INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2011; **24**:75–93
Published online 26 April 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/dac.1141

Monitoring unauthorized internet accesses through a 'honeypot' system

Mario Marchese[1,*,†], Roberto Surlinelli[2] and Sandro Zappatore[1]

[1]*DIST—Department of Communication, Computer and System Sciences, University of Genoa, Via Opera Pia 13, 16145 Genova, Italy*
[2]*Polizia di Stato, Compartimento Polizia Postale, Via Dante 2, 16121 Genova, Italy*

SUMMARY

The role of the Internet is continuously increasing and many technical, commercial, and business transactions are carried out by a multitude of users who exploit a set of specialized/sophisticated network applications. In this context, the task of network monitoring and surveillance is gaining great relevance and honeypots represent promising tools to get information, and understanding about the 'areas of interests' of attackers, as well as about the possible relations among 'blackhat' teams. The paper presents and discusses the results achieved by a group of honeypots deployed within the networks of the Department of Communication, Computer and System Science at the University of Genoa. The collected statistics, measured over 4-month long period, reveal that approximately 10 000 different attackers, coming from 130 different countries, have 'contacted' the honeypot system and that about 60 000 TCP distinct connections have logged in. Our high-interaction honeypot has counted more than 25 000 attempts to access a ssh server, thus permitting to trace many attempts to install rootkits. A comparison with results obtained by similar researches carried out in other laboratories is presented and commented. Copyright © 2010 John Wiley & Sons, Ltd.

Received 27 November 2009; Revised 4 March 2010; Accepted 6 March 2010

KEY WORDS: network monitoring; honeypot; network security

1. INTRODUCTION

The Internet is playing a very important role in modern society: the number of users accessing the network continuously increases, as well as the number and kind of available applications. Though originally designed for military and research purposes, the Internet is now used for fast and reliable mail delivery, business and financial transactions, telemedicine, entertainment, telephony, and for controlling/accessing remote laboratories and a variety of devices, including, for instance, sensor nodes, small processors, and portable medical instruments. At the same time, the Internet is characterized by an increasing number of network attacks aimed at granting unauthorized accesses to computers, disturbing network traffic, damaging services, and intercepting data.

Consequently, network managers and skilled single users are trying to deploy suitable defense tools, such as firewalls, virus scanning, and intrusion detection systems [1–3]. Generally these tools can significantly benefit from the knowledge of the nature of attacks. In other words, keeping up defense against any possible attackers requires a continuous monitoring of network activities

*Correspondence to: Mario Marchese, DIST—Department of Communication, Computer and System Sciences, University of Genoa, Via Opera Pia 13, 16145 Genova, Italy.
†E-mail: mario.marchese@unige.it

in order to recognize traffic patterns, detect net anomalies, and understand the ways used for hacking/cracking a computer system.

In this scenario, honeypots [3–9] can be useful for two main goals. The first one concerns the significant possible aid in discovering rootkits, Trojans, and potential network risks. The second goal regards the chances to get information and understanding about the 'areas of interest' of attackers and the possible relations among 'blackhat' teams. In spite of the relevance of the problem, only a limited number of works devoted to illustrate the results achieved by inspecting the network are present in the literature. Under this perspective, the novelty of the paper is not in the honeypots themselves, but in the results obtained and in the remarks/hints that can arise from the collected statistics.

The paper is organized as follows. Section 2 reports definition and classification of honeypots and summarizes the state of the art. Section 3 illustrates the honeypot system implemented at our Department of Communication, Computer and System Science at the University of Genoa; Section 4 describes and critically discusses the principal results achieved through 4 months of network monitoring and system logging, which are the main scientific value of the paper. The obtained results are compared with the state of the art. Finally, the conclusions are drawn in the last section.

2. HONEYPOT: DEFINITION, CLASSIFICATION, AND STATE OF THE ART

Defining a honeypot is not so simple. The literature reports different definitions of honeypots depending also on the purpose of the publication. From the point of view of this paper, the most suitable definition is reported in [5]: 'a honeypot is a security resource whose value lies in being probed, attacked, or compromised'. Actually a honeypot that is not attacked is useless. When it is attacked it can provide information about attack methods' features and techniques. Honeypots may be roughly classified into two types depending on the level of interaction they have with the attacker: low-interaction and high-interaction. Low-interaction ones have a low level of interaction: the attackers see a limited number of (dummy) services, try accessing them, but there is no real operating system to crack. Low-interaction honeypots are easy to install, configure, and maintain, have a low risk level, but gather limited information, which is essentially reduced to: the attack time and date; the attacker's source IP address and TCP/UDP port; and the destination IP address and the TCP/UDP port of the attacked machine. Low-interaction honeypots' purpose is detection and measure of attackers' behaviors. High-interaction honeypots provide a real operating system to interact with, no service is emulated. They are more difficult to install, to deploy, and, above all, to maintain with respect to low-interaction honeypots. The risk level is high because the attackers have available a real operating system but, in turn, high-interaction honeypots can gather a huge amount of information about attackers' features and behaviors, including unknown and unexpected actions. Under this perspective high-interaction honeypots represent very powerful tools able to hide their actual nature.

The first honeypots were released at the very end of the last century but the literature about them began widespread at the beginning of 2000. Reference [10] contains a summary of the debate about whether honeypots are useful or not, and summarizes honeypot objectives, advantages, and drawbacks. The already mentioned reference [5] is a complete guide to honeypots, which comes directly from the experience of the author. Besides implementation details, the value of honeypots is in the measures they provide. Reference [11] shows data about attack time and attacked ports, and proposes a mathematical model to estimate the number of attacks; reference [12] reports the protocol (TCP, UDP, or ICMP) used to perform the attack, the IP source addresses, and the country from which attacks are brought, as well as the overall number of attacks. The analysis of the results is so important that some literature (e.g. [13]) is dedicated to select the most important data through Principal Component Analysis (PCA), which is used to reduce the dimensionality of a data set into few uncorrelated variables. A great push to the analysis of the collected results has been given by the international projects dedicated to honeypots. They are typically composed of volunteers who join the project and collaborate together to collect and comment data about attacks.

DOI: 10.1002/dac

The authors in [7] present a distributed system for identifying spammers and spambots they use to scrape addresses in web sites. Reference [8] describes a volunteer organization dedicated to the research about computer security. It is divided into Chapters, one for each nation interested in the activity. The aim of the organization along with the honeypot tools used to collect and analyze data are reported in [14] that contains also a long list of web references about honeypots and about the Honeynet project. Concerning National Chapters: among many others, [15] contains daily statistics about attacked TCP/UDP ports, country of the attackers, and operating system used for the attack; Reference [9] reports the activity of the group of Filipino volunteers whose mission is to promote information security, and to help individuals and organizations in the Philippines in protecting their computers and networks through research, education, and training. The web site mentioned in [9] contains a huge amount of collected data that have been used as a comparison in this paper, too. Furthermore, many private web sites contain statistics collected by honeypots: Reference [16] reports the last 24-h statistics about attacked ports and attackers' IP addresses; Juniper Networks maintain honeypots around the world to collect real-time statistics about vulnerabilities and threats; the related measures, available in [17], refer the most exploited vulnerabilities, the most attacked ports, and the number of severe security events. Reference [18] presents historical statistics about attacked ports for a period of 28 months.

This paper describes the use of low-interaction honeypots and one high-interaction honeypot to collect data about: (i) the protocols (TCP, UDP, and ICMP) used for the attacks, as in [12, 18]; (ii) the geographic origin of the attacks, as done in [7, 12]; (iii) the ports used for the attack, as in [11, 17, 18]; and (iv) the used operating system as in [9]. Additional to the state of the art, the paper: (i) suggests an association between the percentage of attacks and the number of Internet users of a nation; (ii) separates the protocol type used in the attack for each tracked nation; and (iii) analyzes the attack types by explicitly identifying the type of action performed by attackers within the high-interaction honeypot. A comparative discussion with the data in the literature is reported whenever possible.

