Research Article

Harmonising Vulnerability Categories

HS Venter JHP Eloff[a]

Department of Computer Science, University of Pretoria, South Africa, [a]eloff@cs.up.ac.za

Abstract

A legion of vulnerabilities are potentially compromising the security status of IT industries infrastructures today. Current state-of-the-art intrusion detection systems (IDSs) can potentially identify some of the vulnerabilities. Each IDS defines its own and unique list of vulnerabilities, making it cumbersome for organisations to assess the completeness and reliability of vulnerability scans. What This furthermore complicates the matter of determining the degree to which a specific IDS complies to with the security requirements of a specific organisation. This paper presents an approach to harmonise different sets of vulnerabilities as currently used by state-of-the-art IDS tools.

Keywords: *Intrusion Detection System (IDS), vulnerability, vulnerability categories, vulnerability scanner, network security, firewall, Internet information security, scan.*

Computing Review Categories: *C.2, H.1.1, K.6.5, D.4.6, K.4.2.*

1 Introduction

Everyone will agree that the Internet has changed our lives dramatically in the past decade. Almost any conventional publishing media such as books and magazines can all be located on the Internet in electronic form these days. The Internet has made life easier in many ways – it has become part of our lives. But this is only half of the story; this is the side of the Internet that everyone can and is *supposed* to see. The "other side" of the Internet, however, is the part that the owner of a web site does *not* want everyone to access, but only those who are authorised to do so!

Furthermore, one should accept that there are always unauthorised intruders who *want* to illegally access the "restricted side" of the Internet. Reasons for this include stealing of information for unethical purposes, or simply jeopardisinge the organisation by making their system resources unavailable. It is for these reasons that a new research field has evolved – **Internet information security**.

Over the past decade various Internet information security tools and techniques have been proposed and implemented to try and keep intruders out. One such tool, known as a **firewall**, is typically used as a *first-line-of-defence* tool. The typical *second-line-of-defence* tools are known as **intrusion detection systems (IDSs)**. Figure 1 shows a typical configuration of how a firewall and IDS typically fit into a the network architecture of an organisation.

IDSs all contain some sort of signature database. A *signature database* contains the specific patterns or modus operandi used to identify *known* vulnerabilities. IDSs, however, have not solved all Internet information security problems. IDSs still have numerous problems: their signature database must be kept up-to-date at all times, they require human operators with technical skills to operate it them sufficiently, and they are *not* intelligent enough to make successful decisions on whether certain combinations of events are intrusions or not or what actions should be taken when intrusions do occur.

In the remainder of this paper, a short background of IDSs is given. The concept of harmonising different sets of vulnerabilities into **harmonised vulnerability categories** is then introduced, followed by a discussion of each harmonised vulnerability category with examples in a bid to demonstrate the usefulness of the proposed harmonised vulnerability categories. Finally, the article illustrates the application of harmonised vulnerability categories and concludes with how this approach can benefit an organisation.

2 Intrusion Detection Systems

The architecture for most currently available IDSs is shown in Figure 2

Current IDSs can be considered as **reactive** or **proactive**. *Reactive* IDSs scan events occurring in a computer system or network, analysing them for signs of vulnerabilities in a bid to detect them *as soon as they occur* [1]. **Proactive** IDSs, on the other hand, scan for known vulnerabilities on a computer system or network by *simulating* intrusions in a bid to see how the network and hosts would react against the intrusions, and generate a report of the findings. The difference between proactive and reactive IDSs is that proactive IDSs attempt to minimise the likelihood that intrusions will occur *beforehand*, whereas reactive IDSs attempt to detect an intrusion as soon as it occurs. Therefore, proactive IDSs are sometimes called *vulnerability scanners*; they are simply a *proactive* form of detecting intrusions. The authors are referring mainly to vulnerability

