|  |
| --- |
| Lomonosov Moscow State UniversityFaculty of Computational Mathematics and Cybernetics |
| Computer Security |
|  |
| **Vasilenko Anatoly**  |
| **211 group** |
| **Moscow, 2013** |

**Contents**

1. Introduction ……………...………………………………………………………………………………………………………3
2. Definitions ………………………………………………………………….………..……………………………………………3
3. Computer attack …..……………………………………………………………..……………………………………………4
	1. Different divisions of computer attacks ………………..………………………………………………4
	2. Types of attacks ..…………………………………………………..………………………………………………5
	3. Security exploits ………………..…………………………………..………………………………………………7
4. Vulnerability ………………………………………………………………………………………………………………………7
	1. Classification .…………………………………………………………………………………………………………7
	2. Causes ……………………………………………………………………………………………………………………7
	3. Vulnerability disclosure ………………………………………………………….………………………………8
	4. Vulnerability disclosure date …………………………………………………………….……………………9
	5. Identifying and removing vulnerabilities ………………..………………………..…………..………9
5. Malware ……………………………………………………………………………………………………..……………….……9
	1. Antimalware strategies ………………………………………………………………….………………..……9
	2. Grayware ……………………………………………….....………………………………….……………………10
6. Computer security approaches ……………………………………………………………….………………..……10
	1. Security by design ……………………………………………………………………….………………………10
	2. Security architecture ………………………………………………………………….………………….……10
	3. Secure operating system ………………………………………………………….…………………………10
	4. Secure coding …………………………………………………………………….….……………………………11
	5. Capabilities and access control list ……………………………………..…………….……….………12
	6. Hardware mechanisms that protects computers and ………………………..………………12
7. Network security …………………..……………………………………………………………………….………………12
	1. Network security concepts ….………..….…………..……………………………….…………………12
	2. Security management …………………….…………….……………………………………………………13
8. Hackers classification ………………………………….………….……………….……………………………………15
	1. White hat ………………………………………..………….…………………………..…………………………15
	2. Black hat …………………………………………………….………………………………………………………15
	3. Grey hat ……………………………………………………….……………………………………………….……15
	4. Elite hacker …………………………………………………..……………………….……………………………16
	5. Script kiddie ………………………………………………….………….…………………………………………16
	6. Neophyte …………………………………………………….…………….……………………………….………16
	7. Blue hat ……………………………………………………..……………….………………………………………16
	8. Hacktivist ……………………………………………………..…………..…..……………………………………16
	9. Bots ………………………………………………………………………………..………………………………..…16
9. Conclusion ………………………………………………………………………………………………………………………16
10. References ……………………………………………………………………………….…………………………………….18

**Introduction**

Almost everybody has got spam at least once. It can be called a successful attack, because the target (you) has got an advertisement. In our world there are a lot of different attacks which happen every second, but not all of them achieve their goals. This happens because many companies try to fight these attacks.

In general a computer attack or a computer network attack is any attempt to destroy, reveal or do something else illegal with information. Usually professionals distinguish three aspects which need protection. It is privacy, integrity and accessibility of information resources.

Creating a secure system is a very hard and expensive task, and to achieve it developers should follow special rules. Creating a secure system is also a very specific problem, in comparison with others tasks that are solved in real world. To solve this problem a developer must understand how the criminals attack modern systems. They should even be half criminals to understand their trade. System designers must understand all stages of creating criminal soft (malware) and making an attack to create defense on every step the criminals tries to perform.

This is why considering computer security definitely requires discussion not only computer security approaches, but at first we must consider themes like computer attacks by itself, vulnerabilities and malware.

**Definitions**

**Computer security** is information security as applied to computers and networks. The field covers all the processes and mechanisms by which computer-based equipment, information or data and services are protected from unintended or unauthorized access, change or destruction. To understand computer security we should understand all aspects of its essence.

Computer security has big branch called network security. **Network security** consists of the provisions and policies adopted by a network administrator to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources. Network security involves the authorization of access to data in a network, which is controlled by the network administrator. Network security covers both public and private computer networks.

Internet engineering task force (IETF) defines **attack** as an assault on system security that derives from an intelligent threat, i.e., an intelligent act that is a deliberate attempt (especially in the sense of a method or technique) to evade security services and violate the security policy of a system.

The term attack relates to some other basic security terms. A resource (both physical or logical), called an asset, can have one or more vulnerabilities that can be exploited by a threat agent in a threat action. The result can potentially compromises the Confidentiality, Integrity or Availability properties of resources (potentially different that the vulnerable one) of the organization.