3. HONEYPOT SETUP

In order to collect data on possible network attacks, a group of honeypots is deployed on a set of virtual machines hosted on a single physical computer. For our measurement campaign, the physical machine is a DELL 'Optiplex 740', equipped with 4 Gbytes of RAM and two Ethernet cards. The first card is used only to access the physical machine from a protected sub-network of our laboratory, while the second card is associated with a virtual Ethernet adapter of a virtual machine, specifically the one running the Honeywall software [19].

The virtualization solution adopted is the 'VMware Server 2', developed by VMware, Inc., and freely downloadable [20]. The VMware Server runs as an application on the physical server under Linux Operating System, Debian GNU/Linux 5.0 (Lenny).

At the moment, the virtualization infrastructure includes three virtual machines, named *Honeywall*, *HoneyPool*, and *Ssh-target*, each of them running a Linux operating system.

Figure 1 diagrammatically depicts the interconnection among the previously mentioned machines and shows the overall honeypot system used in this paper.

The incoming and outgoing traffic from/to the Internet passes through the Honeywall firewall [21] hosted by the virtual machine *Honeywall*. The firewall is configured to (i) log the packets passing through it, (ii) filter every type of packet addressed to *HoneyPool* (but not the packets going to the honeypots hosted by *HoneyPool*), and (iii) scrub the potentially dangerous traffic from *Ssh-Target*. The virtual bridge in Figure 1 is obtained by combining several facilities of the VMware Server with a TUN/TAP [22] server running on the physical machine. The virtual infrastructure made visible by the honeypot systems is sketched in Figure 2.

The machine named *HoneyPool* runs the honeyd (v1.5) [6] process, which allows exposing a group of honeypots that, in turn, mimic the behavior of a CISCO router and some Internet servers, such as a Mail Transport Agent (MTA), a TELNET, HTTP, FTP, and VNC server. Most

DOI: 10.1002/dac

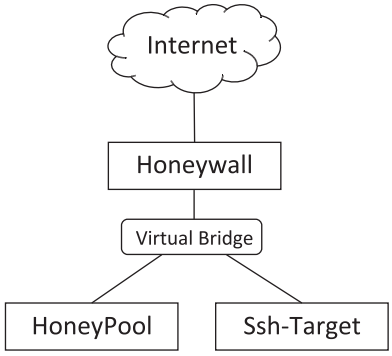

Figure 1. Honeypot system: Virtual machines of the honeypot setup and their interconnection.

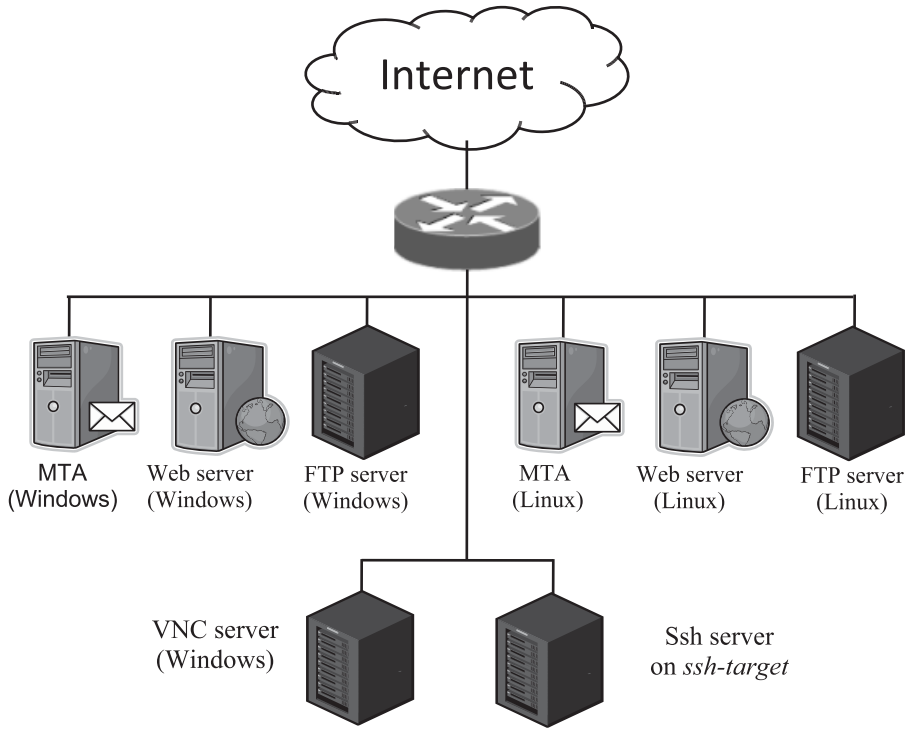

Figure 2. Virtual infrastructure made visible by the honeypot system.

of them are emulated as running in both Windows and Linux/Unix environments. This is a low-interaction honeypot that handles (mimics) the following ports: 21, 23, 25, 80, 110, and a list of ports over 1024.

The servers emulated by honeyd are scripted in Perl: every input received is logged in a specific file.

Furthermore, it should be highlighted that part of information gathered in our log files is inserted in an ad-hoc database, which permits to efficiently execute complex query on the data collected by the components of our measurement/inspection platform.

As concerns *Ssh-target*, the machine plays the role of a high-interaction honeypot [8] and exposes a modified version of an ssh daemon (port 22); no other Internet servers are active. The ssh daemon is suitably modified to log any decrypted received and transmitted packet, and to inhibit dangerous commands that an attacker might issue (e.g. rm –rf * executed from the root directory). Furthermore, in order to better isolate an incoming ssh connection, after a user is authenticated and a separate process (to actually serve the connection) is forked, the ssh executes the system

DOI: 10.1002/dac

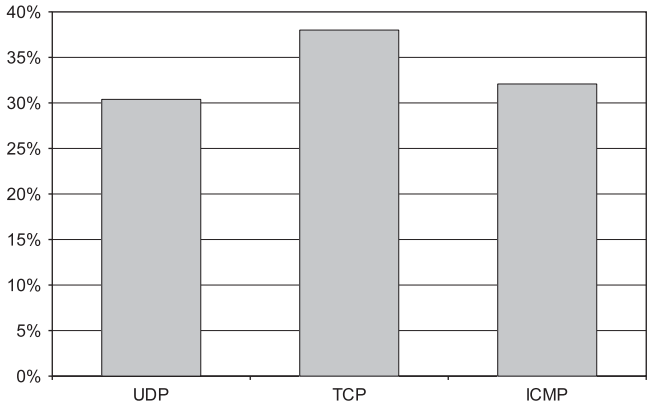

Figure 3. Percentage of protocols used (TCP, UDP, and ICMP) in the attacks.

call *chroot*: in this manner, the filesystem hierarchy 'visible' by the user is limited, thus jailing the attacker in a restricted environment, which mimics the actual root directory of the system.

In summary the honeypot system used in this paper is composed of the following virtual machines:

- 1 PC that runs honeyd and implements a low-interaction honeypot, which emulates Windows and Linux PCs with the services ftp, telnet, smtp, http, pop3, and a list of services connected to ports over 1024;
- 1 PC that operates over Linux operating system and implements a homemade high-interaction honeypot which provides only SSH service;
- 1 PC (Honeywall) which acts as a firewall to limit accesses.

4. RESULTS

The attacks have been monitored over a period of 4 months. The aim of the authors here is to synthesize and show the most meaningful results by using all the features offered by the designed and implemented honeypots. During the 4 months observation period we have registered 905 649 IP packets from 10 156 different IP addresses.