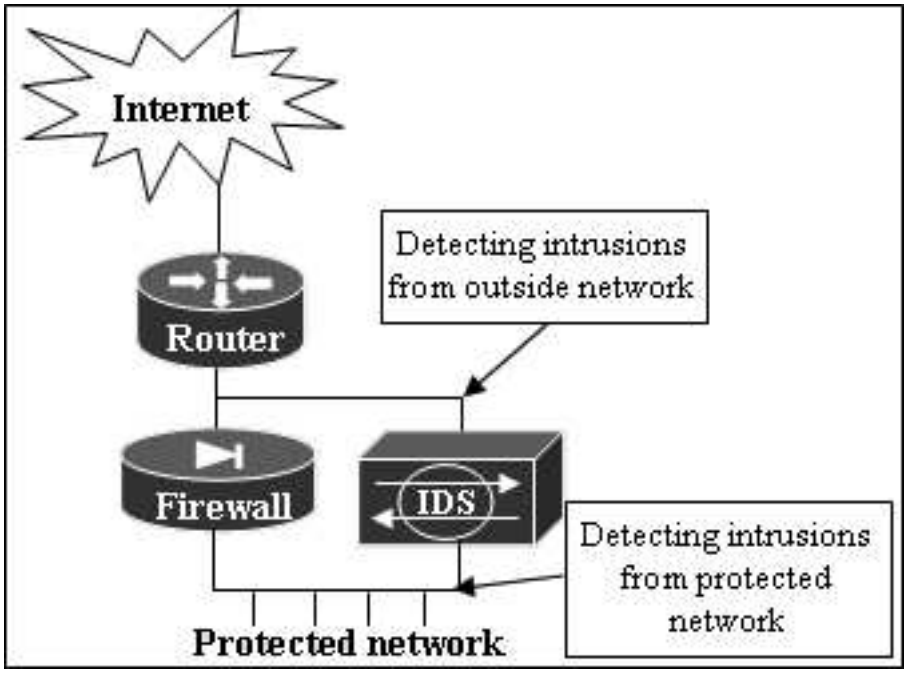

Figure 1: Typical firewall and IDS setup in a network

scanners (VS) in the remainder of this paper. Examples of VS tools are CyberCop Scanner [9], Cisco Secure Scanner [3] and Internet Security Scanner (ISS) [7]. Examples of reactive IDSs are Big Brother [16] and Snort [5].

A major problem with VS tools is that they sometimes attempt to address a too wide variety of vulnerabilities. The specific vulnerabilities that VS tools check for, however, differ significantly from tool to tool. Using only one specific VS tool may prove to be insufficient in scanning for certain types of vulnerabilities. For example, CyberCop Scanner [9] scans extensively for vulnerabilities of the type *misconfigurations*, whereas Cisco Secure Scanner [3] gives minimum attention to misconfiguration vulnerabilities. Furthermore, different VS tools sometimes refer differently to the same vulnerability. For example, CyberCop Scanner refers to *mail transfer* and Cisco Secure Scanner refers to *SMTP*, which is, in essence, the same set of vulnerabilities. How will the results of a vulnerability scan done by a specific tool, *e.g.* CyberCop Scanner, compare with the security results of another VS tool, *e.g.* Cisco Secure Scanner? To answer this question, a **common** set of vulnerabilities is required. The authors of this paper propose such a common set of vulnerabilities, which was determined by evaluating a number of different sets of vulnerabilities. This common set of vulnerabilities will be referred to as a "harmonised" set of vulnerability categories.

The harmonised vulnerability categories were identified by analysing the Internet security vulnerabilities as found in literature [10, 1, 12, 6, 11, 8], as well as those used by popular VS tools such as CyberCop Scanner and Cisco Secure Scanner. The criterion for identifying the harmonised vulnerability categories was based on the following [2]:

- Vulnerabilities of a **similar nature** should be grouped together.
- The classification should be **atomic**, in other words a specific vulnerability may not be classified in two different vulnerability classes.
- Classification should **not be based on the social cause** of the vulnerability. This includes issues like *motive*, *intent* and *malicious or accidental* cause.

The authors have identified 13 harmonised vulnerability categories. These harmonised vulnerability categories are shown in Table 1 and are discussed in the section that follows.

3 Harmonised Vulnerability Categories

A harmonised vulnerability category represents a certain group or class of vulnerabilities, which have the same genre of vulnerability characteristics. For example, all vulnerabilities related to compromising passwords, such as "a password is a dictionary word" or "a password is shorter than 8 characters" or "a password is sent in clear text", can form a harmonised vulnerability category called *password cracking and sniffing*. It is well known that VS tools in the industry represent solutions for rectifying vulnerabilities as well. It should be mentioned that the rectification of vulnerabilities is beyond the scope of this paper. In other words, the purpose of this paper is to identify harmonised vulnerability categories only, and not to present solutions on various vulnerabilities. Before discussing each harmonised vulnerability category in detail, a summary of the harmonised vulnerability categories is shown in Table 1.

