

Technical and Human Issues in Computer-Based Systems Security

Budi Arief and Denis Besnard

Centre for Software Reliability
School of Computing Science
University of Newcastle upon Tyne
Newcastle upon Tyne NE1 7RU

Abstract

Computer systems and internet are becoming pervasive in our everyday life. Being online brings the consequence that such systems are prone to malicious attack. This vulnerability, along with our reliance on these systems, implies that it is important for us to do our best in securing them to ensure their proper functioning. In this paper, we are trying to tackle the security issues from both technical and human perspectives. From this dual standpoint, we hope to obtain a better understanding on how computer attacks are performed, including how to gain illicit access, the types of attacks, as well as the potential damage that they can cause. We also uncover sociological and psychological traits of the attackers, including their community, taxonomy, motives and work ethics. This survey paper will not provide a concrete solution on how to secure computer systems, but it highlights the socio-technical approach that we must take in order to obtain that goal.

Keywords: computer security, intrusion techniques, human actors, attackers' motives.

1. INTRODUCTION

Our society is becoming more and more dependent on computer systems, which nowadays are used in everyday life, from business to banking, from entertainment to healthcare. Most of these systems are interconnected through the internet, which inherently is very open and vulnerable to cyber-attacks. Attacks on these systems cause a lot of disruption at the very least, and may lead to severe financial or safety impacts. Therefore, securing computer systems has become a very important part of system design, development and deployment. Since we cannot eradicate these attacks, the aim of this course of action is to minimize their effects. This can be achieved by a better understanding of the technical and human issues involved in cyber-attacks. This knowledge, in turn, can allow one to make the attackers' life as hard as possible. This is our motivation in exploring the human aspects of attacks.

We often refer to attackers as "hackers", although "crackers" is probably a more appropriate term [28]. In this paper, we will not enter into this kind of debate, so several equivalent terms will be used: hackers, crackers, attackers, intruders. Hackers try to break into computer systems for various reasons. Some do it for personal gain (e.g. stealing data such as credit card details, which can then be sold to interested parties), corporation advantages (e.g. spying rival companies), or even just for the fun of doing it. These motives are discussed further in Section 4.3. By understanding hackers' motives, it is hoped that computer systems can be protected in a suitable way, hence improving their security.

The hackers' motives are only one of the many facets of computer security that needs to be investigated. As part of the DIRC¹ interdisciplinary research project on dependability of

¹ www.dirc.org.uk

computer-based systems, we are interested in “socio-technical” aspects of computing, and this should also be applied in securing those systems. For that reason, both technical aspects and human aspects of computer security are addressed in this paper.

The rest of the paper is outlined as follows: Section 2 provides a summary of hacking methods in technical terms; Section 3 outlines different players involved in computer security; Section 4 discusses our study in understanding hackers from human perspective, including their community, motivations, etc.; and Section 5 concludes our paper and highlights issues that can be addressed in order to make computer systems more secure.

2. TECHNICAL SIDE OF HACKING

Hacking is usually a technical activity, although that does not necessarily mean that attackers are always technically capable. Most of the attackers are *script kiddies*, who know just about enough in order to use other (more competent) hackers’ work. That fact aside, it is necessary to know the technical side of hacking, in order to understand the kind of knowledge that some attackers have. This section provides a brief explanation of how to get into a target system and how to exploit this as a full-scale hacking activity.

2.1. Cracking Computer Systems

There are many ways for attackers to obtain illicit access to computer systems. This kind of access is often called “intrusion”, and the first thing an intruder does is usually trying to obtain special/administrative privileges (a root access) on that system. Having a root access is very important for the attackers, since this means that they can do whatever they want on the system, including covering their tracks, strengthening their hold and doing damage.

In general, there are three main ways to intrude into a system [32]:

- Physical Intrusion
This kind of intrusion happens when an intruder has a physical access to the target machine. This might allow the intruder to gain full control of the system – for example by booting with a special floppy or by taking the system apart physically (e.g. removing the hard-drive to another system owned by the attacker, which then enables him/her to read/write to it).
- System Intrusion
In this case, it is assumed that the intruder has already got low-level privileges on the system. They then exploit un-patched security vulnerabilities in order to escalate their privileges to administrative level.
- Remote Intrusion
With remote intrusion, an attacker tries to get into the system remotely through the network. They initially do not have any privileges to the system, but one way or another – e.g. by finding out some valid account names and cracking their (usually weak) password, or by exploiting common security vulnerabilities (buffer overflow, etc.) – they manage to get in and obtain a root access.

This paper focuses on remote intrusion, as this is the most common type of attack associated with hackers. Nevertheless, there are some cases of system intrusion, for example, the insider attack, where a legitimate user (could be a disgruntled or former employee) performs an attack due to various reasons (revenge, cyber-espionage, etc.).

In order to minimize intrusion, many organisations install *Intrusion Detection Systems (IDS)*. Such a system inspects inbound and outbound network activity and identifies suspicious

patterns that may indicate a network or system attack from someone attempting to break into or compromise a system [30]. There are many IDSs available. Most of these are commercial software and are primarily concerned with remote intrusion. We will not discuss IDS in great detail in this paper since our focus is on the attackers along with their hacking activities and some insights into their human aspects.

2.2. Types of Attacks

Attackers can cause various levels of damage, depending on their skill level and/or their motives. There is a common pattern though: they usually follow a similar set of steps of information gathering before launching the attack.

