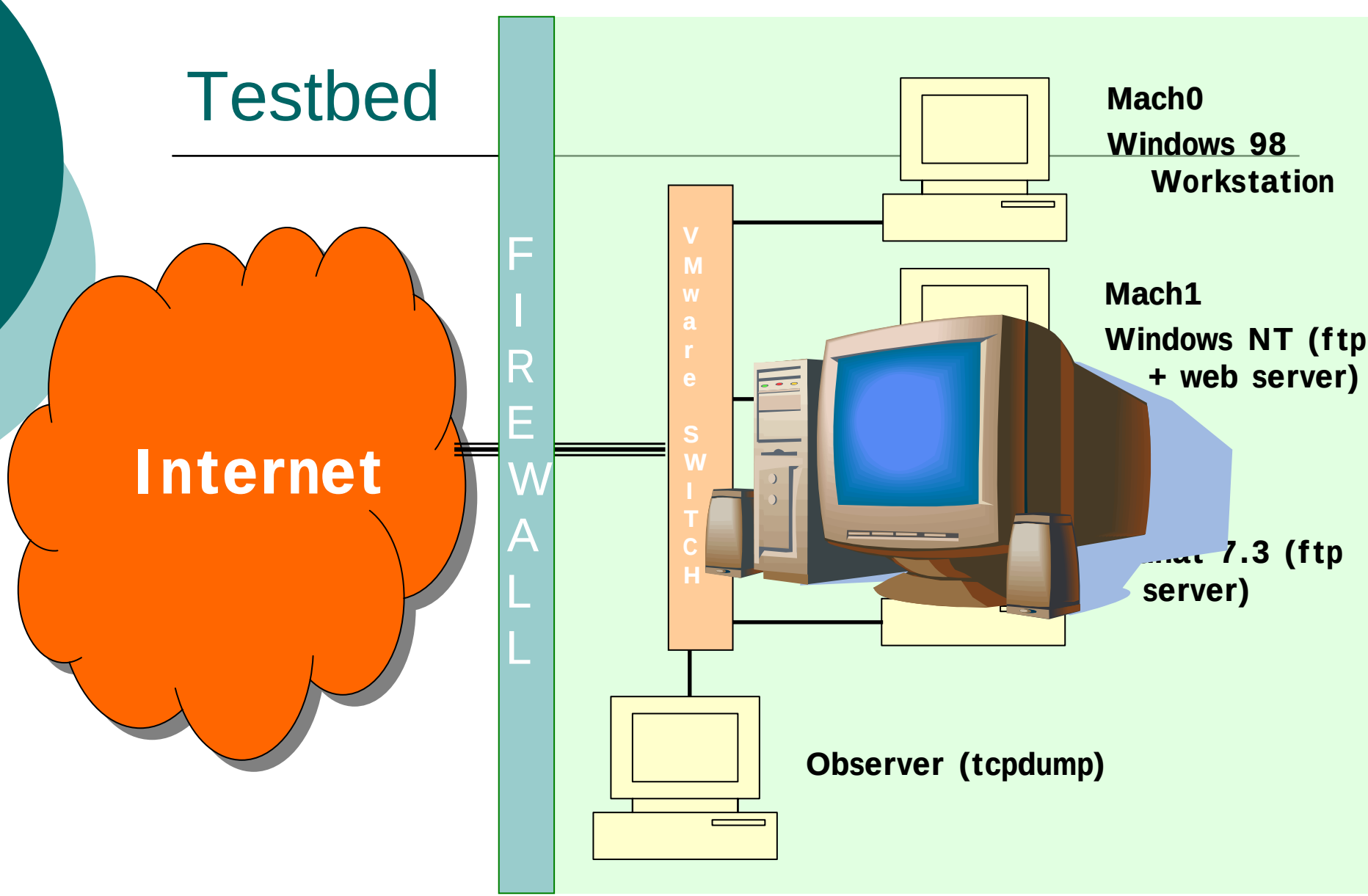
Overview

- Introduction :
  honeypots and related work
- Experimental set up
- Results:
  - The attacking machines
  - The attacked ports
  - The attacked machines
- Conclusions:
  - Proposal for a collaborative international set up

Environmental Set Up

- 3 different technologies for honeypots have been used:
  - The simplest one: a machine with all ports closed, running tcpdump
  - The most common one: a machine running honeyd to emulate services offered by 3 machines on a LAN
  - The most sophisticated one: a machine running VMWare to run 3 machines on a LAN.

Testbed


Internet
FIREWALL
VMware SWITCH
Mach0
Windows 98
Workstation
Mach1
Windows NT (ftp
+ web server)
7.3 (ftp
server)
Observer (tcpdump)

Testbed (ctd.)

- The observer stores the whole content of every packet sent to or from any of our virtual machines
- Tcpdump log files are transferred regularly and securely to another machine
- Data is fed into a MySql DB …
- … and enriched with geographical data and with OS fingerprinting information.