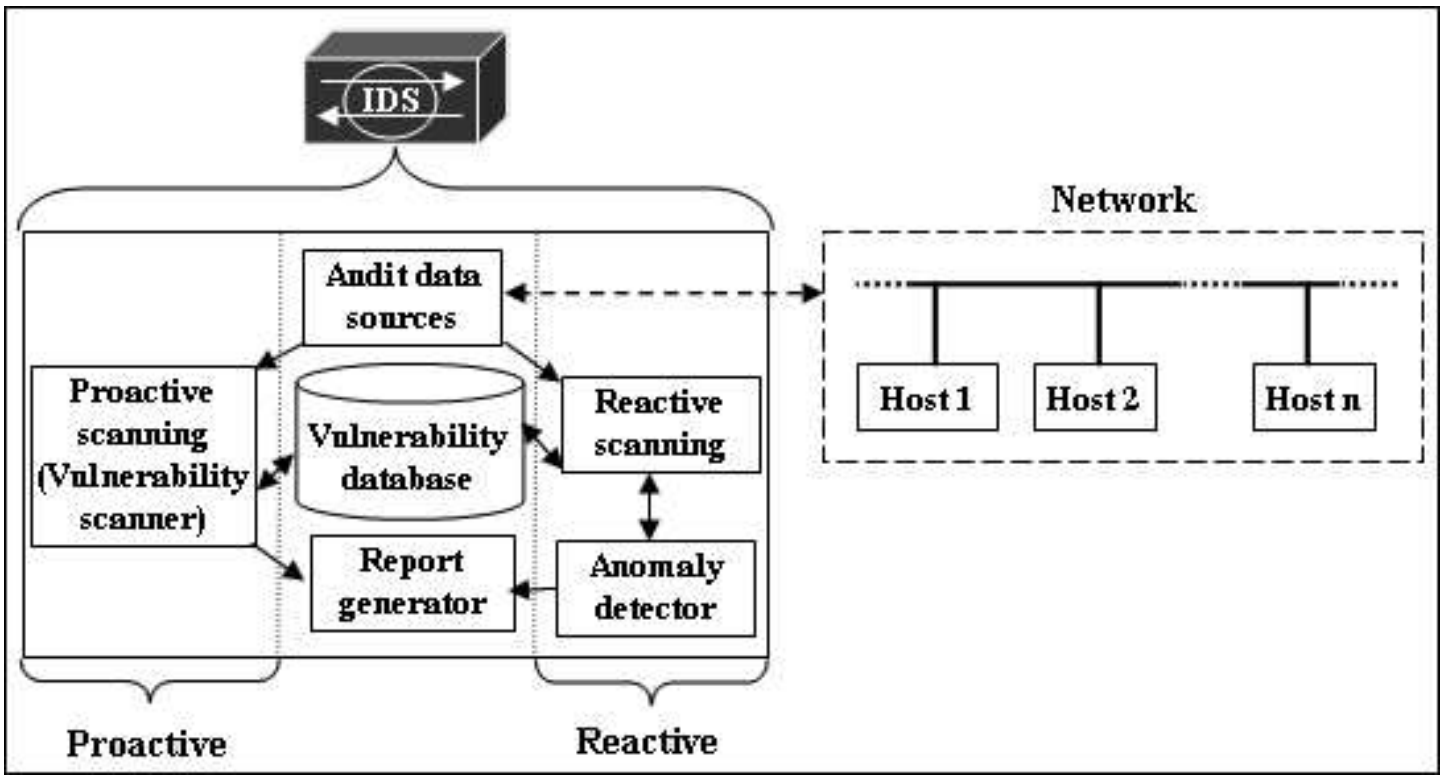

Figure 2: The architecture of current IDSs

3.1 Password cracking and sniffing

This category involves vulnerabilities with a root cause of having accounts with weak or no passwords. Tools are readily available on the Internet that can be used to intercept passwords from any transmission over the Internet. These kind of tools are better known as a *sniffers*.

On some systems, passwords are stored in clear text, or transmitted in clear text over the Internet. If an attacker manages to intercept clear text passwords, the passwords do not even need to be cracked. To solve this problem, passwords are transmitted or stored on a system in encrypted format. Still, it is possible to sniff these encrypted passwords from the Internet and then use password-cracking tools, for example L0pht Crack [17], to crack the passwords. Given that a user has administrative access, L0pht Crack can also retrieved the *stored* encrypted passwords on a system in an attempt to crack them.