2.2.1. Preparation for Attacks

In order to make a successful attack, attackers need to carry out some information gathering on the target, plan their way into the system, and at the same time, reduce the chance of getting caught. At this point, their activities do not seem to be dangerous since the network traffic generated will look normal. There are three main stages in information gathering, namely footprinting, scanning and enumeration [20].

- Footprinting

The aim of this activity is to obtain a complete profile of the target organisation's network and its security arrangement. The information of interest includes the technology that the organisation is using (e.g. Internet, Intranet, Remote Access, Extranet) and its security policies and procedures. Although there are many different methods attackers can use to perform footprinting, there are four general steps that they are likely to follow:

- Determine the scope of the footprinting activities

In some cases, it might be a bit too much to determine all entities associated to a target organisation. Therefore attackers often need to narrow down the scope of their footprinting activities. A lot of information about the target organisation (and its take on security) can be obtained from the internet, either from their web page or *Google* searches related to the organisation (news articles, press releases, etc.). This information could be useful in conducting social engineering (see later) to break into the system.

- Network Enumeration

In this step, attackers try to find out the domain names and associated networks related to the target organisation. There are many databases that provide this kind of information, such as the InterNIC database (run by Network Solutions – <http://www.networksolutions.com>), the American Registry for Internet Numbers (ARIN – <http://www.arin.net>) and *whois* databases (for example, <http://whois.ripe.net> and <http://whois.nic.gov>).

- Domain Name System (DNS) Interrogation

Once network enumeration is done (i.e. all the associated domains have been identified), an attacker can begin to query the DNS. A DNS that is not configured securely might reveal information about the organisation, or even allow untrusted internet users to perform a DNS zone transfer. A *zone transfer* is a mechanism for a secondary DNS server to update its zone database from the primary server (for redundancy). This operation should normally only be performed by the secondary server, but a misconfigured DNS may allow anyone to obtain a copy of the zone they ask for, which includes the public (external) and private (internal) DNS information. It is the leak of the private DNS information that helps attackers to roadmap the organisation's internal network, hence assisting them to carry out the attack.

- Network Reconnaissance
Now that the potential target within the system has been identified, the attacker can try to map the target’s network topology and identify potential access paths to that network. This can be done using the widely available `tracert` program, which lets you view the routes that an IP packet follows from one host to the next, hence providing the network topology as well as identifying access control devices (application-based firewall or packet-filtering routers).
- Scanning
If footprinting is comparable to casing an establishment for information, scanning can be considered as an equivalent to knocking on the walls to find all the doors and windows. The aim is to find what systems are alive and reachable from the internet using various tools and techniques such as ping sweeps, port scans and automated discovery tools [20]. At this stage, the network intrusion detection system will indicate that “someone is checking the doors and windows handles”, but nobody has actually tried to open them yet.
- Enumeration
This is a process of extracting valid account or exported resource names from systems. This involves active connections to the target systems (the previous two stages – footprinting and scanning – are not this intrusive), although the information collected through enumeration may appear to be harmless. After valid account names are found, an attacker can try to crack their passwords (which are too often weak or poorly protected) in order to get into the system.

After information gathering is completed, an attacker can then proceed to do the actual attack, which starts with gaining unauthorized access to the target system.

2.2.2. *Unauthorized Access*

In most cases, before attackers can exploit a system, they need to gain access to it. There are many methods to do so, namely by acquiring password, through software bugs, through system configuration bugs and by using malicious programs.

- Acquiring password
One way to get into a system illegally is by figuring out the password of a valid account. Valid account names can be those of default accounts, such as “Administrator” or “Guest”, or those gathered earlier by performing enumeration as mentioned above. The corresponding password for these accounts can be obtained in various ways [32]:
 - Clear Text Sniffing
Several protocols such as telnet, FTP and HTTP Basic do not encrypt the password at all as it is passed from the client to the server. In other words, the password is sent as a clear text, which makes it easy for attackers to eavesdrop the network using a protocol analyzer to obtain the password and use it immediately to get in.
 - Encryption Sniffing
Most protocols, however, do encrypt the password. This means that the attacker will need to decrypt the password, for example using dictionary or brute force attack. There are many tools available for this purpose, such as, L0phtcrack3 (LC3), which performs dictionary, brute force or hybrid cracks.
 - Replay Attack
Sometimes, the attackers do not need to decrypt the password at all. By reprogramming the client software, they can use an encrypted password to log in to the system.

- Password File Stealing

In most operating systems, the entire user database (including the passwords) is stored in a single file, such as `/etc/passwd` (in UNIX) or SAM – Security Accounts Manager (in WinNT). The attacker can then just steal this file and run cracking programs to find some weak passwords in that file.

- Observation

To increase the security of their passwords, people use long and difficult-to-guess passwords. This makes it more difficult for attackers to run dictionary or brute force crack. On the other side, these passwords are more difficult to remember, and as a consequence, people often write them on a piece of paper or even in an electronic format. Attackers with physical access to the vicinity of the system can then find the password by looking for that piece of paper under the keyboard, stuck on the keyboard, or maybe in the bin (a term for it is “dumpster diving”). They can also get the password by milling around the system to watch legitimate users typing in their password (shoulder surfing).