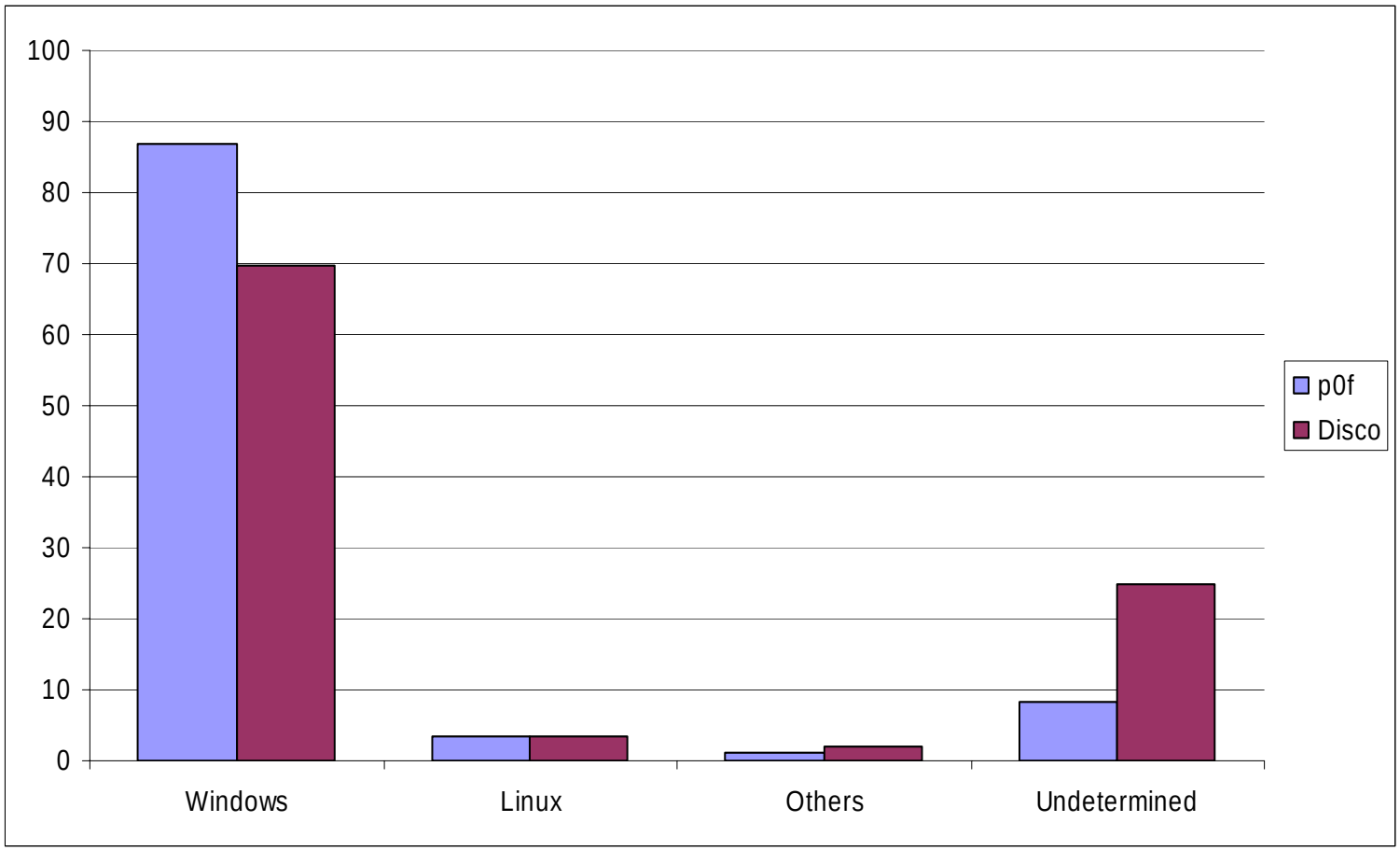
Overview

- Introduction :
  honeypots and related work
- Experimental set up
- Results:
  - The attacking machines
  - The attacked ports
  - The attacked machines
- Conclusions:
  - Proposal for a collaborative international set up

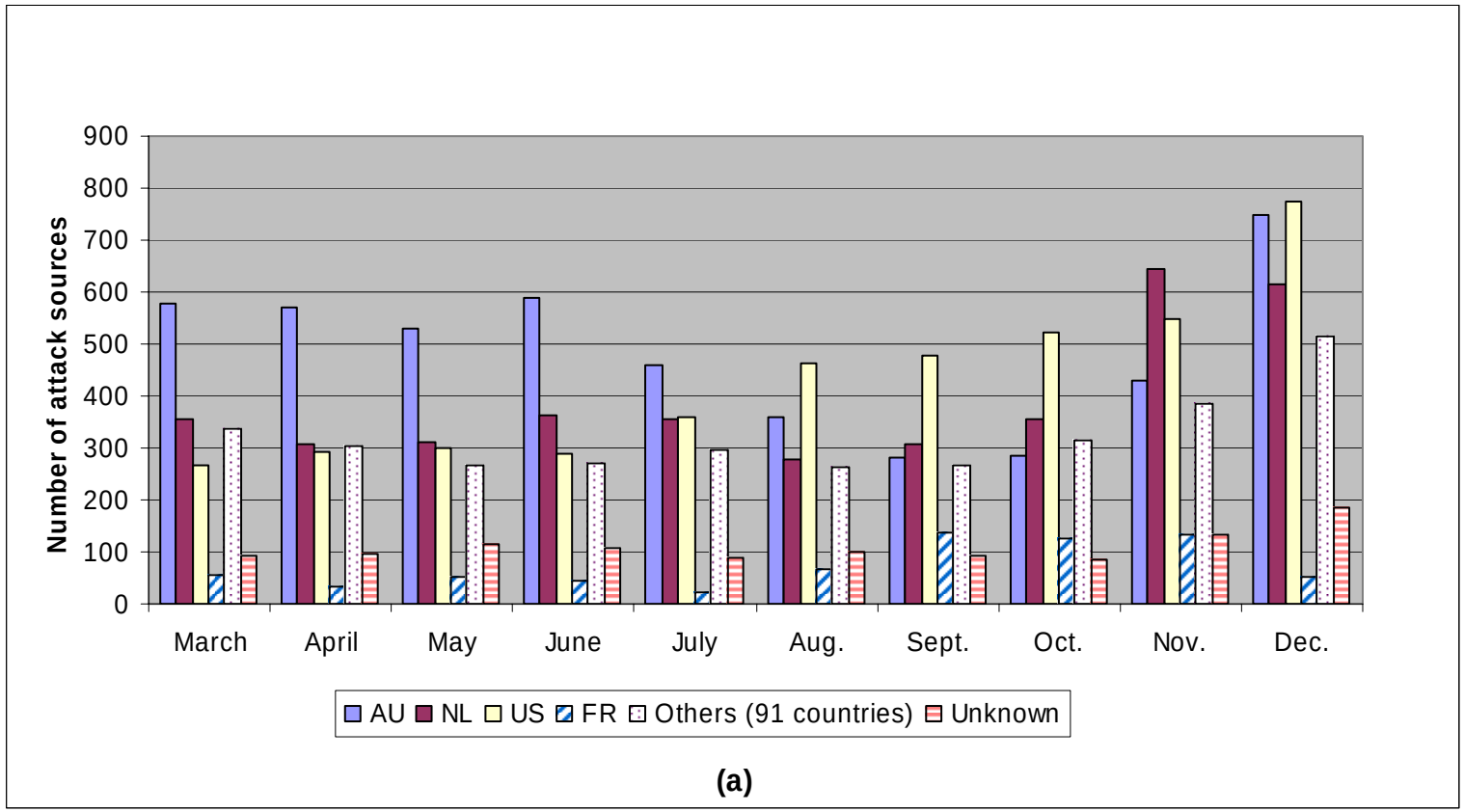
Results: the big picture

- 10 months of data (march-december 2003)
- 3028316 packets from 18075 different IP sources (i.e. 2 new attacking source per hour)
  - TCP 84.7 %, ICMP 1.6 %, UDP 13.7 %.
- 188 different ports have been scanned.
- 69% of the IP addresses have scanned the three honeypots, always in the same order, always in a few seconds.
- 5.5% of the IP addresses have scanned only 2 out of the three honeypots.
- 25.5% of the IP addresses have sent requests to only 1 honeypot.