4.1. Protocols used for the attacks

The first result concerns the analysis of the protocol encapsulated in the IP packets involved in the attack. Figure 3 shows the percentage of the protocols: ICMP, TCP, and UDP, over the overall period of analysis. If an attack comes from the same IP address and uses the same protocol (i.e. ICMP, TCP, or UDP), it is counted just once to get Figure 3. The measured percentages for each protocol range between 30.39% of UDP to 38% of TCP. There is a substantial balance among the previously mentioned protocols. This is not true by analyzing the number of packets that flow during the attack. Figure 4 contains the percentage of overall packets employed in the attacks for each single protocol: more than 96% of the packets belongs to TCP, about 3% to ICMP, and less than 1% to UDP. All packets are counted to compute the percentage in Figure 4. The differences in the percentages shown in Figures 3 and 4 may be partially motivated by the traffic generated by each protocol, by the usage of the scan tools (Nmap and Nessus), but they are essentially motivated by the fact that many attacks come from the same machines and by the same protocol (i.e. ICMP, TCP, or UDP). This is clearly evidenced in Table I that shows Top 11 IP source hosts carrying ICMP together with the number of performed attacks and its percentage over the overall number of attacks: many attacks come from the same IP addresses. The strong unbalance among the number of packets belonging to the different protocols (TCP, UDP, and ICMP) and its motivation is evident also in [12], which shows that most attacks come from the same IP addresses

DOI: 10.1002/dac

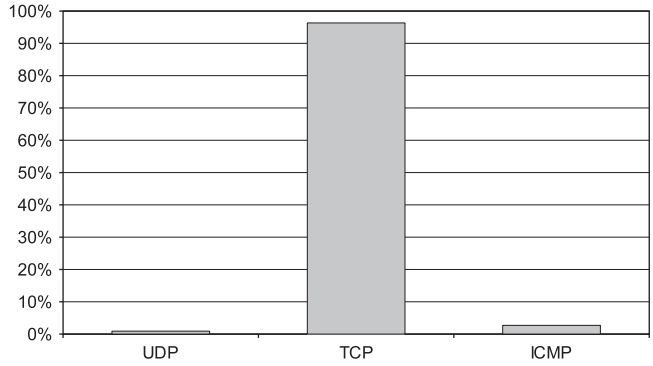

Figure 4. Percentage of the overall number of packets used in the attacks for each protocol (TCP, UDP, and ICMP).

Table I. Top 11 attacker source IP addresses carrying ICMP.

Source IP address of the attacker	Number of performed attacks	Percentage of performed attacks over the overall number of attacks
213.140.15.163	388	1.51
61.134.0.29	365	1.42
151.23.228.98	332	1.29
24.124.116.111	42	0.16
59.16.50.44	40	0.16
12.36.231.78	36	0.14
211.78.4.99	36	0.14
202.138.134.162	32	0.12
64.135.252.47	32	0.12
24.241.231.247	30	0.12
89.46.83.92	30	0.12

and reports the used upper layer percentage counting all the attacks (also the ones coming from the same IP address and using the upper layer protocol). The result in [12] is that 91.7% of violation attempts are TCP, 4.85% are UDP, and 3.45% are ICMP. Extracting the percentage of only TCP and UDP packets from [18], the unbalance is even more evident: 99.8% TCP and 0.2% UDP. The percentages are comparable with the ones shown in Figure 4. Briefly, Figure 3 says that the attacks coming from different machines use similarly TCP, UDP, and ICMP, but Figure 4 adds that most attacking traffic is TCP. The machines should build their defenses consequently with this information.

4.2. Geographic origin of the attacks

The second meaningful result concerns the geographic origin of the machines used for attacks. If an attacker uses more than one machine to make an attack by exploiting a chain of compromised computers, the results in the following consider only the last touched machine. Figure 5 presents the percentage of attacks from a specific nation, measured through the IP addresses of the attackers, associated with the percentage of Internet users of that nation out of the overall Internet users (data taken from [23], updated at June 30, 2009). Only the 10 most attacking nations are shown.

The percentage of attacks per nation is confirmed also in [12], whose numerical results are shown in Figure 6. The two nations, Taiwan and The Netherlands, which appear in Figure 6 but not in Figure 5, rank, respectively, 12th (with 2.18%) and 15th (with 1.69%) in our measures. It is also interesting to check the statistics measured by the Project HoneyNet [8]. Even if the project is focused on Internet robots and e-mail spammers and reports statistics on harvesting, spam

DOI: 10.1002/dac

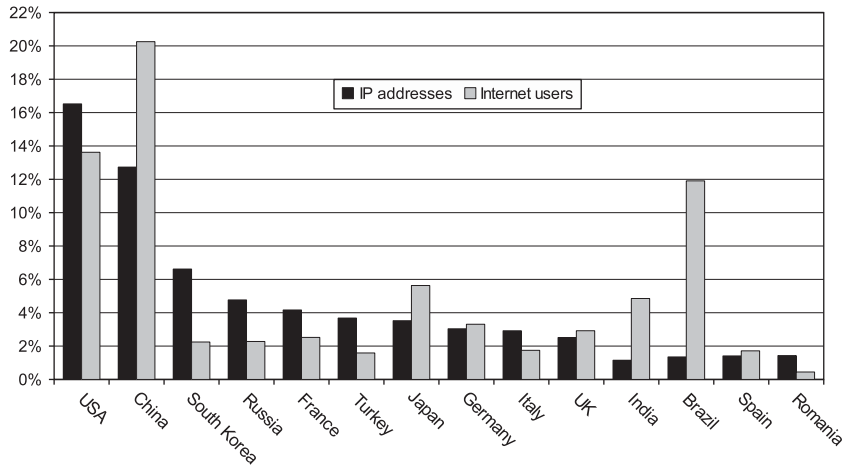

Figure 5. Percentage of attackers IP addresses and of Internet users from a specific nation.

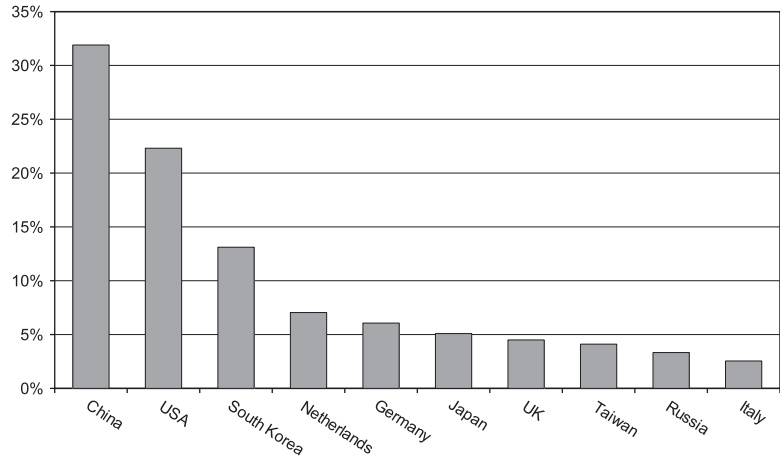

Figure 6. Percentage of attackers IP addresses from a specific nation—values from [12].

sending, dictionary attacks, and comment spamming, their measures confirm the important role of China, United States, Russia, Turkey, and Germany. On the other hand, Reference [8] certifies the importance of Brazil, concerning spam sending, dictionary attacks, and comment spamming; India, concerning dictionary attacks and comment spamming; Spain and Romania, concerning harvesting, which are not included in our Top 10 of Figure 5. Brazil ranks 22nd (1.35%), India 14th (1.85%), Spain 21st (1.41%), and Romania 19th (1.43%) in our tests. This may suggest the presence of special expertise for peculiar actions within any nation, as should be clearer also from the results reported in the remainder of this paper.

Figure 5 suggests that the association between the number of Internet users and the number of hackers in a specific nation is not simple to understand: more than 6% of the attacks are carried out from South Korea, but South Korea provides about 2% of the overall Internet users in the world. On the other hand, India provides 4.85% of the Internet users, but only 1.85% of the attacks. Actually India, not being in the top 10 of our attackers, is not even shown in Figure 5 but, as said, it seems to play a relevant role for attacks and comment spamming dictionary.