- Social Engineering

Social Engineering is the term used to describe cracking techniques that rely on weaknesses in *wetware* (i.e. human users attached to the system – administrators, operators, etc.) rather than software. The aim is to trick people into revealing passwords or other information that compromises a target system's security [17]. One example is phoning up the system's operator (who has the required information), posing as a field service technician or a fellow employee with an urgent access problem. There are various methods to perform social engineering (as described in [13]): false authority, impersonation, sympathy, personal stake, boosting egos, inconspicuous occupation, and reward. It is surprising how effective social engineering can be, as illustrated in detail in Mitnick's book [22].

This section discussed the many techniques attackers can use to get into systems by obtaining the passwords of existing accounts. It is therefore important to have strong passwords and to protect them properly (e.g. by not writing them down, and not divulging them to anyone). However, this does not guarantee that your system will be safe as there are still other ways to get into and/or exploit them (see below).

- Software Bugs

Another way to get into a system is through security vulnerabilities brought by bugs in the software (operating system, server daemons, client applications, etc.). It is almost impossible to have bug-free software and the attackers only need to find one hole in order to break in. Software bugs differ in cause and severity [32], but the most common one is the *buffer overflow* bug. A buffer is a holding area in memory, and a buffer overflow happens when someone tries to enter more data into the buffer than it can handle. As a result, the program may crash and very often, this gives the attackers a root access and/or allows them to run any arbitrary code. Attackers can find buffer overflow bugs by:

1. Browsing the web for known buffer overflow vulnerabilities on certain programs;
2. Searching for these bugs in the program directly if the source code is available;
3. Examining every place the program prompts for input and trying to overflow it with random (massive) data. If the program crashes, there is a chance that by carefully constructing the input, access to the system can be obtained.

For a comprehensive introduction to buffer overflow, see [23].

- **System Configuration Bugs**
 The way a system is configured may lead to it being insecure. Four of the most common configuration bugs are [32]:
 - **Default Configurations**
 Most systems are delivered to the customers with default, easy-to-use configurations. As a consequence, it is easy to break into these systems since the default configurations for a given manufacturer are well known to the public and are usually weak in term of security.
 - **Poor System Administration Practices**
 Some system administrators are too lazy to configure the root password properly when they set up the machine. They want to get the machine up and running quickly so they often leave the root password blank, with the intention to change it later. Unfortunately, they might forget to do so, and this provides an attacker with an easy target.
 - **Running Unnecessary Services**
 There are services that are inherently insecure (e.g. telnet and ftp deemons) or require some effort to make them secure (e.g. web server). The best policy here is to turn off everything that is not absolutely needed to run on the system, in order to avoid accidental holes.
 - **Trust Relationships**
 A group of users often trust each others' machines through shared file systems, etc. This kind of trust relationships pose a security risk since it allows attackers to "island hop" through the network to gain access to many computers with minimal effort.

- **Malicious programs**
 Another means that attackers can use is in the form of malicious programs. Very often, these programs require legitimate users' cooperation or ignorance in order to be installed and run on the target systems. Malicious programs can be classified into [13]:
 - **Trojan Horses**
 This type of programs are designed to circumvent the security of the target system but are disguised as something benign (e.g. screen savers, anti-virus tools, etc.). They seem to provide something useful or appealing, but they also contain some function that either creates or exploits some security hole in the system. They become dangerous only if the user downloads, installs and runs them on his/her system. This allows the attackers to read and/or write files, create network connections, attempt to break into other machines and run any arbitrary commands. There are two further variations to trojan horse programs:
 - **Trojaned source code**
 Many programs nowadays come with their source code (open source, free software), or the source code is available to download from the web or FTP sites. There are occasions where the source code has been replaced with a trojaned version, which looks like it does what it's supposed to do, but actually it also has extra code to bypass the security. It is not easy for attackers to create trojaned source code, since they would have had to compromise the software's distribution site or find some way to corrupt the software before it is installed or compiled on the target system.
 - **Trojaned binaries**
 After a successful intrusion, attackers may replace system binaries with trojaned binaries that contain backdoors (to allow them to exploit the system again later)

or hide their activities (e.g. hiding files/processes/connections, modifying/erasing the log files, etc.).

One way to detect trojaned source code and binaries is by performing checksum on the software, for example using MD5 checksum tool.

– Viruses

Viruses are similar to trojan horses in that they perform something undesirable on/to the target system without user's knowledge or permission. In addition, once a virus is activated, it will infect other programs on the computer. In contrast, trojan horses are stand alone programs and will not propagate themselves. A virus may spread to other computers as well, but only with help or lack of awareness from the human user.

– Worms

Worms are similar to viruses, but they can infect both local and remote machines. In other words, a worm spreads itself automatically by attacking or using other network programs or by using file-sharing features of the computer. This means that worms pose much greater threat to systems because they do not rely on the gullibility of the users.

Most malicious programs are actually a hybrid of the three categories above.

As we can see above, there are many techniques an attacker can use in order to get access to a system. However, many of them are only possible by exploiting the weaknesses of the wetware (i.e. human user). The human aspect of computer system is therefore important to be investigated and understood, so that we can improve its security. Before we get into this though, let us have a look at the damage that a successful attack can cause.