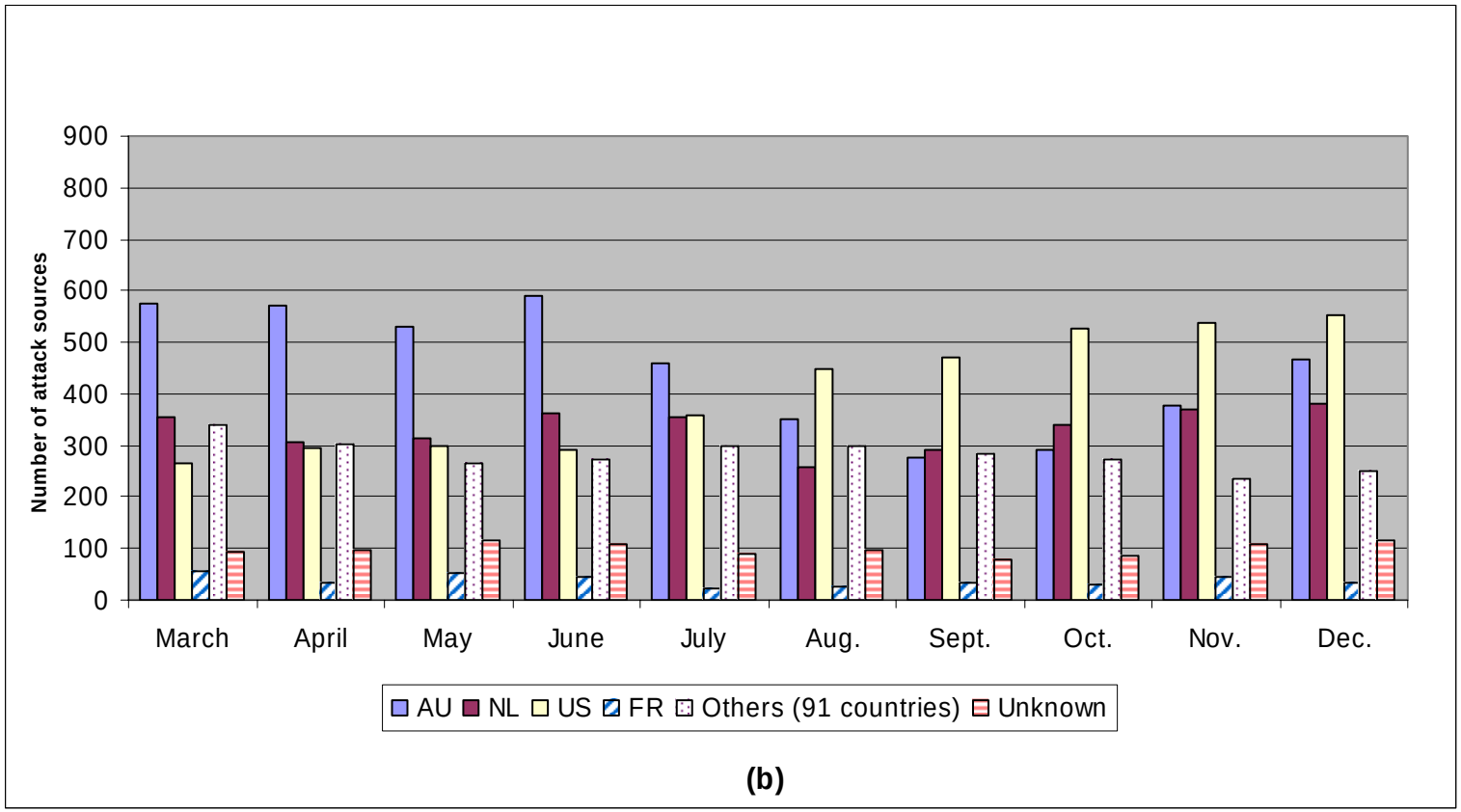
Results: OS of the attacking Machines



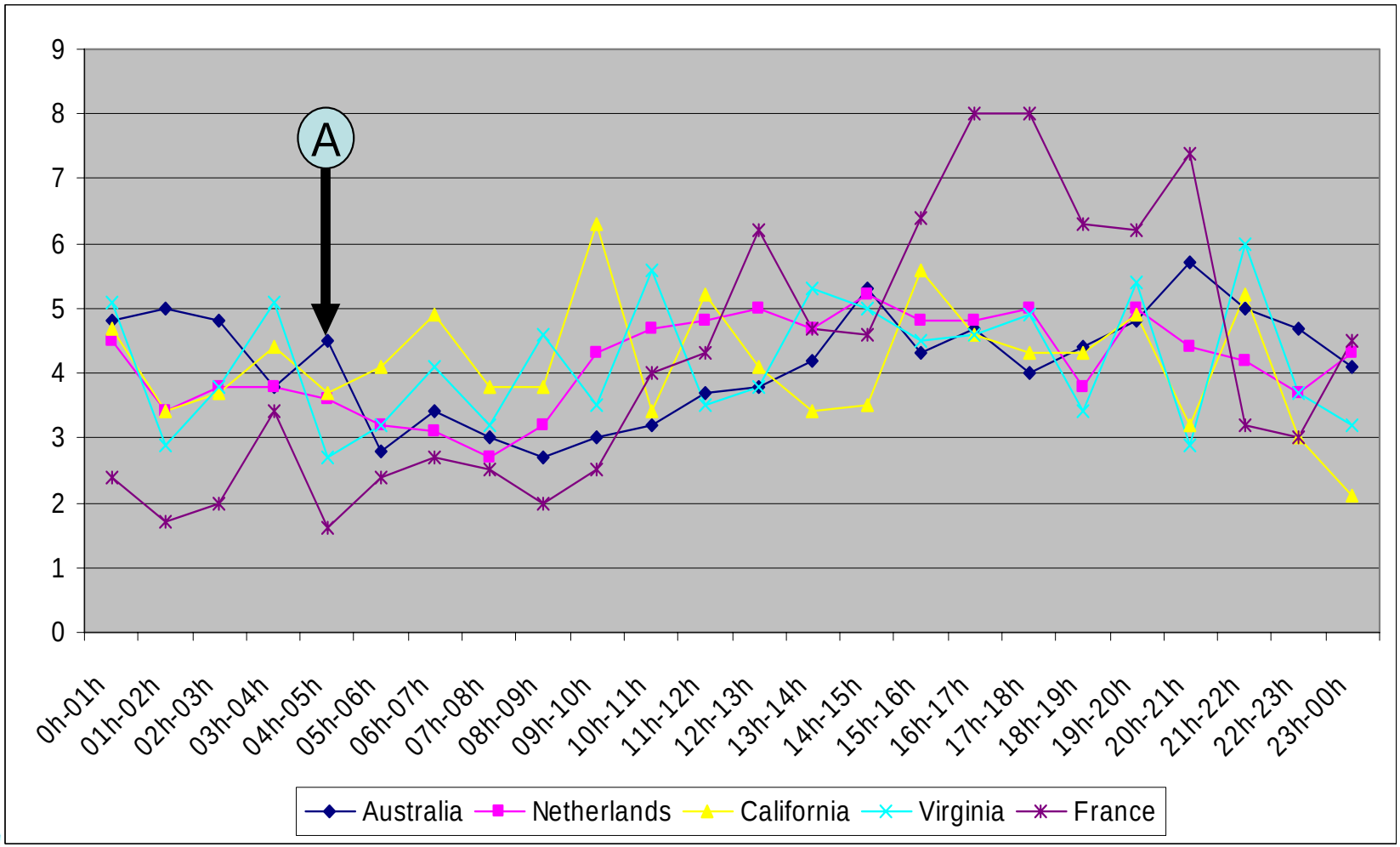
Results: Origin of the attacks (1/2)