The percentage of protocol type (TCP, UDP, and ICMP) used for the attacks from each single top 10 nation is worth noting. Figure 7 contains the measures. The histogram confirms that there is a sort of specialization and preference for one specific protocol in each country. To better clarify this

DOI: 10.1002/dac

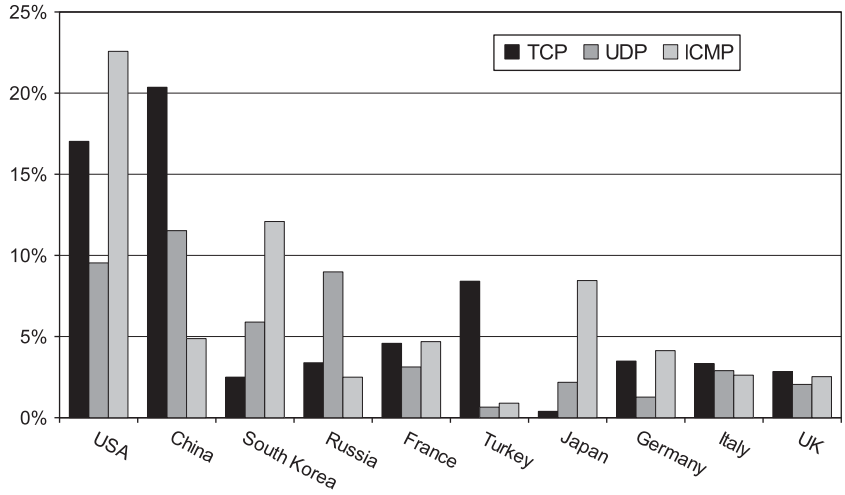

Figure 7. Percentage of protocol type (ICMP, TCP, and UDP) used for attacks.

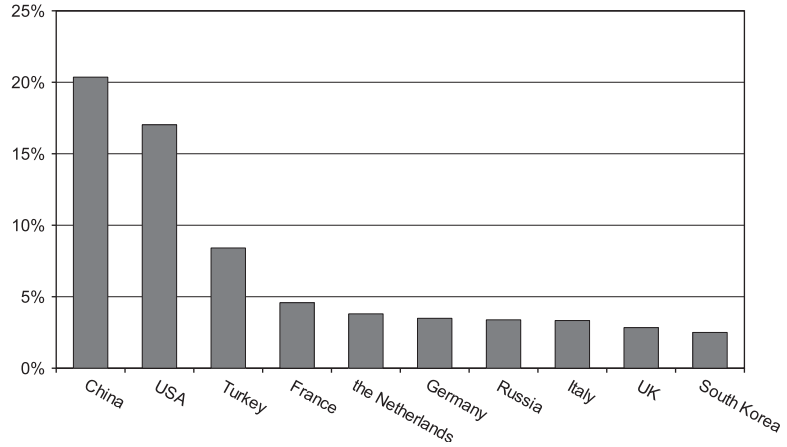

Figure 8. Percentage of TCP packets for specific nation.

aspect, Figures 8–10 show the top 10 nations for the protocols TCP, UDP, and ICMP, respectively. It is interesting to note: (i) the activity of the Netherlands and the low interest of Japan, concerning TCP; (ii) the importance of Ukraine, India, Vietnam, and Philippines concerning UDP. Looking at UDP measures, it is also interesting to note the lower difference between the more active country (China, 11.52%) and the tenth one (Italy, 2.90%) with respect to the difference measured for TCP (China—20.36% and South Korea—2.50%). As concerns ICMP, the big role of USA and the importance of South Korea and Japan, together with the presence of Canada and Taiwan in the Top 10, should be noted.

There is a specialization also for single attackers: trying to evaluate whether there are attacks from the same IP address, but using different protocols, taking Figure 4 as reference, the result is that only 0.44% of IP addresses use both TCP and ICMP; only 0.23% use both TCP and UDP; and only 0.1% use UDP and ICMP. Each attacker, identified by the attacker's IP address in this case, is specialized to use a specific protocol.

4.3. Attacked ports

Figure 11 shows the average percentage of attacks divided for TCP/UDP ports over the entire period of investigation. The overall number of attacked ports has been 57,249. The list of available

DOI: 10.1002/dac

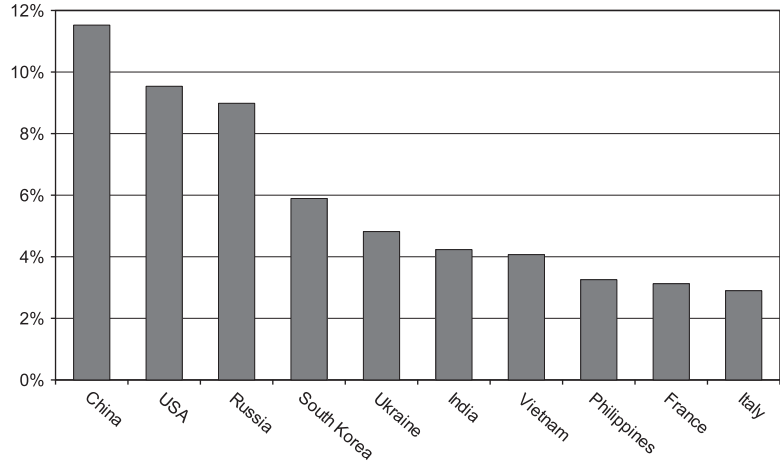

Figure 9. Percentage of UDP packets for specific nation.

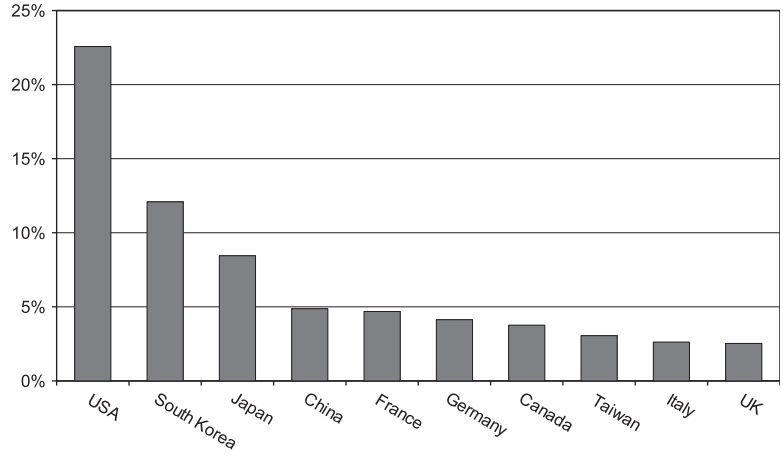

Figure 10. Percentage of ICMP packets for specific nation.

ports on our honeypot system (viz. low-interaction honeyd plus the high-interaction honeypot) is quite limited: ports 21, 22, 23, 25, 80, 110, and some ports over 1024. SSH service at port 22 is available in the high-interaction honeypot, all the others in the low-interaction honeypot. SSH service has been privileged by the attackers. The result is hardly comparable in the literature because the number of available services is different, as well as the overall number of attacks, but some measures can be provided. Reference [18] reports a huge amount of data focused on TCP and UDP ports extended over a period of 28 months. The overall number of attacked ports throughout this period has been over 4 300 000. The percentage of attacks divided for TCP/UDP port from [18] is reported in Figure 12. Ports 135, 139, 445, dedicated to Microsoft services seem to attract most threats. This impression is confirmed also from the data contained in [11], whose percentages are reported in Figure 13. The overall number of attacks in [11] is 461 047 over a period of 1 year. The mentioned Microsoft services are not available in our honeypot system and the effect of this seems to be the focalization of the attacks on port 22 and, partially, on port 80, which, anyway, results heavily attacked also in [11, 18]. This is due to the fact that port 22 is the only one served on the high-interaction honeypot, which obviously attracts much more threats than low-interaction honeypot. Probably if the services on port 22 were not available in the high-interaction but implemented in the low-interaction honeypot, the measured percentage for port 22 would have been comparable to the measures in [11, 18]. The comparison of the three