2.2.3. *Damage Caused by the Exploits*

At this point, the attacker has control of the target system, which means that he/she is free to exploit it to cause damage. Howard, in his Ph.D. thesis [16] categorises the damage into four groups. These are seen as the *results* of attack, although they are not the final objectives of the attack itself (the objectives are considered as the *motives* of attack and discussed in Section 4.3).

- Corruption of Information

This includes any unauthorized alteration of files stored on the target computer (e.g. web defacement, deletion of files, etc.) or corruption of data in transit across the network.

- Disclosure of Information

The attacker may wish to disseminate the (classified) information gathered during the attack.

- Theft of Service

This term is used to describe an unauthorized use of computer or network services without degrading the service to other (legitimate) users. The attackers exploit this for many purposes. For example, they might use the target computer as a storage space or exploit the network bandwidth for downloading files.

- Denial of Service (DoS)

In contrast to theft of service, denial of service is an intentional degradation or blocking of computer network resources, which makes it hard or impossible for legitimate users to obtain the intended service. This kind of attack has become very popular, with many big corporations' web services becoming the target. A major difference of DoS, as compared to the other three results of attack above, is that it does not require the attackers to actually gain access of the target system. The attackers might still want to gain access to another system first before launching DoS from this compromised system, in order to

make it more difficult to track their identity. There are four common types of DoS attacks [20]:

- **Bandwidth Consumption**
Here, the attacker tries to consume all available bandwidth of the target network. There are two scenarios for this method. If the attackers have more bandwidth than the target network, they can flood and completely saturate the victim's network with traffic (this is another reason why attackers want to gain access to other systems, especially those with large bandwidth. Even if the attackers do not have a large bandwidth, they can *amplify* their DoS attack by using multiple sites to flood the target's network (Distributed Denial of Service or DDoS attack).
- **Resource Starvation**
Instead of consuming the target network's bandwidth, resource starvation attack focuses on consuming the target system's resources such as CPU utilization, memory, disk space, process allocation, etc. As a result, the resources become unusable since the system crashes, the file system becomes full or the processes hang.
- **Programming Flaws**
By exploiting programming flaws (i.e. failures of an application, operating system, or embedded logic chip to handle exceptional conditions), attackers can cause the target system to crash. Malformed packets are sent, which cannot be handled by the application, leading to a kernel panic and complete system crash.
- **Routing and DNS Attacks**
In routing-based DoS attack, the attackers manipulate routing table entries to deny service to legitimate systems or network by altering the legitimate routes. This causes the victim's traffic to be routed through the attackers' network or into a non-existent network (*black hole*). This denies the actual service offered by the victim's system. Routing-based DoS attacks are possible because most routing protocols have no or very weak authentication.
DNS-based DoS attack involves convincing the victim DNS server to cache bogus internet address information (*DNS cache poisoning*). When a lookup operation is performed, the DNS server will return the bogus address, hence redirecting the traffic to the site chosen by the attacker or even to a black hole.

DoS attack is relatively easy to perform and can cause more disruption than other attacks. Hence, it is quite popular among script kiddies, hackers with limited talents or even cyberterrorists. For details and example of DoS attacks, as well as possible countermeasures, see [20].

Security exploits can cause damage to varying levels of severity, in term of confidential information breached, services stolen and disruption caused. It is therefore very important to minimize these threats by making our systems more resilient to them.

Currently available security solutions, such as Intrusion Detection Systems, anti-virus programs, firewalls, and cryptography, only deal with technical aspects of security. Although these certainly help, there is still a need to improve them. One thing that is often overlooked is the human aspect in computer security. To say the least, the attackers are human, with psychological and sociological issues that are interesting to address. Moreover, in a lot of cases, the attackers do take advantage of human weaknesses – for example, through social engineering or by exploiting bad system administrating practices – in order to perform their attacks.

Therefore, in order to create more secure systems, we must combine technical skills with socio-technical knowledge. In doing so, we must consider the human aspects in security of computer-based systems, which are discussed in the next section.

3. HUMAN PLAYERS IN COMPUTER SECURITY

As highlighted in previous section, an attacker with good technical skills can potentially gain full control of a target system. Not only is this a bad news for the system's owner, but it could also worry others, as the attacker might use the compromised system to attack other systems, for example to launch a Denial of Service attack.

It has also been mentioned that humans play important roles in computer security. This section provides a brief outline on the people involved in computer security. These people play different roles, but mainly, they belong to three sides: those who are users of a system, those attacking computer systems and those defending them. On top of these groups, we speculate that there is possibly another group that consists of people who behave like a double agent: they might secure a system but at the same time, they leave or create a hole for their own advantages.

- **Protectors**
People in this group are trying to protect their systems or helping to protect other people's systems from attack. Included in this category are system administrators, bug fixers, anti-virus and security companies, as well as security response teams such as CERT[®] Coordination Center [1] and SANS Institute [2].
- **Attackers**
These include anyone who aims to use computer systems illegally or cause disruption to those systems. This group is the focus of our paper, and will be discussed further in Section 4.
- **Users**
They are people who can legally use the system, and are usually given certain privileges. Some of them might turn out to be a protector (by helping to secure the system, either in their capacity as system administrator, or by following the security guidelines). It is surprisingly common though that a user might turn into an attacker. This kind of attacker is often called *insiders* and they can do a lot of damage due to inside knowledge (e.g. they know the weaknesses of the system or they have high privileges).
- **Double Agents**
We speculate that there is another group that behaves like a double agent. People belonging to this group appear to be helping to improve security, but they actually have a hidden agenda behind it. For example, someone could be offering security products on the web (such as anti-virus, firewall, etc.) but these programs actually contain some backdoors, which allow unauthorized access to the computer on which the programs are installed.