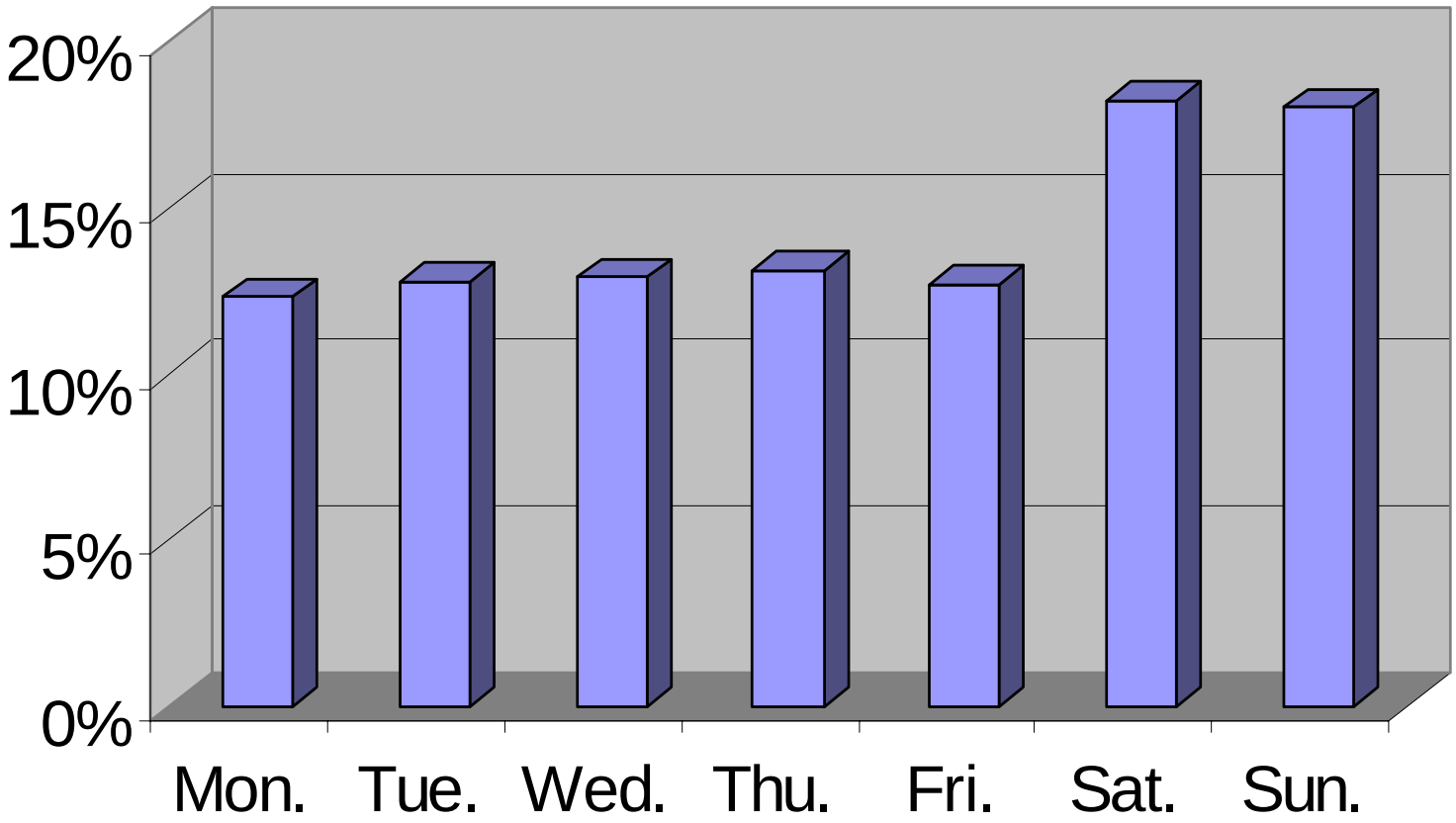
Results: Origin of the attacks (2/2)