Examples of vulnerabilities that belongs belonging to this category includeare the following:

- If the FTP service is enabled, anyone can try to guess passwords in a bid to connect to the FTP service.
- A malicious user could remotely retrieve the systems password file. This can lead to further system access, including administrator access.

3.2 Network and system information gathering

This category involves vulnerabilities concerned with scanning a network to discover a map of the available hosts, as well as to detect vulnerable services on the hosts and the network. Furthermore, it means to get information on the hosts found on the network in a bid to determine the specific hardware or software applications used.

Having a map of a network and information on which software applications are used in an organisation may help an intruder to gain sufficient information on the target and leads to a situation where the intruder is facilitated regarding specific hacking techniques to use. *Footprinting*, *network mapping*, *target acquisition*, and *network reconnaissance* are synonyms found in the literature [12, 11] for network and system information gathering.

Examples of vulnerabilities belonging to this category are the following:

- The routing table could be retrieved, which reveals information of the physical network set-up.
- Using the FTP SYST command, attackers can discover operating system version information. This can lead to administrator access and malicious activity.

3.3 User enumeration and information gathering

This category involves vulnerabilities concerned with retrieving information of user accounts from a specific system, for example, the user account name (*e.g. bretl*) and the user details (*e.g. Bret Lee, General Manager, Office 227, Accounts Department*).

An attacker can use this information typically to identify that Bret Lee is a general manager, whose computer, in

	Harmonised vulnerability category	Short description
1	Password cracking and sniffing	Vulnerabilities with a root cause of having accounts with weak or no passwords
2	Network and system information gathering	Vulnerabilities concerned with scanning a network to discover a map of available hosts and vulnerable services
3	User enumeration and information gathering	Vulnerabilities concerned with retrieving information of user accounts from a specific system
4	Backdoors, Trojans and remote controlling	Vulnerabilities concerned with having hidden access mechanisms installed on a system
5	Unauthorised access to remote connections & services	Vulnerabilities concerned with the risk that an unauthorizsed person has the ability to connect to and misuse a system
6	Privilege and user escalation	Vulnerabilities concerned with the risk that the access rights of an existing user account can be upgraded by an unauthorised user, granting more privileges to the user
7	Spoofing or masquerading	Vulnerabilities concerned with the risk that an intruder can fake an IP address in a bid to act as another person
8	Misconfigurations	Vulnerabilities concerned with the risk that applications have been incorrectly configured
9	Denial-of-services (DoS) and buffer overflows	Vulnerabilities concerned with the risk of one or more intruders launching an attack designed to disrupt or deny legitimate users' or applications' ability to access resources
10	Viruses and worms	Vulnerabilities concerned with malicious programs
11	Hardware specific	Vulnerabilities concerned with having hardware peripherals that execute ROM-based or firmware-based programs
12	Software specific and updates	Vulnerabilities concerned with the risk that specific software applications contain specific, well-known bugs
13	Security policy violations	Vulnerabilities concerned with the risk that an Internet security policy has been violated

Table 1: Summary of the harmonised vulnerability categories

turn, could contain more sensitive data information than a normal employee's computer, making the manager's computer a more sought-after target. Furthermore, as soon as an intruder has retrieved a list of the user account names registered on a specific system, it is often only a matter of time before he/she obtains the password by using a password-cracking program, for example, L0pht Crack [17]. After all, the user account names have to be obtained before any attempt can be made to crack passwords.

Examples of vulnerabilities belonging to this category are the following:

- Using the "finger" command on a specific system will retrieve a list of all the user account names on that system.
- Null session connections can be used by an attacker to list sensitive user account information, such as revealing the identity of a user on the system.

3.4 Backdoors, Trojans and remote controlling

This category involves vulnerabilities concerned with having access mechanisms installed on a system which are almost hidden and not obvious. In other words, when a covert channel is created.

Often a backdoor is installed with the goal to aim of controlling a system remotely. The backdoor becomes a hidden entry point where the intruder can connect to the system unnoticed at any given time. Most of the time, the "vehicle" for establishing such backdoors, is called a "Trojan horse" or a "Trojan" [12]. A Trojan is a software application that operates under the impression that it is intended for a specific purpose, but actually performs hidden operations as well. For example, most of the time Trojans are sent to someone as an e-mail attachment in the form of a game. As soon as the person opens that attachment, the game can be played successfully while a backdoor is unknowingly created in the background by the game.