The rest of this paper will focus on the attackers' side, trying to understand their community, taxonomy and motives, as well as providing some arguments why this kind of people have some advantages over protectors type. By gaining better understanding on the attackers, we hope to highlight areas that can be addressed in order to build more secure systems and more efficient security policies.

4. UNDERSTANDING HACKERS

In order to better understand how computer attacks are conducted, it is very important to gain insight into what drives people to do those attacks. This will allow us to understand the way they work, their community, motives, etc., which will then enable us to protect our systems in a more efficient way. Some of the issues addressed here have, to some extent, been addressed in the previous sections of this paper. Nevertheless, they will be presented now under a different angle.

4.1. Hacker Community

4.1.1. Who are hackers?

Quoting Steele [26], Denning [9] reports that the word *hacker* has taken on many different meanings ranging from a person who enjoys learning the details of computer systems and how to stretch their capabilities to a malicious or inquisitive meddler who tries to discover information by poking around possibly by deceptive or illegal means. Interviewing hackers gave Denning a more precise definition. One of them asserted:

"A hacker is someone that experiments with systems... [Hacking] is playing with systems and making them do what they were never intended to do. Breaking in and making free calls is just a small part of that. Hacking is also about freedom of speech and free access to information – being able to find out anything. There is also the David and Goliath side of it, the underdog vs. the system, and the ethic of being a folk hero, albeit a minor one."

The term “hacker” itself has evolved somewhat from its original meaning. Rogers [24] breaks down the evolution into four generations:

- First generation: talented students, programmers and scientists (mostly from MIT)
These were academics or professionals interested in the working of computer code. They enjoyed tweaking the code, in order to produce more efficient or elegant program, or just to create program that can assist them in everyday computing life. They were the technically elite group and were often pioneers in their field (e.g. Richard Stallman).
- Second generation: technological radicals
They evolved from the technically elite, with forward thinking to recognize the potential of a second computer niche from mainframe to personal systems. Their radical nature means that minor criminal activity was not uncommon (e.g. phone phreaking [31], etc.).
- Third generation: young people who embraced personal computers (PC)
These people recognized the potential entertainment value of PC and began developing games (or making illegal copies of games and cracking their copy-protection).
- Fourth generation: the current generation, those embracing criminal activity as if it is some sort of game or sport.
Some of these people got arrested and claimed that the motivation was curiosity or hunger for knowledge, although the actual motivation seems to be greed, power, revenge or other malicious intent.

The public perception is that hackers are usually young males who tend to be working alone. With the proliferation of both the bulletin boards and of tool kits for virus makers, the last characteristic is questionable [7]. On the values side, hackers share some particularities. For instance, Levy [19] identified some behaviours and values in the *hacker ethic*, some of them having an obsession with “hands-on” use of computers, desire that all information should be in the public domain and mistrust of authority. Some more personological dimensions have been identified by Gordon [11] who examined the ethical development of a large number of

virus writers. According to Gordon's results, hackers can be divided into the following categories:

- adolescent, who is not typically concerned by problems caused by their viruses;
- adult, ethically abnormal;
- ex virus-writer, undecided concerning the legitimacy of virus writing [12].

4.1.2. Hackers' ethics

Some hackers' ethics are not that far from more traditional views shared by the public at large. For instance, some hackers are concerned about the increasing amount of information about individuals that is stored in large data banks, and the inability of the individual to have much control over the use of that information. In this view, hackers say that it is our social responsibility to share information, and that it is information hoarding and disinformation that are the crimes. Some hackers say they are outraged when peers cause damage or use resources that would be missed, even if the results are unintentional and due to incompetence. Even among the hackers' community, some break-ins are considered unethical (e.g., breaking into hospital systems) as well as reading confidential information about individuals, stealing classified information and committing fraud for personal profit [9].

It is often said that breaking into a system implies a lack of caring for the system's owner and authorized users. But Denning [9] reports that one hacker said that the ease of breaking into a system reveals a lack of caring on the part of the system manager to protect user and company assets, or failure on the part of vendors to warn managers about the vulnerabilities of their systems. Unfortunately, hackers are not the only ones to think along that line. Risto Siilasmaa (President and CEO of F-secure; interviewed by Armstrong [4]) thinks that "the danger provided by viruses is in direct proportion to the complacency that seems so prevalent today". Of course, there is the view that hackers cause damage by increasing the amount of paranoia, which in turn leads to tighter security controls that diminish the quality of life for the users. Hackers respond to these points by saying they are the scapegoats for systems that are not adequately protected. There is obviously a wide range of arguments standing between these two positions.