Results: Hourly rate of the attacks



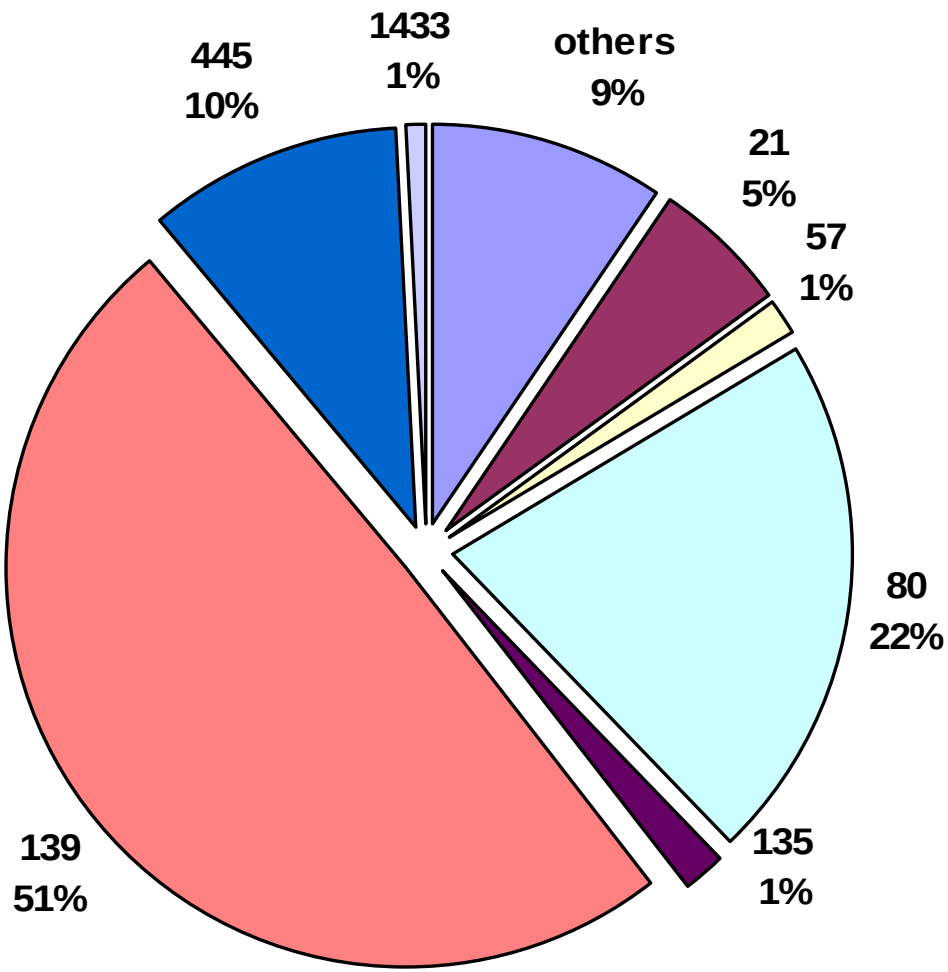
Results: Daily rate of the attacks



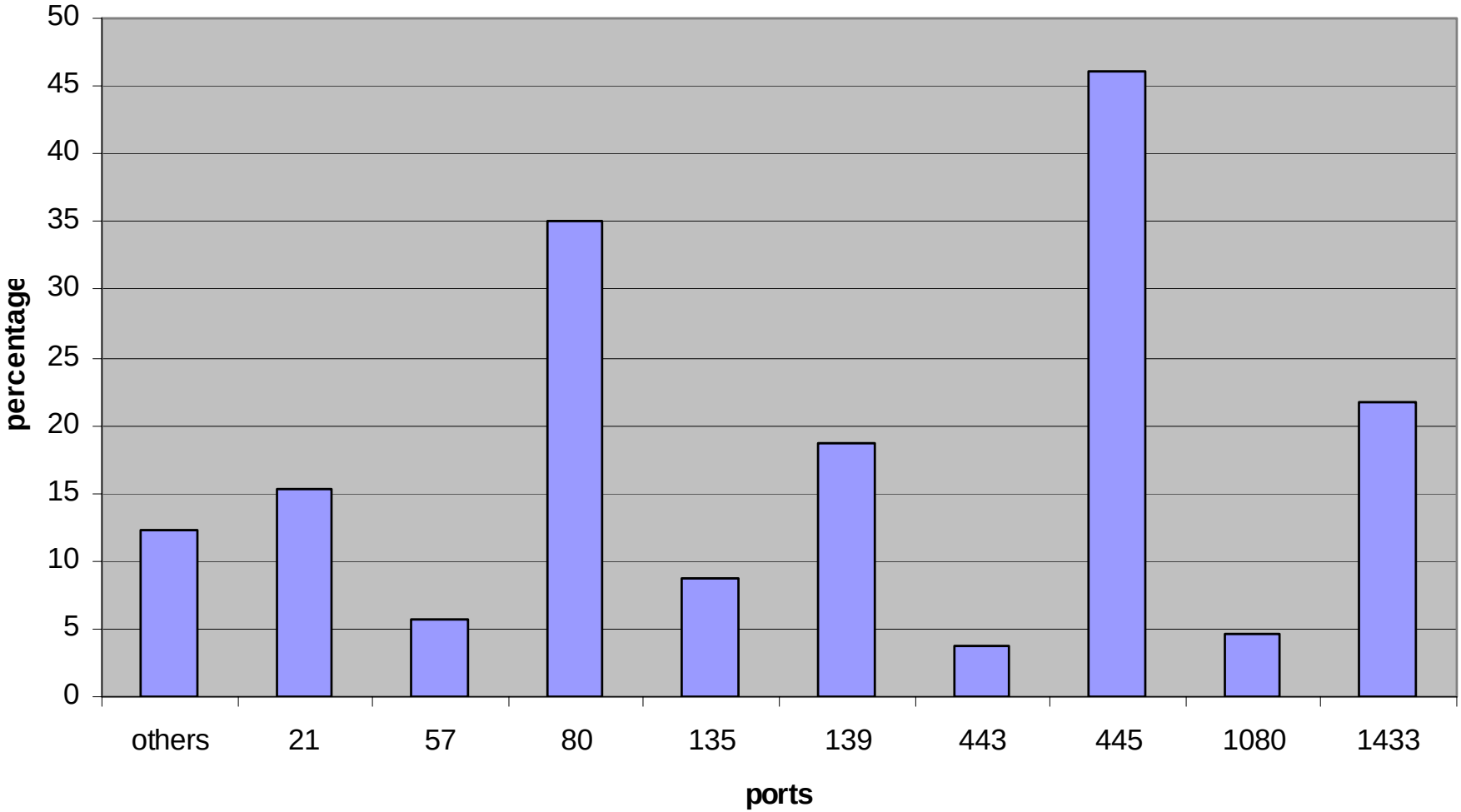
Overview

- Introduction :
  honeypots and related work
- Experimental set up
- Results:
  - The attacking machines
  - The attacked ports
  - The attacked machines
- Conclusions:
  - Proposal for a collaborative international set up