DOI: 10.1002/dac

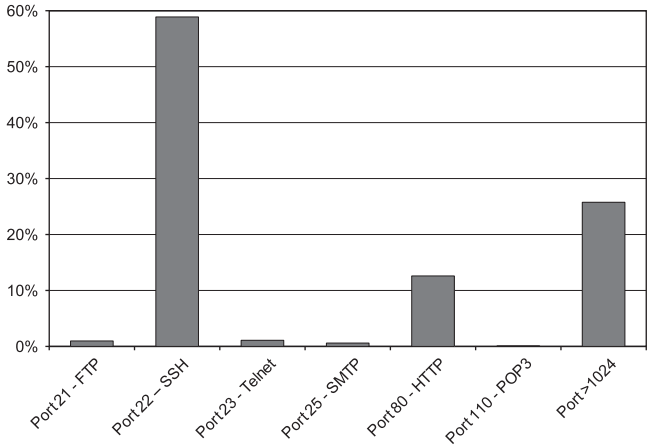

Figure 11. Percentage of attacks divided by TCP port.

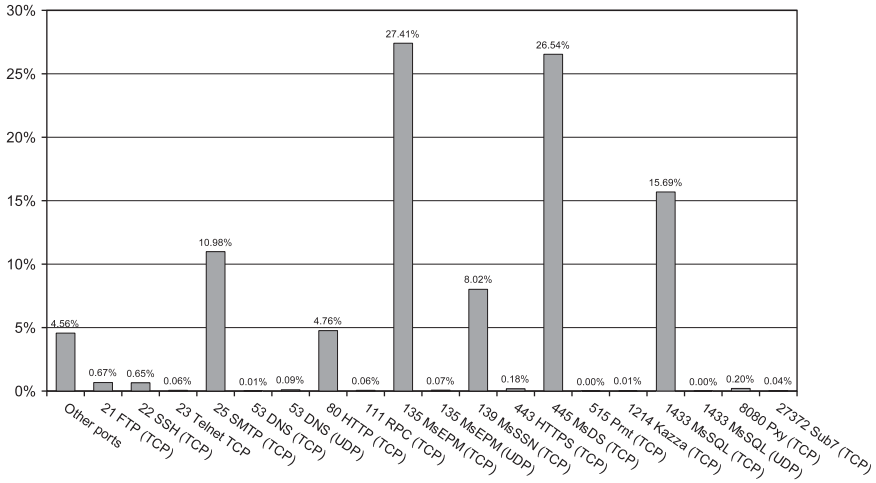

Figure 12. Percentage of attacks divided by TCP/UDP port (data from [18]).

experiments confirms the low attraction of ports 21 and 23. Ports 25 is not so much used in our measures and in [11], while it is heavily attacked in [18]. The motivation of this behavior is simply comprehensible from the data in [18], which are divided for months: except for three specific months (April, May, and June 2004, where port 25 represents, respectively, 58.15, 60.05, and 9.80% of the overall attacks), the percentage of port 25 attacks is conformant to the values shown in Figures 11 and 13; dropping the three mentioned months the percentage of attacks through SMTP is 0.18%. Also the other ports have meaningful oscillations depending on the observed month but the concentration of huge peak traffic into a so short time period is peculiar of only port 25.

The concept of specialization previously mentioned is even clearer by observing the used ports. Figure 14 shows the number of attacks (in logarithmic scale) versus specific TCP/UDP ports. Reported data are divided for nation. The attackers from the same country preferably use the same port (e.g. China and port 22) and tend to ignore other ports as happens for the attackers from Croatia who ignore Port 23 and 25.

Figures 15 and 16 show the percentage of attacks divided by nations involving, respectively, port 22 and port 80, which are the most attacked ones in our measures and therefore are the most meaningful. The specialization of attacks is outstanding, in particular for Peru, concerning port 22, and the Netherlands, concerning port 80.

DOI: 10.1002/dac

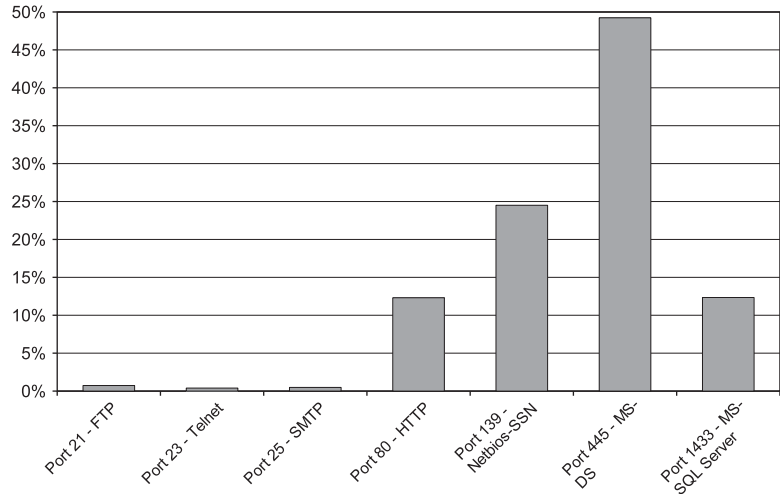

Figure 13. Percentage of attacks divided by TCP/UDP port (data from [11]).

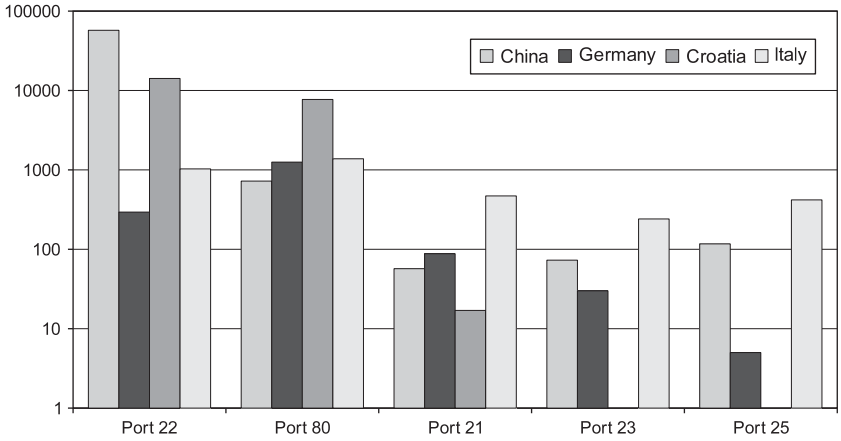

Figure 14. Number of attacks (in logarithmic scale) versus specific TCP/UDP ports.

4.4. Operating systems

The information about the operating system used by the attacker is also very interesting. Figure 17 contains the percentage of attacks by using both Windows and Linux operating systems. Percentages are computed through a *honeyd* facility and refer only to attacks carried through TCP. Almost 78% of the attacks are performed by using Windows in our measures. The result is coherent with the same measure carried out by the 'Philippine HoneyPot Project' [9], also reported in Figure 17 in dark grey.

The numbers are about reversed when showing the percentage of packets generated by the attackers, divided again between Windows- and Linux-generated packets. Figure 18 shows the percentages we have measured. Linux machines are dominant concerning the generated traffic.