4.1.3. How do hackers work?

Socially speaking, the activity of spreading malicious code is highly worrying because the number of targets that can be easily harmed by email-transmitted code is absolutely huge. Moreover, once a virus has entered the public domain, it is a much simpler task to alter the existing virus rather than to invent a completely new one, and this potentially augments its destructive effects [7]. This partly explains Meckbach's data [21] (quoted by Bissett & Shipton in [7]) according to which 12,000 different viruses have been launched in 1997. And these are not isolated figures. Later, Gordon [12] reports that between the 15th of March 2000 and the 18th August 2000, a Google search on viruses showed an increase of 28% in the number of the pages making live viruses available. So it may be wise to follow Denning's advice according to which we should not underestimate the effectiveness of the networks in which hackers learn their craft. They do research, learn about systems, work in groups, write, and teach others. Hackers share information and operate as communities and this is a significant part of the problem. According to Dave Kroll (Director of security research for Finjan Software; see [4]), "Gone are the days of our image of the rocket scientist hacker. Nowadays, 12 years-old can download a worm generator tool and point and click their way to a devastating worm".

4.2. Hacker Taxonomy

Rogers [24] did a study on hacker community and he proposed hackers taxonomy as follows:

- *Tool Kit/newbies* are people who are new to hacking and who tend to have limited computer and programming skills. As a result, they usually rely on software (tool kits) already written by other people in order to conduct their attack. These people are often called *script kiddies*.
- *Cyber-punks* consist of people with better computer skills (than newbies) and some might have programming capabilities. It is their intention to engage in malicious acts, such as web defacement, credit card number theft, telecommunication fraud or spamming (sending junk mails).
- *Internals* are made up of disgruntled employees or ex-employees. It may be surprising to learn that a lot of attacks were actually carried out by people with internal knowledge/connection.
- Coders
- *Old guard hackers* are similar to those of first generation hackers. They appear to be interested on the intellectual endeavor instead of criminal intention, but they seem to have little respect for personal property.
- Professional criminals
- Cyber-terrorists

These groups are not mutually exclusive, but they might help in understanding the motives of those individuals involved in hacking activities.

4.3. Hacker Motives

It is always tricky to obtain a precise answer to the question “*Why do hackers do that?*” There will be many different motives, but in general, they can be divided into two categories: *sociological/psychological* motives and *technical* motives. The former lays deeper in the human aspect of the hackers, whereas the latter usually stem from the former.

4.3.1. Sociological/Psychological Motives

Jordan and Taylor [18] compiled a list of reasons that might motivate someone to hack into computers. These are internal accounts of the motives, as gathered from interviewing the hackers themselves:

- Addiction to computers and/or computer networks, feeling compelled to hack
- Curiosity
- The thrill of illicit searches in online life to compensate boring offline life
- The attraction of the ability to gain power over computer systems
- Community peer recognition
- Service to future computer users or society by identifying security loopholes in computer systems

In his book [27], Taylor replaced the last motive with another: political acts. Some hackers oppose the re-establishment of traditional values that are based upon physical property rights in the newly emerging information society. This anti-authoritarian stance is fuelled by their dislike of government information gathering bureaucracy as well as the unjustified privatisation of information by big corporations.

Citing Chantler’s survey of 164 hackers [8], Denning [10] outlines the top motives as follows:

- 49% were positive aspects beneficial to discovery learning, such as challenge, knowledge and pleasure

- 24% were recognition, excitement (of doing something illegal), and friendship as their motives
- 27% were self-gratification, addiction, espionage, theft, profit, vengeance, sabotage and freedom

These two studies highlight the differences between how hackers perceive themselves and how those outside hackers' community see of the hackers. Internal accounts tend to portray hackers as curious, knowledgeable, if not bored people who are up to new challenges and exciting adventures. On the other hand, outsiders perceive them as malicious, arrogant or even dangerous bunch of irresponsible people.

4.3.2. Technical Motives

The *Honeynet* project [14, 15] is a non-profit research group that aims to learn the tools, tactics, and motives of the blackhat community (i.e. malicious hackers) and share the lessons learned. This project developed *honeypots* [29], which are used to study hackers' activities and monitor how they break into systems. Among other things, the honeypots managed to capture some *Internet Relay Chat* (IRC) discussion sessions involving several hackers. These logs are valuable because from analyzing them, several technical motives for hacking were uncovered:

- To launch Distributed Denial of Service attacks
- To hide their source and identities (using many compromised systems as stepping stones, making it more difficult to trace)
- To maintain administrative rights on IRC, which is also useful for the communication among attackers
- To win bragging rights
- To steal the bandwidth (using the network and storing files for free)

Along with these technical motives, the IRC sessions captures also provide a confirmation on the sociological and psychological motives outlined in the previous sub-section, namely: political agenda (hacktivism) and personal motives (e.g. for fun, just to cause mischief, etc.).

4.4. Why do attackers have many advantages?

An analogy to warfare can be made with regard to computer security in cyberspace. There are mainly two opposing parties: the attackers and the protectors (security professionals). In this case, one party – the protectors – is limited in its involvement to primarily perform the defense. The hackers, on the other hand, are the ones doing the offense, and they have several advantages while doing it [25]:

- Relative mobility
Not being fixed to a particular location in cyberspace allows an attacker to launch his/her attack from anywhere. This makes it difficult for the security professionals, as well as the victims and the law enforcement people to find the attacker, let alone arrest him/her.
- High level of knowledge sharing
The hacker community is very accomplished at sharing their knowledge and tools of the trade. There is also a loosely defined hacker code of ethics, which encourages them to bond together. In contrast, security professionals' common bond, in most cases, is their job to stop the attack. There might even be some bickering on how they should conduct their duty.
- Intensity
Hackers usually are prepared to spend many more hours in conducting their attack than most security professionals are willing in securing their systems. As a result, even though

hackers are under-funded (and maybe even under-educated) compared to system administrators, their dedication, passion and patience often compensate that lack.