Examples of vulnerabilities belonging to this category are the following:

- Back Orifice [4] or Netbus (recently called Spector) [15] are Trojan horse programs that, as soon as they are installed on your system, create backdoors, enabling remote controlling of the system.
- Remote controlling software is installed on the system, but it is not password-protected, allowing anyone to remotely connect and take over the system.

3.5 Unauthorised access to remote connections and services

This category involves vulnerabilities concerned with the risk that an unauthorised person has the ability to remotely connect to a system via a specific port with the aim of misusing the system.

Gaining access to remote connections and services is often used in an attempt to exploit more vulnerabilities, since gaining this will "open more doors" to other vulnerabilities. For example, if the TELNET service is running, anyone can attempt to connect to, for example, a guest account. Connecting to the TELNET service itself can do no harm. An attacker, however, can now gain information on the particular operating system that runs the TELNET service. This could lead to additional malicious activity by the attacker.

Examples of vulnerabilities belonging to this category are the following:

- An attacker could use an anonymous FTP server to launch exploits against another system to gain special access. An attacker could use this special access to possibly bypass firewalls.

- After anonymous access to the FTP server was has been gained, the attacker can try to exploit further vulnerabilities in the FTP service, for example, to see if the FTP root directory is write-enabled in a bid to store unauthorised data or information.

4 Privilege and user escalation

This category involves vulnerabilities concerned with the risk that the authorisation properties of an existing (probably compromised) system account can be changed so that this user account has more privileges or more powerful access rights allocated to it than was initially intended.

More privileges and more powerful access rights will allow a specific user account to access data or system resources in an effort to access specific data or information that was previously inaccessible to the user account. For example, an account with standard user rights might have been escalated to an account with administrative rights.

Examples of vulnerabilities belonging to this category are the following:

- An attacker could execute arbitrary commands remotely as the user who is running the HTTP server. If the owner of the HTTP server has administrative access, the attacker can remotely execute commands as an administrator.

- Some registry entries on a Windows system may be remotely accessible, allowing the modification of the permissions of these registry entries.

4.1 Spoofing or masquerading

This category involves vulnerabilities concerned with the risk that an intruder can fake an IP packet's source address to hide an intruder's identity or activity amongst a storm of other network traffic.

For example, assume *network A* is protected by a firewall that only allows IP addresses with source addresses in the subnet mask of 123.213.44.0. Assume an attacker is sitting in *network B* with a subnet mask of 211.143.2.0. The attacker could now create a packet in *network B*, which will have a source address of, for example, 211.143.2.67. By using the appropriate spoofing tool, the attacker can now easily change this source address to, for example, 123.312.44.67. The firewall in *network A* will now allow the packet created by the attacker through into *network A*.

Examples of vulnerabilities belonging to this category are the following:

- If a poorly configured firewall is installed, attackers can launch attacks using the identity of the firewall server, thus masking their true identity. If any hosts or networks allow special access to this server, then the attacker has the same access.

- IP forwarding was found to be enabled, allowing the host to act as a router so that other hosts can forward packets through this host. If this host is running a firewall, then the firewall can be bypassed using IP forwarding.

4.2 Misconfigurations

This category involves vulnerabilities concerned with the risk that applications have been incorrectly configured, leaving these applications vulnerable to several of the other harmonised vulnerability categories mentioned here.

Misconfiguration vulnerabilities mostly tend to occur after the installation of new software, because new software is always installed with *default* configuration settings. It is of the utmost importance that newly installed software is immediately reconfigured after installation. In addition, the new configurations must be tested to make sure that they are *correct* and not misconfigured.

Examples of vulnerabilities belonging to this category are the following:

- If anonymous FTP is not configured securely, an attacker may be able to perform reconnaissance, delete or modify files, or use anonymous FTP as a distribution mechanism for unwanted files, such as pornography or pirated software.

- If permissions are incorrectly set in the Windows registry to "Everyone", an attacker could gain access to the registry and commence with arbitrary attacks.

4.3 Denial-of-Service (DoS) and buffer overflows

This category involves vulnerabilities concerned with the risk of one or more intruders launching an attack designed to disrupt or completely deny legitimate users' access to networks, servers, services or other resources.

DoS vulnerabilities are not concerned with stealing information or changing data, but simply with downgrading the performance of the computer and/or network resources to such a level that services are disrupted significantly or completely. Consider an online shop that is completely reliant on the Internet to conduct business. Suppose an attacker manages to fill up the storage space of the online shop's servers by uploading junk data to it. This can potentially cause the servers to crash. It could take hours or perhaps days to sort out and restore the servers again, causing the online shop to lose so much money that it might have to close down.