4.5. Attack types

We have registered three main attack types:

- Authentication violations
- E-mail server spammer
- Network scan

DOI: 10.1002/dac

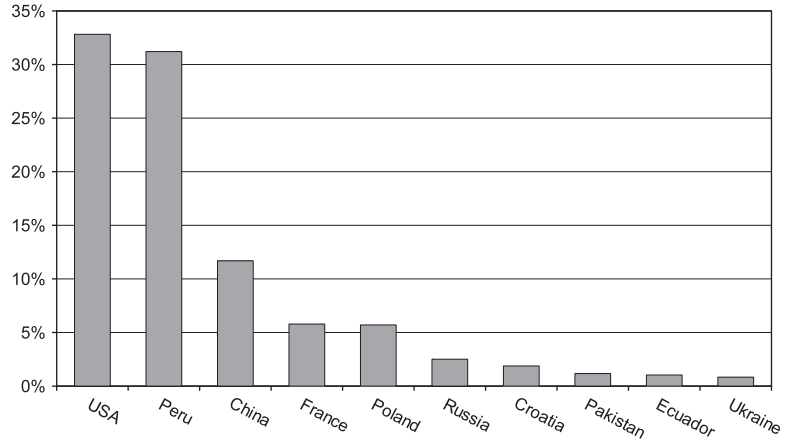

Figure 15. Percentage of attacks to Port 22 divided by nations.

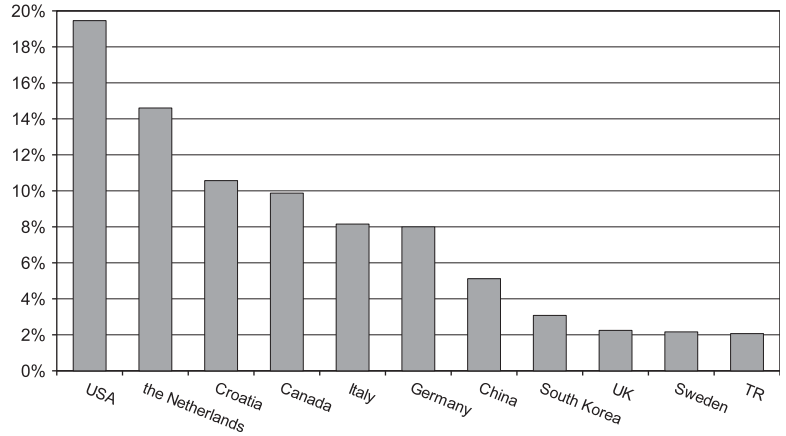

Figure 16. Percentage of attacks to Port 80 divided by nations.

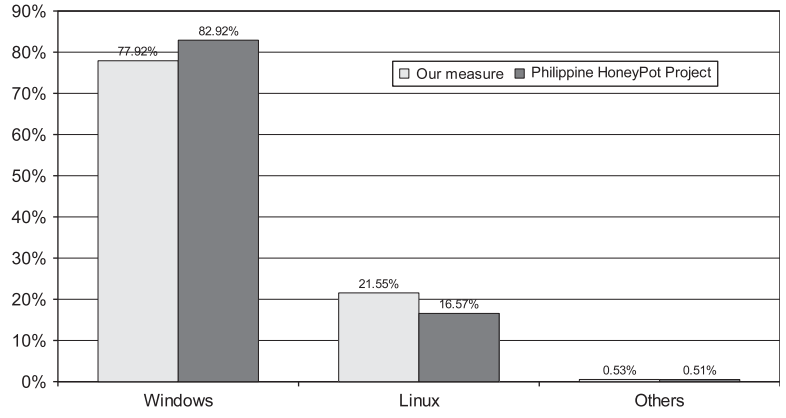

Figure 17. Percentage of attacks by using Windows and Linux operating systems.

Authentication violations are performed through software attempts in sequence by trying different users and passwords. No particular intelligence is used and attacks typically employ a user/password dictionary.

DOI: 10.1002/dac

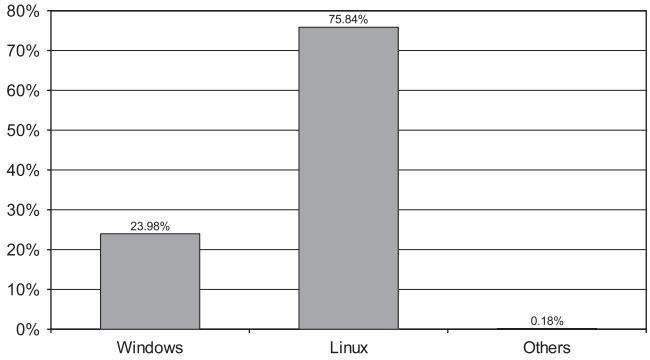

Figure 18. Percentage of packets generated by the attackers by using Windows and Linux operating systems.

Table II. List and percentage of most frequently used 'username'—FTP and SSH service.

FTP		SSH	
Username	Usage percentage	Username	Usage percentage
Administrator	9.55	root	31.62
admin	5.79	test	0.79
user	4.27	admin	0.70
www	3.91	oracle	0.34
john	3.04	user	0.32
peter	3.04	test123	0.27
apache	3.03	password	0.25
dave	2.94	test1	0.24
test	2.23	test2	0.21
guest	2.14	123456	0.20
info	1.52	test3	0.20
jeff	1.52	test4	0.20
lisa	1.52	guest	0.18

Referring to the e-mail server spammer and, specifically, to an example of sending e-mail: the attacker contacts the e-mail server via the SMTP protocol and creates an e-mail conforming RFC821 specifications in order to check the actual availability of the service.

Concerning network scan a simple example related to SSH service may help understand: the available machines are contacted in sequence; the attacker looks for the SSH service over the TCP port 22; once got a machine hosting the server, a connection is created. A passive fingerprinting tool recognizes the operating system of the attacker.

More detail is reported below for each single attack type.

4.5.1. Authentication violations. The attacker has no information about username and password needed to authenticate and guesses a number of usernames and passwords. The large number of received attempts suggest that attackers use software for automatic connection and attempt. There are two offered services in our honeypot system for which authentication is needed: FTP (port 21) and SSH (port 22). As said, the former is offered within the low-interaction honeypot, the latter within the high-interaction one. FTP is offered (mimicked) both over Microsoft and Linux; SSH only over Linux. The two services are investigated concerning used usernames and passwords both separately and jointly. The results are also compared with the measures found in [24]. Tables II and III contain, respectively, the list of most frequently used 'username' and 'password' concerning FTP and SSH services along with their usage percentage. It is interesting to compare the authentication attempts used in FTP and SSH. Although the attackers prefer using

DOI: 10.1002/dac

Table III. List and percentage of most frequently used password—FTP and SSH service.

FTP		SSH	
Username	Usage percentage	Username	Usage percentage
Cowboy	0.11	123456	3.34
dragon	0.11	1234	2.10
fuckyou	0.11	123	1.89
changeme	0.10	changeme	0.79
Basket	0.10	password	0.52
reddog	0.11	test	0.48
Amanda	0.09	newpass	0.39
peter	0.09	test123	0.35
apache	0.08	admin	0.34
Password	0.06	root	0.29
andrew	0.06	test1	0.27
george	0.06	testing	0.25
matthew	0.06	testuser	0.24
michael	0.06	tester	0.23

usernames that guarantee powerful rights both for FTP and SSH, SSH evidences in Table II a strong unbalance towards the username 'root'. This unbalance is less evident for FTP where usernames 'Administrator', 'admin', and 'root' represent, together, 19.05% of the attempts. Moreover, the used dictionary is different. This is due to the difference both in the operating system and in the service. Actually the username 'root' is the only Linux/Unix superuser; as a consequence, it is simple to assume that an SSH attacker tries acquiring the most powerful rights by trying the username 'root' almost exclusively. Things are different for an FTP attacker for two reasons: (1) being used also in Windows, a superuser may be different from 'root' because any username may have superuser rights and (2) an FTP attacker typically does not try to take control of the machine but attempts to access only a portion of the disk and, after getting the access, tries controlling the entire machine. Comparing the passwords in Table III is also interesting. FTP shows a substantial equivalence among the passwords used in the attacks; no password is really dominant and the usage percentage for each password is low; it means that attackers use a wide range of passwords that have a low reuse frequency. The percentage differences among passwords are more remarked for SSH; it implies a relatively smaller number of used passwords and a higher reuse frequency of single passwords. Also the used passwords are different; the only correspondences among the most frequently used passwords are the words 'changeme' and 'Password'. All the others are different. It implies that adopted dictionaries are different for FTP and SSH. As in the case of usernames, this is likely due both to the different nature of service and to the difference in the operating system. The results shown in Table III for SSH are very similar to the results obtained in [24] concerning both dictionary and measured percentages.