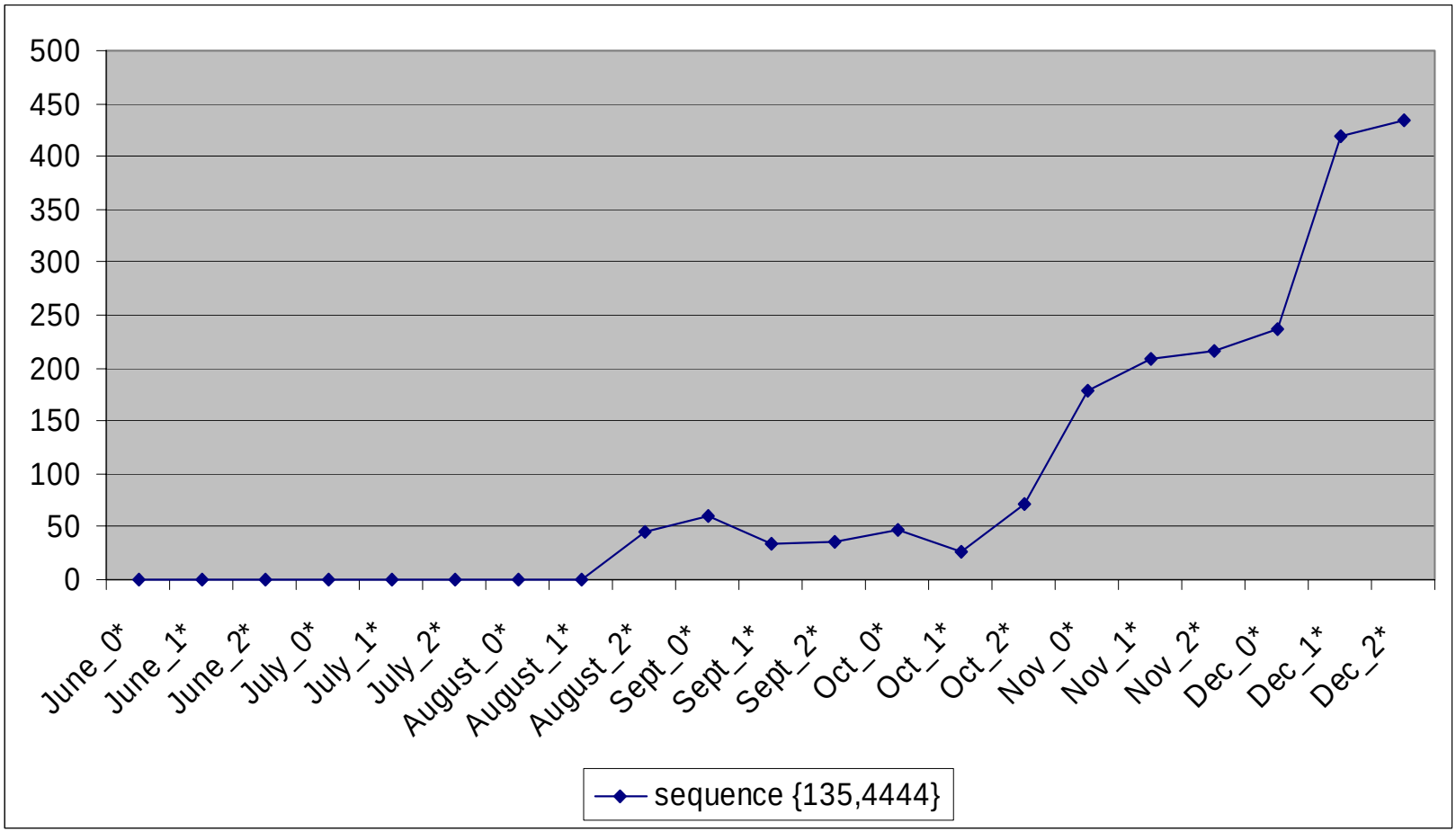
Results: packets per dest. port



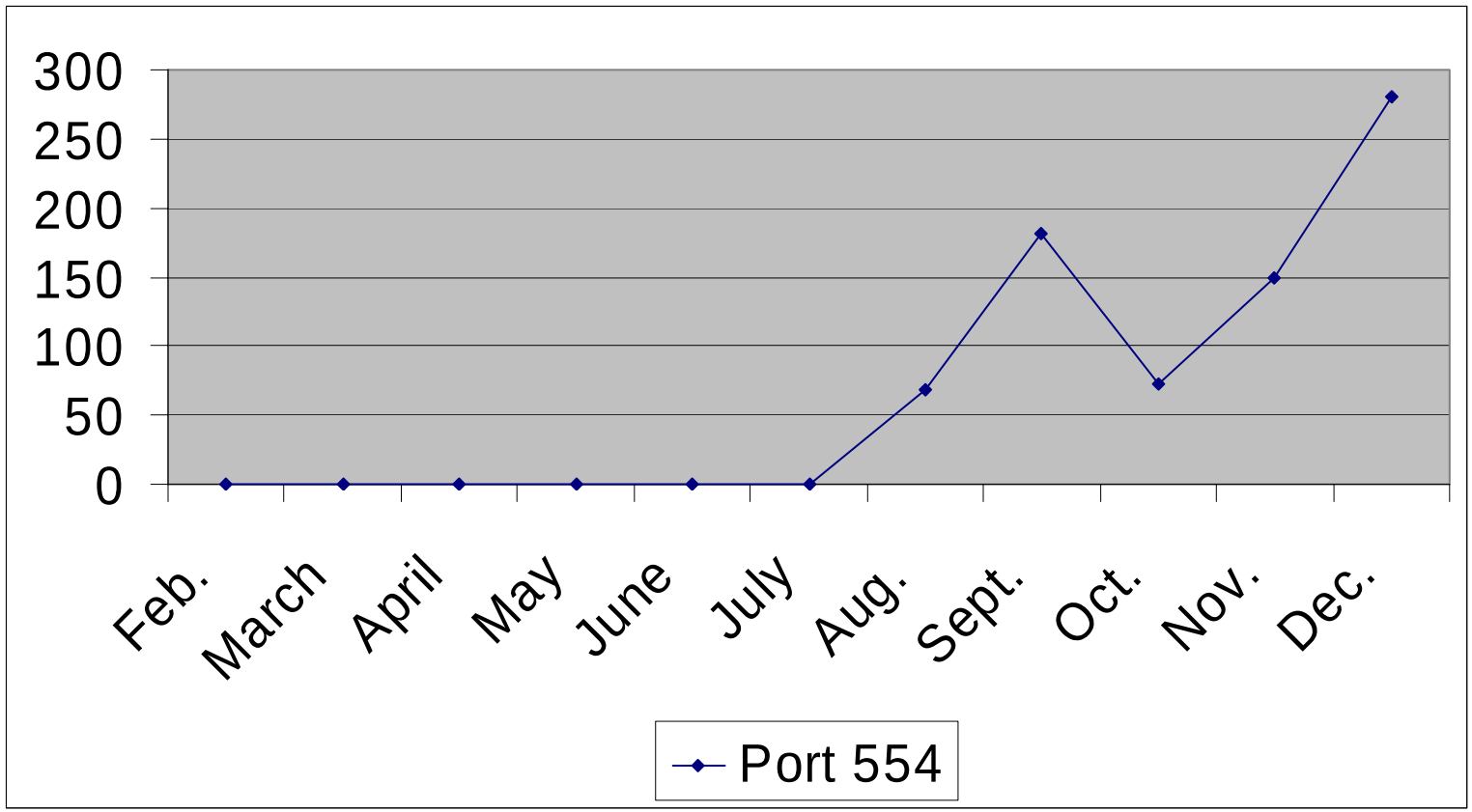
Results: source IP per dest. port



Worm Observation: W32 Blaster



Scanning Tools: sequence {554}

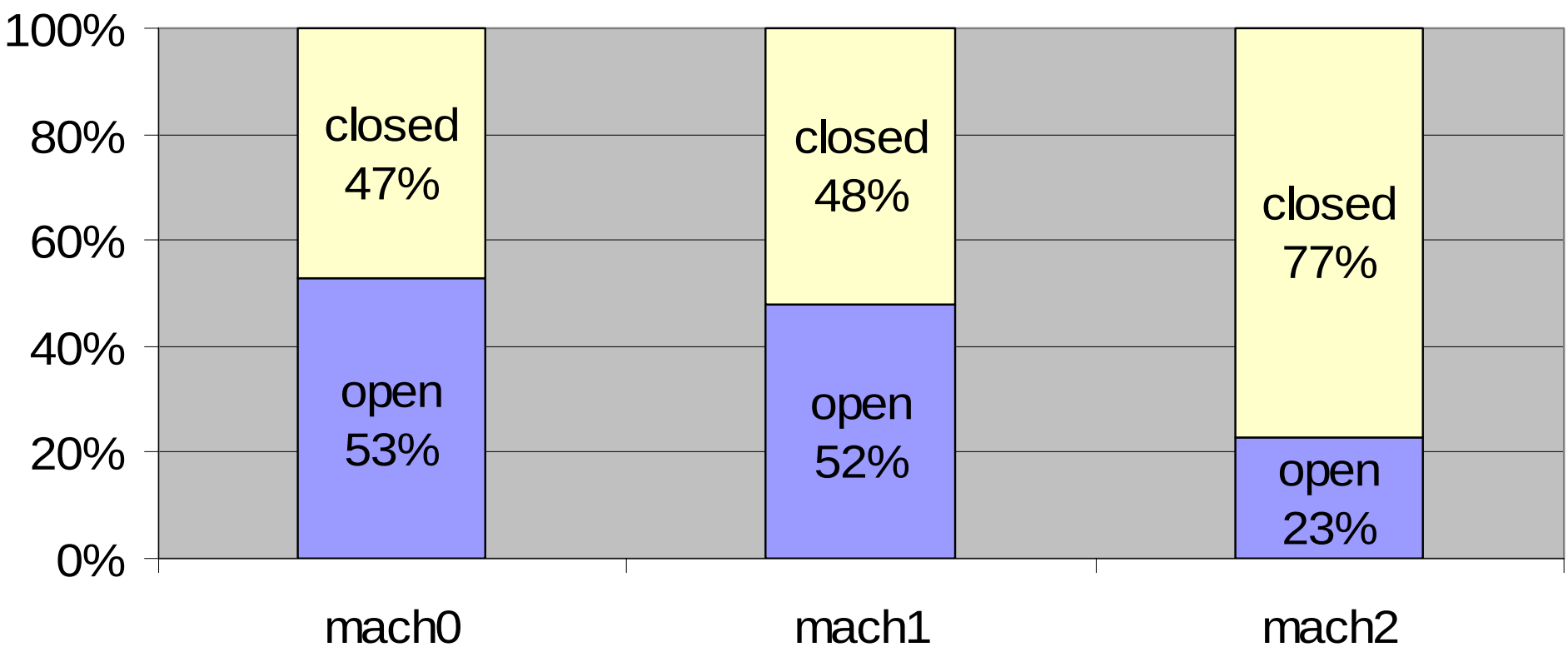


RTSP scanner

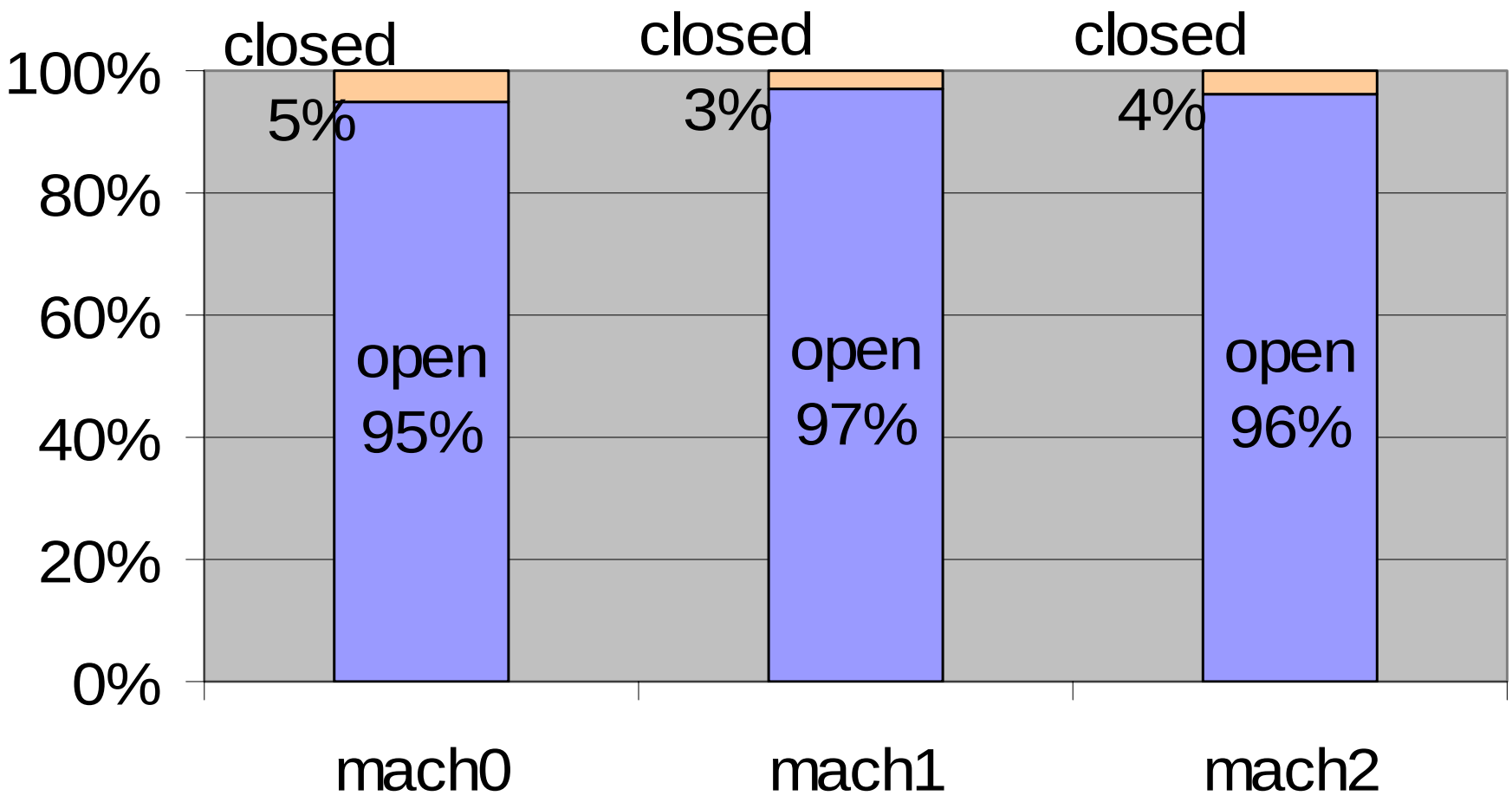
Overview

- Introduction :
  honeypots and related work
- Experimental set up
- Results:
  - The attacking machines
  - The attacked ports
  - The attacked machines
- Conclusions:
  - Proposal for a collaborative international set up

Results: the scanning machines



Results: the attacking machines

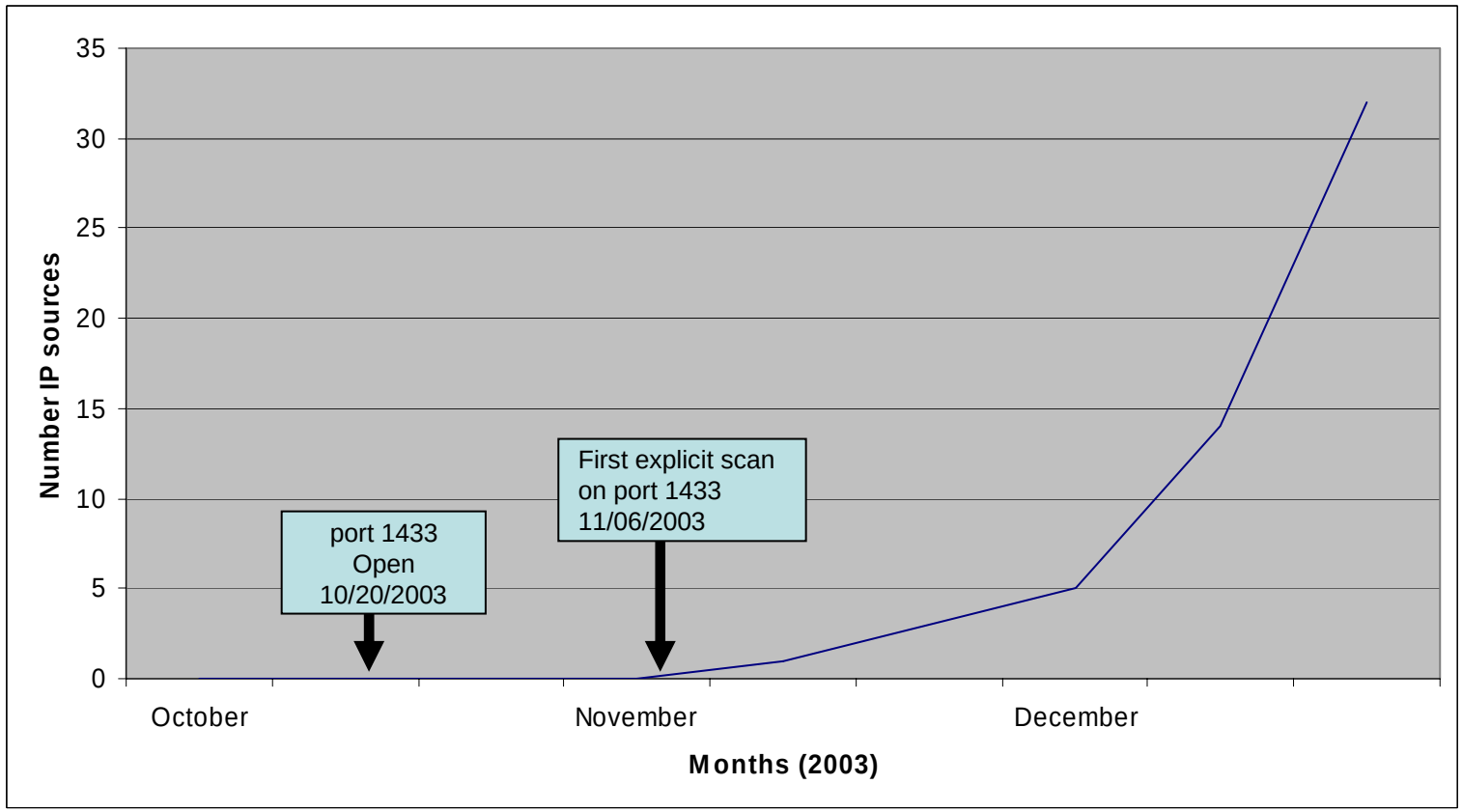


Attacks focusing on one target only

- Statistically speaking, very unlikely phenomena

- It shows that those machines benefit from the results of scans performed by other machines.

Targeted attacks: Port 1433 example