- Relative lack of assets
Being under-funded actually has a huge advantage: the attackers have very little to lose. In comparison, companies usually put a lot of investment on their online services. If these services are under attack (e.g. from a Denial of Service attack), it is bound to cost the company dearly.
- General complacency on the part of Internet Service Providers (ISPs) and software vendors
This ensures that software vulnerabilities continue to be created. Also, ISPs tend to be lax in preventing or following up on security incidents. Very often, they leave widely known security vulnerabilities un-patched, which in turn makes it easier for the attackers to exploit these holes.
- Advantages of attack (potential surprise, economy of effect)
The attackers have an advantage in term of a potential surprise of the attack. Attacking is also easier to perform, in the sense that the defender has to defend every possible point of entry whereas the attacker can focus on a shortlist of targets.

Because of these advantages, it is difficult to completely eliminate attack and/or bring repercussion to those responsible. There are several cases where governments managed to arrest some attackers, but only in very few cases were they able to successfully prosecute them. One of those successful prosecutions is that of a notorious hacker, Kevin Mitnick. He was arrested in 1995 and sentenced to 46 months in prison in 1999 [3]. More recently, a virus writer from Wales called Simon Vallor has been jailed for two years in January 2003 for infecting thousands of computers around the world with “mass-mailer” viruses that he created [5].

5. DISCUSSION

Public vision of hackers is of those who want to steal or exploit valuable information. This means that people often do not bother with securing their system because they think there is nothing valuable to steal from it. This is not a totally correct picture: hackers do get into systems for other purposes, for example, using them as storage or launching pad for DOS. Moreover, hackers sometimes intrude into systems as a mere challenge with the sole intention of getting in. The harm is that it constitutes a practice that is not valuable for us to offer them. Therefore, it is important to educate people so that they treat computer security as a serious matter. Obviously, the blame, if any, is not on the end-users themselves. However, a better understanding of the threats might provide end-users with some real protection power, in the form of better-informed practices.

However obvious it may seem, the previous point is a critical one given that systems are intrinsically and globally imperfect. There are many holes and these holes are not just technical ones (bugs in programs, etc.). They also stem from bad security practices and procedures. Protecting a system just from a technical standpoint (e.g. firewall, antivirus, IDS) is not sufficient to minimize the threat of attacks. We need to employ security policy that takes into account socio-technical issues. This means that we must also (among other things) educate system administrators. They have to acknowledge that contrary to the technical protection, human components are not deterministic. The way the latter apply procedures is un-enforceable: they might deviate from prescribed practices in a way that is difficult to foresee. For this reason, as acknowledged by Besnard & Arief in [6], security policies must

be designed by taking into account their usability and practical aspects. Security should not only be seen as protection measures but also as a set of rules that have an implementation cost. The higher the cost, the less likely the implementation will be. Security must be effortless for end-users. If this not the case, some rules will be violated, thereby creating or maintaining breaches.

Lastly, because it has not the visibility it deserves, we wish to emphasize the social side of security. Let us assume that the human threats are both from external and internal sources. To us, a very dangerous external threat is social engineering. The internal threats include flawed practices or badly designed policies, disgruntled employees, as well as gullible users. The latter could inadvertently help the hackers through social engineering by disclosing seemingly benign information, which in a wrong hand, can severely breach the security. In our opinion, social engineering seems to be a highly dangerous threat because it directly targets the human components of a system. In other words, social engineering could bypass technical security measures altogether. With this form of attack, we reach the boundaries of the efficiency of technical measures. This is where we think human aspects of security are best highlighted, imposing the adoption of a multidisciplinary approach to security. Social scientists and psychologists have to assist in devising security policies, spot and alter bad practices in order to impact on the social component of protection of socio-technical systems.

6. CONCLUSION

This paper provides a comprehensive survey on human and computer security. It goes over the technical aspects of an attack, including the steps that an attacker usually takes, some methods for obtaining illicit access into a system and potential damage such an attack can cause. The paper also offers some insights into the human aspects of computer security, most notably from the attackers' side. This covers attackers' community, taxonomy, motives, and the advantages that they have. It is hoped that by taking into account both the technical and human aspects of computer security, we can design and run computer systems that are more secure without sacrificing its usability.

At the end of the day, it is up to us to secure our own system in order to minimize the chance of it being attacked in the first place. It has been shown that the wetware (human user) is one of the weakest points in security, being prone to social engineering attack. With that in mind, there are several measures that we need to take:

- **Securing the Environment**
This includes the installation of an intrusion detection system, a firewall and an anti-virus program. It is also important to fully utilise the IDS, to check the logs of the firewall to see if there is any suspicious traffic coming to/from the system, and to keep the anti-virus program up to date.
- **Educating Security Administrators**
The security administrators are expected to perform most – if not all – of the activities in securing the environment above. Therefore, they need to be trained on configuring security policies, informed about the actual threat out there, as well as kept up to date on the recent security alerts and patches.
- **Educating Users**
As demonstrated by social engineering, a gullible user could be an Achilles' heel of system security. For that reason, it is necessary to educate the users so that they will follow security policy guidelines and do their best to help maintaining the security. Kevin Mitnick, in his book *The Art of Deception* [22], emphasises the importance of information

security awareness and training. He also recommends steps and procedures that one needs to take in order to avoid being a victim of a social engineering attack.