Regarding authentication it is also interesting to evaluate the probability distribution of the password length. Figure 19 contains the probability distribution of the password length, measured in number of characters. The distribution is computed on the basis of the relative frequency of a given length within our SSH tests. The highest relative frequency is measured for 5-character passwords. Meaningful frequencies refer to passwords whose lengths range from 2 to 12 characters. Frequencies of passwords longer than 18 characters are practically negligible and close to zero (e.g. only 1 occurrence each has been measured for single passwords between 22 and 44 characters).

4.5.2. E-mail server spammer. We have measured that most attacks come from China, Taiwan, and USA, and that most source IP addresses employed to attack are already registered in known spam lists (e.g. the Spamhaus Project in [25]). A spammer typically acts as follows: (1) he/she sends an e-mail to one of his/her own addresses to check the presence of a real service so

DOI: 10.1002/dac

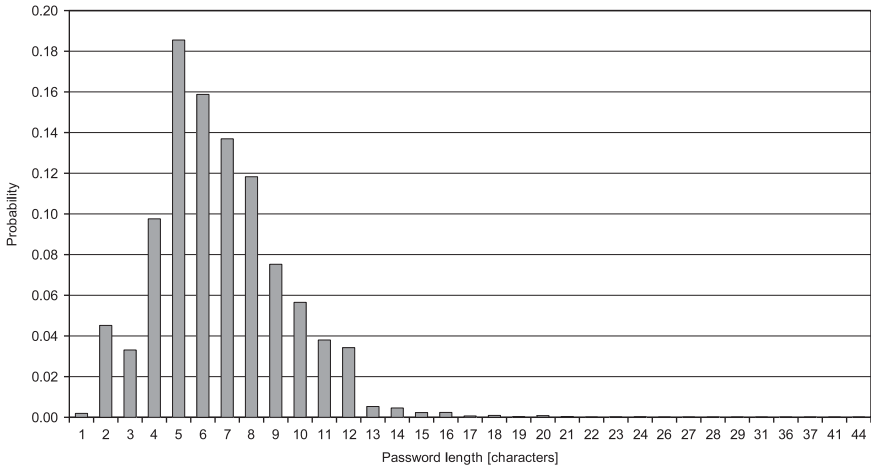

Figure 19. Probability distribution of the password length measured in number of characters.

Table IV. E-mail server spammer: sender and receiver e-mail addresses and relative percentages.

Sender	Sender percentage	Receiver	Receiver percentage
aaaaa@yahoo.com.tw	23.3	dvdr2000@yahoo.com.tw	8.8
ttc585ttc585@yahoo.com.tw	19.2	byvc84942@yahoo.com.tw	5.0
usdrh21sdhh32df@yahoo.com.tw	3.3	wxjszv0822@yahoo.com.tw	5.0
idahj23das@yahoo.com.cn	3.3	flveo7847@yahoo.com.tw	3.8
ehyqgpy@yahoo.com.tw	1.7	iqvio4255@yahoo.com.tw	3.8
qoilvkn@yahoo.com.tw	1.7	grpt42942@yahoo.com.tw	3.8
vjfhdtug@yahoo.com.tw	1.7	zrzpw9175@yahoo.com.tw	3.8
mvszz@yahoo.com.tw	1.7	oqhmomuf28376@yahoo.com.tw	3.8
hjwyz@yahoo.com.tw	1.7	dcu846eg@yahoo.com.tw	3.8
uvzljxce@yahoo.com.tw	1.7	csrsh2878@yahoo.com.tw	3.8
ooyyw@yahoo.com.tw	1.7	vjei3b9e@yahoo.com.tw	2.5
vearbka@yahoo.com.tw	1.7	zauuejh99734@yahoo.com.tw	2.5
ysdidash312@yahoo.com.cn	1.7	jyybe7066@yahoo.com.tw	2.5
udsfh324sd@yahoo.com.cn	1.7	zfdjtyex93472@yahoo.com.tw	2.5

as to avoid fake services introduced by low-interaction honeypots; (2) in the case of a positive output of point (1), the user begins the systematic sending of spam e-mails. Having provided the e-mail service only in the low-interaction honeypot, we have checked only point (1). Table IV shows the result of this analysis. It presents sender and receiver e-mail addresses involved in action (1) together with their usage percentage in the tests. The percentage of senders is more concentrated on specific addresses, while the percentage of receivers is more distributed. This is due to the fact that many different attackers exploit the same software to generate messages and thus the sender address is common to a large number of attacks against different servers.

4.5.3. Network Scan: *Attacks through SSH.* Concerning the attacks to the high-interaction honeypot, which provides only the SSH service, three different attack types have been observed. For each of them the behavior of the attacker has been monitored attentively.

- *Investigation of the attacked machine*
  The attacker checks the features of the attacked machine and the characteristics of the operating system, and then removes the traces of the intervention. The list of performed actions is reported in Figure 20. In this case the attacker seems simply curious.

DOI: 10.1002/dac

```
w
passwd
passwd
ls        -a
pwd
uname     -a
cat       /etc/issue
cat       /proc/cpuinfo
cat       /etc/passwd
ls        -a
ls        -all
cd        l1
ls        -a
cd        /tmp
cd        /var/tmp
cd
ls        -a
cd        ~
cd        /
ls        -a
cd        /home/admin
ls        -a
rm        -rf  .bash history; touch     .bash history; exit
```

Figure 20. Actions to check the features of the attacked machine.

```
w
passwd
password
uname     -a
cat       /proc/cpuinfo
uptime
dir
ps        x
ps        aux
cd        11
ls
cd        l1
ls
cd        ..
rm        -rf *
```

Figure 21. Actions to delete all files.

- *File Deletion*
  In this case the attacker after checking the features of the machine and of the operating system, as in the previous case, tries deleting all files. This attacker wants to make severe damages to the attacked system. Detailed actions are reported in Figure 21.
- *Creation of an IRC* (*Internet Relay Chat*) *channel*
  Internet Relay Chat (IRC) is a form of real-time text messaging called 'chat', mainly designed for group communication in discussion forums said 'channels'. The attacker downloads the package containing the IRC software, attempts to install it and tries activating an IRC channel with the purpose to attack a third system. Figure 22 shows the details of this attack.

5. IMPLEMENTED HONEYPOT OPERATIVE APPLICATIONS

A copy of the high-interaction honeypot described in this paper is installed by a location of the Compartimento Polizia Postale, Italy, which is the main police department to contrast electronic crimes. The remainder of this section reports the practical and operative use of the implemented honeypot performed by the mentioned police department. The results provided by the implemented honeypot have been used to catalogue the attack types concerning:

(a) dates and hours of the attacks
(b) type of action to crack the system and perform the attack

DOI: 10.1002/dac

```
ls
w
ls        -a
cd        l1
ls
cat       mech.set
uname     -a
uptime
#c4       cat  mech.pid
cd        mech.pid
ls
cat       mech.pid
passwd
kill      -9   11907
ls
kill      -9   11907
die       -9   11907
wget      vmv.do.am/booti/dr.tgz
cd        /var/tmp
cd        /tmp
ls
cd        darwin
```

Figure 22. Actions to create an IRC channel.