Examples of vulnerabilities belonging to this category are the following:

- An attacker can create files on the hard disk of the web server and fill it up, leaving the service of the hard disk interrupted and unavailable.
- An out-of-band data attack can consume all memory and cause a system to reboot. This attack could also cause a system to be unable to handle network traffic. The only way to recover is to either reset or reboot the system.

4.4 Viruses and worms

Viruses and worms are different types of software applications, but with the same goal of spreading from one system to another to conduct malicious activity.

Viruses and worms can be considered as some of the most active and malicious vulnerabilities that can be found on a system. Unfortunately, this is the vulnerability category that is often completely neglected by IDSs. Almost any new virus that appears on the Internet scene these days causes havoc all over the world in a matter of hours. Why? Because they all spread through the Internet, be it through e-mail messages or through vulnerabilities exploited in networking services. For example, if an IDS could also detect for viruses and worms, the famous Code Red and Code Blue worms [HANC 01] would never have caused such havoc around the world in such a short time – it infected systems around the world in less than a day by spreading through an exploit in well-known web servers all over the world! It should be mentioned that it becomes evident that this problem is addressed in the newest *reactive* IDSs.

Examples of vulnerabilities belonging to this category are the following:

- An e-mail attachment is opened without having it scanned first by a virus detection program. This might allow a virus to infect the system.
- Certain updates or patches are not installed for the web server, making the server susceptible to a denial-of-service attack.

4.5 Hardware specific

This category involves vulnerabilities concerned with having hardware peripherals which do not run software applications, but which rather run ROM-based or firmware-based programs. These peripherals also contain exploits that cannot be easily updated, patched or corrected, except if the hardware is physically replaced or the firmware is updated.

Examples of such hardware peripherals are network switches, routers and terminals. The main reason why updating the firmware of these hardware peripherals is often neglected is that it does not have dedicated *owners* as opposed to a computer workstation which has one or more specific dedicated owners. Often the system administrator alone has to see to all of these peripherals in a network. Chances are better for an attacker to discover and exploit vulnerabilities on these peripherals before the administrator will discover that irregularities are happening on them.

Examples of vulnerabilities belonging to this category are the following:

- An attacker can cause a router or switch device to crash and reload. Possible loss of configuration information may result as a consequence of this attack.
- A shared printer was found on the network without having any authentication enabled on it, leaving it open to a variety of possible attacks. For example, some modern printers host a complete operating system on them. A network printer is often considered as highly trusted and trust relationships are set up accordingly as "wide open". If access to the operating system of such a printer is gained, an attacker can gain access to all those systems connected to the printer.

4.6 Software specific and updates

This category involves vulnerabilities concerned with the risk that specific software applications contain specific, well-known bugs. Because these bugs or exploits are published widely on the Internet [14], anyone, including an attacker, is able to access the Internet and collect information about these bugs to try and exploit them.

Software applications must be updated to *patch* their exploitations in an effort to fix security bugs or loopholes to avoid successful future attacks on them. For example, recently there have been enormous denial-of-service attacks on Microsoft's Internet Information Server by the very famous Code Red and Code Blue worms [13]. Therefore, Microsoft had to make *software patches* available to fix the vulnerabilities that were exploited so lustily by these Internet worms.

Examples of vulnerabilities belonging to this category are the following:

- A service pack installed is outdated. Vulnerabilities discovered after the specific service pack was installed on this system leave a potential threat unless they are patched by the latest service pack.
- An insecure logon method is allowed for a web server, causing a threat that a user name and password may be sniffed through this method.

4.7 Security policy violations

This category involves vulnerabilities concerned with the risk that an Internet security policy has been violated. An Internet security policy is a set of security rules created internally by an organisation. It can specify how systems in the organisation should be configured to be on a security level that is acceptable for the organisation. One of the policy statements might specify, for example, that the user's password will expire every 30 days.

When a security policy violation is found, it means that a different configuration setting on the system was detected and thus violates the prescribed policy setting. It is of the utmost importance, though, that management specifies the security policy *correctly before* it is implemented electronically. The policy might be implemented correctly according to the policy document, but if the document specification is wrong, its electronic implementation will also be wrong!

Examples of vulnerabilities belonging to this category are the following:

- The system's event or security log is not restricted according to the system's security policy. Anyone will thus be able to alter or delete the logs.
- The system's screensaver lockout is not enabled according to the system's security policy and will not automatically lock the system if the owner of the system neglected to lock the system himself/herself.