If it is true that hackers do hack systems only for satisfying personal motivations (challenge, curiosity, causing disruption) then it is fairly obvious that the battle will never end. Thus better security practices and policies are needed. They may stem from a better understanding of the technical and human issues involved, for which we hope this paper has provided some insight.

7. ACKNOWLEDGEMENT

This paper was written at the University of Newcastle upon Tyne within the DIRC project (<http://www.dirc.org.uk>) on dependability of computer-based systems. We would like to thank Peter Ryan and Jeremy Bryans as well as anonymous reviewers for useful comments. We are also grateful to the sponsor EPSRC for funding this research.

8. REFERENCES

- [1] “CERT® Coordination Center”, online at <http://www.cert.org/>.
- [2] “SANS Institute”, online at <http://www.sans.org/>.
- [3] “Takedown”, online at <http://www.takedown.com/>.
- [4] Armstrong, I., “Viruses: Preparing for the Onslaught”, *Secure Computing, May issue*, pp. 24-30 (2001).
- [5] BBC-News, “Computer Virus Author Jailed”, online at <http://news.bbc.co.uk/1/hi/wales/2678773.stm>.
- [6] Besnard, D. and B. Arief, “Computer Security Impaired by Legal Users”, in *preparation* (2003).
- [7] Bissett, A. and G. Shipton, “Some Human Dimensions of Computer Virus Creation and Infection”, *International Journal of Human-Computer Studies*, Vol. 52, pp. 899-913 (2000).
- [8] Chantler, N., “Profile of a Computer Hacker”, Faculty of Law, Queensland University of Technology, Australia (1996).
- [9] Denning, D., “Concerning Hackers who Break into Computer Systems”, *Proc. 13th National Computer Security Conference*, Washington, D.C., pp. 653-664 (1990).
- [10] Denning, D., *Information Warfare and Security*, Addison Wesley - ACM Press Books (1999).
- [11] Gordon, S., “The Generic Virus Writer”, *Proc. International Virus Bulletin Conference*, Jersey, Channel Islands, pp. 121-138 (1994).
- [12] Gordon, S., “Virus Writers: The End of The Innocence?”, *Proc. International Virus Bulletin Conference* (2000).
- [13] Hatch, B., J. Lee, and G. Kurtz, *Hacking Linux Exposed: Linux Security Secrets & Solutions*, Osborne/McGraw-Hill (2001).
- [14] HoneyNet-Project, “The HoneyNet Project”, online at <http://www.honeynet.org/>.
- [15] HoneyNet-Project, *Know Your Enemy: Revealing the Security Tools, Tactics, and Motives of the Blackhat Community*, Addison-Wesley (2001).
- [16] Howard, J., “An Analysis Of Security Incidents On The Internet”, Carnegie Mellon University, USA, PhD thesis (1997).
- [17] Jargon-Dictionary, “Social Engineering Definition”, online at <http://info.astrian.net/jargon/>.
- [18] Jordan, T. and P. Taylor, “A Sociology of Hackers”, *Sociological Review*, online at http://fc.vdu.lt/Conferences/INET98/2d/2d_1.htm, Vol. 46, No. 4, pp. 757-780 (1998).

- [19] Levy, S., *Hackers, Heroes of the Computer Revolution*, Penguin Books (1994).
- [20] McClure, S., J. Scrambray, and G. Kurtz, *Hacking Exposed: Network Security Secrets & Solutions*, Osborne/McGraw-Hill (1999).
- [21] Meckbach, G., “Viruses Growing out of Control”, *Computing Canada*, Vol. 23, No. 15, pp. 1-2 (1997).
- [22] Mitnick, K. and W. Simon, *The Art of Deception: Controlling the Human Element of Security*, Wiley (2002).
- [23] Nelißen, J., “Buffer Overflows for Dummies”, online at <http://rr.sans.org/threats/dummies.php> (2002).
- [24] Rogers, M., “Psychology of Hackers: Steps Toward a New Taxonomy”, online at <http://www.infowar.com/hacker/99/HackerTaxonomy.shtml> (1999).
- [25] Stackhouse, B., “Why Do Hackers Have the Advantage?”, online at http://rr.sans.org/hackers/hackers_advantage.php.
- [26] Steele, G. L., D. R. Woods, R. A. Finkel, M. R. Crispin, R. M. Stallman, and G. S. Goodfellow, *The Hacker’s Dictionary*. New York, Harper & Row (1983).
- [27] Taylor, P., *Hackers: Crime in the Digital Sublime*, Routledge (1999).
- [28] Webopedia, “Hacker Definition”, online at <http://www.webopedia.com/TERM/h/hacker.html>.
- [29] Webopedia, “Honey-pot Definition”, online at <http://www.webopedia.com/TERM/h/honey-pot.html>.
- [30] Webopedia, “IDS Definition”, online at http://www.webopedia.com/TERM/i/intrusion_detection_system.html.
- [31] Webopedia, “Phreaking Definition”, online at <http://www.webopedia.com/TERM/p/phreaking.html>.
- [32] Wilson, Z., “Hacking: The Basics”, online at http://rr.sans.org/toppapers/hack_basics.php (2001).