(c) source IP addresses of the attacker
(d) nation from which the attack has been brought

Collected measures have shown that the attacks performed through new techniques are particularly concentrated in the days immediately consecutive to the release of new security patches and are brought on not-updated systems. The high-interaction honeypot allows tracking each single step followed by the attackers. The analysis performed by the police confirms that, as shown in Section 4.5.3, once exploited the system vulnerability, an attacker installs some utilities and toolkits, which can help attack a third system, and from there another one, so to create an attacked systems' chain. The creation of a chain of attacked systems, due to connection nesting, also allows removing the traces of the attack or, at least, making attack tracing very difficult. This increases the threat power because attackers feel safer. Information about attackers' IP addresses has been used to create a black list and to inhibit the access to special servers and, in particular cases, to block all the traffic coming from the most active IP addresses. Analyzing the attacks' statistics concerning the nations, the result is that some attack's types are limited to a group of nations. This helped isolate the attacks and take countermeasures.

6. CONCLUSIONS

The paper has presented the implementation and measured results of a honeypot system to monitor the attacks from different viewpoints. The honeypot, registering the details of intrusions and attackers' single actions, has allowed checking the used protocols (TCP, UDP, and ICMP), the geographic origin and the port used for the attacks, as well as the employed operating systems. In all the cases, the results confirm the ones obtained by other research groups. Furthermore, the paper has analyzed the association between the percentage of attacks and the number of Internet users of a nation; the protocol type used in the attack for each tracked nation; and attack types by focusing on authentication violations, e-mail server spammer, and network scan through SSH, also specifying the detailed actions performed by the attackers.

Future investigation may be dedicated: (i) to focus on high-interaction honeypots also increasing their number, thus creating a high-interaction honeynet; (ii) to extend the period of measure; (iii) to put the system in full operation for the benefit of universities, public administrations, and companies; and (iv) to use PCA to select the most meaningful data and reduce the dimensionality of the data set.

DOI: 10.1002/dac

REFERENCES

1. Sobh TS. Wire and wireless intrusion detection system: classification, good characteristics and state of art. *Computer Standard Interface* 2006; **28**(6):670–694.
2. Münz G, Li S, Carle G. Traffic anomaly detection using K-means clustering. *Proceedings of GI-IGT Workshop*, *MMBnet*, Hamburg, September 2007; 13–14.
3. Sourour M, Adel B, Tarek A. Ensuring security in depth based on heterogeneous networks security technologies. *International Journal of Information Security* 2009; **8**(4):233–246.
4. Intrusion Detection, Honeypots, and incident Handling Resources. Available from: www.honeypots.net.
5. Spitzner L. *Honeypots*: *Tracking Hackers*. Addison Wesley: Boston, MA, 2002. ISBN 0-321-1095-7.
6. Development of the Honeyd Virtual Honeypot. Available from: www.honeyd.org.
7. Project Honey Pot. Available from: www.projecthoneypot.org.
8. The Honeynet Project. Available from: www.honeynet.org/project.
9. Philippine HoneyNet Project. Available from: http://www.philippinehoneynet.org/.
10. Sink M. The use of Honeypots and Packet Sniffers for Intrusion Detection. *SANS Institute 2000–2002*, 15 April 2001; 1–6. Available from: http://www.lib.iup.edu/comscisec/SANSpapers/msink.htm.
11. Oumtanaga S, Kimou P, Gaza Kevin K. Specification of a model of honeypot attack based on raised data. *World Academy of Science*, *Engineering and Technology* 2006; **23**:59–63.
12. Barenco Abbas CJ, García Villalba LJ, López López V. Implementation and attacks analysis of a honeypot. *Computational Science and Its Applications—ICCSA 2007*. Lecture Notes in Computer Science. Springer: Berlin/Heidelberg, 2007; 489–502.
13. Almoitari S, Clark A, Mohay G, Zimmermann J. Characterization of attackers' activities in honeypot traffic using principal component analysis. *Proceedings of the 2008 IFIP International Conference on Network and Parallel Computing*, Shanghai, China, October 2008; 147–154.
14. Watson D, Riden J. The honeynet project: Data collection tools, infrastructure, archives and analysis. *Proceedings of WOMBAT Workshop on Information Security Threats Data Collection and Sharing*, *WISTDCS 2008*, Amsterdam, The Netherlands, April 2008; 24–30.
15. Brazilian Honeypots Alliance. Available from: http://www.honeypots-alliance.org.br/stats/.
16. NOAH Honeypots. Available from: http://stats.fp6-noah.org/top.php.
17. NOAH Honeypots. Available from: http://www.juniper.net/security/honeypot/.
18. JP's Honeypot. Available from: http://www.jpsdomain.org/infosec/honeypot.html.
19. Chamales G. The Honeywall CD-ROM. *IEEE Security and Privacy* 2004; **2**(2):77–79.
20. VMware Server 2.0. Available from: http://www.vmware.com/products/server/.
21. VTun—Virtual tunnels over TCP/IP networks. Available from: vtun.sourceforge.net.
22. Spitzner L. Honeypots: definitions and values of honeypots. Available from: www.tracking-hackers.com/papers/honeypots.html.
23. InternetWorldStats. Available from: http://www.internetworldsts.com.
24. Norwegian Honeynet Project. Available from: http://www.honeynor.no/.
25. The Spamhaus Project. Available from: www.spamhaus.org/.

AUTHORS' BIOGRAPHIES

Mario Marchese (S'94-M'97-SM'04) was born in Genoa, Italy in 1967. He got his 'Laurea' degree cum laude from the University of Genoa, Italy in 1992 and the Qualification as Professional Engineer in April 1992. He obtained his PhD (Italian 'Dottorato di Ricerca') degree in 'Telecommunications' from the University of Genoa in 1996.

From 1999 to 2004, he worked with the Italian Consortium of Telecommunications (CNIT), in the University of Genoa Research Unit, where he was Head of Research. From February 2005 he has been Associate Professor at the University of Genoa, Department of Communication, Computer and Systems Science (DIST). He is now the founder and the technical responsible for CNIT/DIST Satellite Communications and Networking Laboratory (SCNL) at the University of Genoa.

He chaired the IEEE Satellite and Space Communications Technical Committee from 2006 to 2008. He is the author and co-author of about 200 scientific works, including international magazines, international conferences and book chapters and of the book 'Quality of Service over Heterogeneous Networks', John Wiley & Sons, Chichester, 2007.

His main research activity concerns: Satellite and Radio Networks, Transport Layer over Satellite and Wireless Networks, Quality of Service and Data Transport over Heterogeneous Networks, Emulation and Simulation of Telecommunication Networks and Satellite components.

DOI: 10.1002/dac

Roberto Surlinelli was born in Savona, Italy in 1969. He got his degree in Electronic Engineering in November 1997 from the University of Genoa, Italy. From 2003 to 2005 he was the Main Technical Director for the Italian State Police in Palermo and Rome, Italy. Since 2006 he has been the Director of the Analysis Division in the Postal and Communication Police Department in Genoa, Italy. His main activities concern: network and computer security, digital forensic and computer crimes prevention and repression.

Sandro Zappatore was born in Savona, Italy. He received the Laurea and PhD degrees in Electronic Engineering from the University of Genoa, Italy, in 1985 and 1990, respectively. In 1990 and 1991, he was awarded two scholarships from the Italian National Council of Research (CNR) within a National Project on Telecommunications.

From 1992 he has been working in the Department of Communication, Computer and Systems Science of the University of Genoa, first as Assistant Professor and then as Associate Professor of telecommunications and teaches the course 'Digital transmissions'. His interests are both in the area of signal processing, especially audio and video coding, and computer networks. His current research is devoted to Multimedia Network Applications. In this field, he is the technical manager of some national projects, funded by the Italian Ministry of the Education, University and Scientific Research (MIUR), concerning the networked access and management of remote and complex laboratories. Currently, he is interested in grid-based platforms for the control of remote laboratories, in wireless sensor networks, and network security. He is the author of many papers, has appeared on international journals and on the proceedings of international conferences.

DOI: 10.1002/dac