Overview

- Introduction :
  honeypots and related work
- Experimental set up
- Results:
  - The attacking machines
  - The attacked machines
  - The attacked ports
- Conclusions:
  - Proposal for a collaborative international set up

Conclusions

- Our honeypots have gathered a large amount of data the analysis of which has shown:
  - the existence of stable attack processes.
  - evidences of their sophistication as well as evidences of collusions between attackers
  - The richness and the usefulness of that kind of data set.

Questions

- Are our observations influenced by :
  1. The IP addresses used ?
  2. The ISP used ?
  3. The type of environment they belong to (academic) ?
  4. The geographical location (Sophia – France – Europe) ?
  5. Something else ?

Invitation

In order to answer these questions but also to help you understanding the threats that your own environment is facing,

we invite you

to join the large data gathering environment we are building.

The win-win ‘deal’

- We have built an environment that is very easy to deploy and to manage remotely.
- You just need to give us a couple of IP addresses (4 or 5) and an ‘old’ machine. We take care of the rest of the installation.
- By participating to this experiment, you will be allowed to query the whole DB where your data but also those from the other participants will be stored.

Interested ?

- Contact:

pouget@eurecom.fr
dacier@eurecom.fr
www.eurecom.fr/~dacier

Invitation (ctd.)

- **ESORICS 2004** (European Symposium on Research in Computer Security) and **RAID 2004** (Recent Advances on Intrusion Detection) will take place in **Sophia Antipolis**, in the French Telecom Valley (Nice), respectively from **September 13 to 15 and 15 to 17 2004**.

Questions

pouget@eurecom.fr