5 Illustrating The Application Of Harmonised Vulnerability Categories

How can the harmonised vulnerability categories be applied? The following example illustrates by means of a graph how the vulnerabilities in the vulnerability databases of CyberCop Scanner and Cisco Secure Scanner in general adhere to the harmonised vulnerability categories. This is done by mapping each VS tool's vulnerabilities from their respective vulnerability databases to the 13 harmonised vulnerability categories as shown, for example, in Figure 3.

Consider category 2, *Network and system information gathering*, in Figure 3. It shows that from the vulnerability database of CyberCop Scanner, about 125 of those vulnerabilities are classified as *Network and system information gathering* vulnerabilities. From the vulnerability database of Cisco Secure Scanner, about 70 vulnerabilities are classified as *Network and system information gathering* vulnerabilities.

Consider category 8, *Misconfigurations*, in Figure 3. Furthermore, assume that, for example, an organisation experiences a high number of misconfiguration vulnerabilities. This specific organisation has a requirement for a VS tool that would extensively point out such misconfigurations. Having the results of Figure 3, the organisation would opt for CyberCop Scanner rather than for Cisco Secure Scanner. Neither of these two tools, however, would be a good choice as far as *Virus and worm* detection, category 10, is concerned.

6 Conclusion

What is the significance of the 13 harmonised vulnerability categories? The significant aspect of the 13 harmonised vulnerability categories is that they aid in the evaluation process of VS tools when an organisation needs to decide which VS tool would suit the particular organisation best according to its needs.

The harmonised vulnerability categories can furthermore serve as a useful management tool. These harmonised vulnerability categories reflects all vulnerabilities in current state-of-the-art VSs today as well as those vulnerabilities found in current literature. The 13 harmonised categories will serve as generic categories for categorising vulnerabilities found in current state-of-the-art VS tools. The 13 harmonised categories will expand and evolve along with the evolution of information technology and its applications.

Be that as it may, such a construction of harmonised vulnerability categories will contribute significantly to safer and better managed Internet information security.

References

[1] R.G. Bace. Password-Cracking. *Intrusion Detection, ISBN 1-57870-185-6*, pages 3, 31, 136, 150–151, 179, 279–280, 2000.

[2] M. Bishop. Vulnerabilities analysis. In *Proceedings of the Recent Advances in Intrusion*, pages 125–136, 1999.

[3] Cisco Systems Inc. Cisco Secure Scanner, Version 2.0.1.2. `http://www.cisco.com`, 2000.

[4] Cult Of A Dead Cow. Back Orifice. `http://www.cultdeadcow.com`, 2002.

[5] Data Nerds. Snort. `http://www.datanerds.net/~mike/snort.html`, 2001.

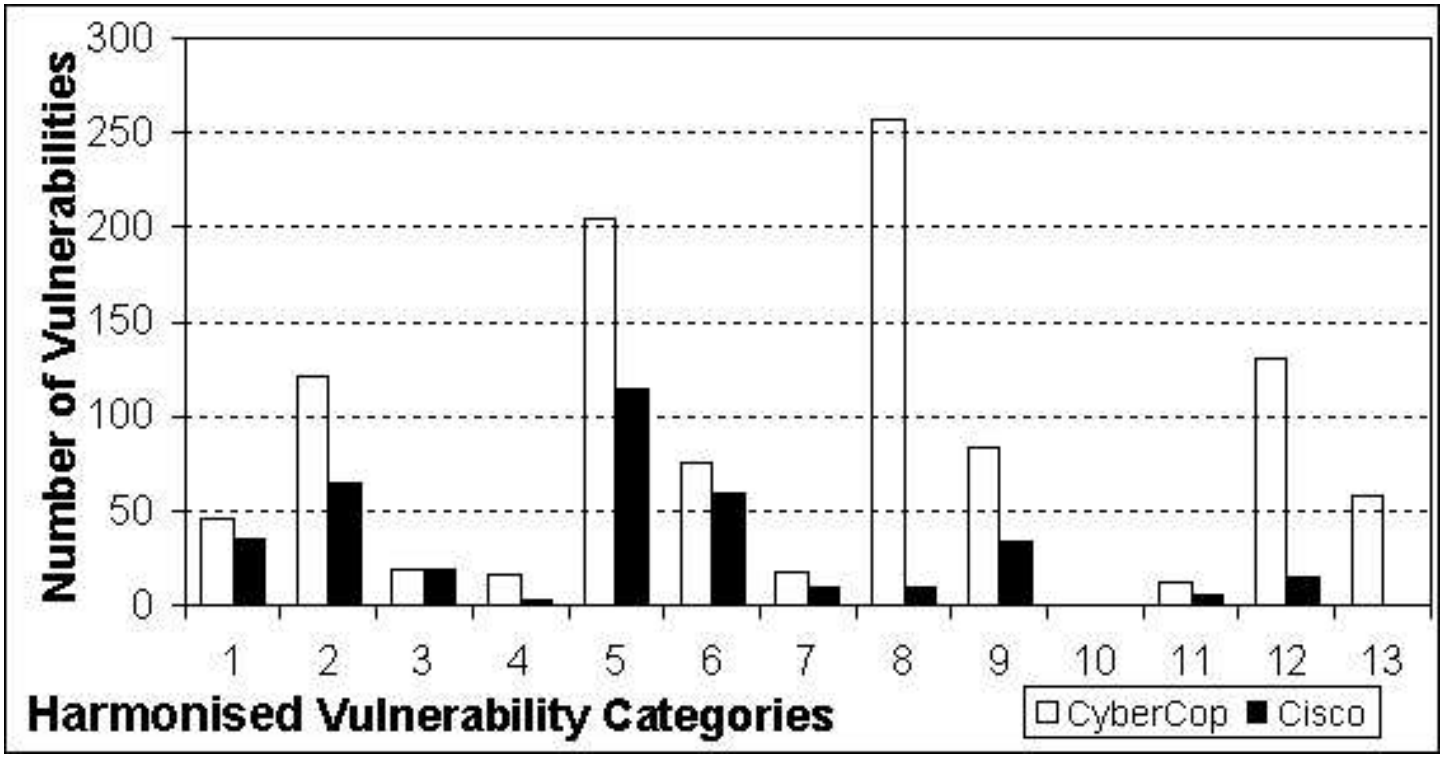

Figure 3: How the vulnerability databases of CyberCop Scanner and Cisco Secure Scanner adhere to the 13 harmonised vulnerability categories

[6] M. Greenstein and M. Vasarhelyi. Risks Associated with Viruses and Malicious Code Overflows, ISBN 0-07-241081-7. *Electronic Commerce – Security, Risk Management, and Control*, pages 242–245, 2002.

[7] Internet Security Systems. ISS Resource Center, Introduction to RealSecure Version 5.0. `http://documents.iss.net/literature/RealSecure/rs_guide.pdf`, 2002.

[8] C.M. King, C.E. Dalton, and T.E. Osmanoglu. Security Policies, Standards, and Guidelines. *Security Architecture - Design, Deployment & Operations, ISBN 0-07-213385-6*, pages 13–39, 2001.

[9] Network Associates. PGP Securities, CyberCop Monitor. `http://www.pgp.com/products/cybercop-monitor/default.asp`, 2002.

[10] S. Northcutt, M. Cooper, M. Fearnow, and K. Frederick. Passwords. *Intrusion Signatures and Analysis, ISBN 0-7357-1063-5*, pages 57–65, 76–85, 134–143, 149–168, 189, 233–250, 2001.

[11] S. Northcutt, J. Novak, and D. McLachlan. Misconfigured Systems. *Network Intrusion Detection, ISBN 0-7357-1008-2*, pages 159, 188, 213–214, 387–391, 2001.

[12] J. Scambray, S. McClure, and G. Kurtz. Footprinting. *Hacking Exposed, ISBN 0-07-212748-1*, pages 5–34, 87–95, 164–174, 238–241, 252–257, 287–290, 308, 339–340, 433–437, 453–456, 483–506, 507–653, 2001.

[13] Security Focus. Advisories, IIS Worms Detector. `http://www.securityfocus.com`, 2002.

[14] Securityfocus.Com. Bugtraq, Bugtraq Archives. `http://www.securityfocus.com/forums/bugtraq/intro.html`, 2002.

[15] Spectrosoft. Netbus. `http://www.netbus.org`, 2002.

[16] BB4 Technologies Inc. Big Brother: The big brother system and network monitor. `http://bb4.com`, 2002.

[17] @Steak.Com. L0pht Crack, L0pht Crack Version 3.0. `http://www.atstake.com/research/lc3/index.html`, 2002.