IETF RFC defines **vulnerability** as a flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's security policy.

But vulnerability does not constitute a menace by itself. There should be someone to produce attacks. In the computer security context, a **hacker** is someone who seeks and exploits vulnerabilities in a computer system or computer network. Hackers may be motivated by a multitude of reasons, such as profit, protest, or challenge. The subculture that has evolved around hackers is often referred to as the computer underground and is now a known community.

All hackers usually use specific software to produce their attacks. It is called malware. **Malware**, short for malicious software, is software used or programmed by attackers to disrupt computer operation, gather sensitive information, or gain access to private computer systems. It can appear in the form of code, scripts, active content, and other software. Malware is a general term used to refer to a variety of forms of hostile or intrusive software.

**Computer Attack**

In computer and computer networks an attack is any attempt to destroy, expose, alter, disable, steal or gain unauthorized access to or make unauthorized use of an asset.

A threat is a potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm. That is, a threat is a possible danger that might exploit a vulnerability. A threat can be intentional or accidental, but first happens more often.

An organization should make steps to detect, classify and manage security incidents. In order to detect attacks, a number of countermeasures can be set up at organizational, procedural and technical levels. Computer emergency response team, information technology security audit and intrusion detection system are example of these.

**Different divisions of computer attacks**

Today many different classifications of computer attacks exist. Here I will describe some of them in general.

First type of division of attack classifies them by whom they are accomplished. An attack can be perpetrated by an insider or from outside the organization. An “inside attack” is an attack initiated by an entity inside the security perimeter (an “insider”), i.e., an entity that is authorized to access system resources but uses them in a way not approved by those who granted the authorization. An “outside attack” is initiated from outside the perimeter, by an unauthorized or illegitimate user of the system (an "outsider"). In the Internet, potential outside attackers range from amateur pranksters to organized criminals, international terrorists, and hostile governments.

Another division of attacks classifies them by their influence on system. The attack is “active” when it attempts to alter system resources or affect their operation, so it compromises integrity or availability. A “passive attack” attempts to learn or make use of information from the system but does not affect system resources, so it compromises confidentiality.

**Types of attacks**

The attacks can also be classified according to their origin, i.e. if it is conducted using one or more computers (in the last case it is called a distributed attack). Botnet are used to conduct distributed attacks.

Other classifications are according to the procedures used or the type of vulnerabilities exploited: attacks can be concentrated on network mechanisms or host features.

Some attacks are physical, i.e. theft or damage of computers and other equipment. Others are attempts to force changes in the logic used by computers or network protocols in order to achieve unforeseen (by the original designer) result but useful for the attacker. Software used to for logical attacks on computers is called malware.

A typical approach in an attack on Internet-connected system consists of three steps. Firstly, network enumeration, discovering information about the intended target. Secondly, vulnerability analysis: identifying potential ways of attack, and finally, exploitation: attempting to compromise the system by employing the vulnerabilities found through the vulnerability analysis.

In order to do so, there are several recurring tools of the trade and techniques used by computer criminals and security experts.

The following is a partial short list of attacks:

1. Passive
	1. Network
		1. wiretapping
		2. Port scanner
		3. Idle scan
2. Active
	1. Denial-of-service attack
	2. Spoofing
	3. Network
		1. Man in the middle
		2. ARP poisoning
		3. Ping flood
		4. Ping of death
		5. Smurf attack
	4. Host
		1. Buffer overflow
		2. Heap overflow
		3. Format string attack
		4. SQL injection

Now I would like to explain some types of attacks

A vulnerability scanner is a tool used to quickly check computers on a network for known weaknesses. Hackers also commonly use port scanners. These check to see which ports on a specified computer are “open” or available to access the computer, and sometimes will detect what program or service is listening on that port, and its version number.

Password cracking is the process of recovering passwords from data that has been stored in or transmitted by a computer system. A common approach is to repeatedly try guesses for the password.

A packet sniffer is an application that captures data packets, which can be used to capture passwords and other data in transit over the network.

A spoofing attack (fishing) involves one program, system, or website successfully masquerading as another by falsifying data and thereby being treated as a trusted system by a user or another program. The purpose of this is usually to fool programs, systems, or users into revealing confidential information, such as user names and passwords, to the attacker.

A rootkit is designed to conceal the compromise of a computer's security, and can represent any of a set of programs which work to subvert control of an operating system from its legitimate operators. Usually, a rootkit will obscure its installation and attempt to prevent its removal through a subversion of standard system security. Rootkits may include replacements for system binaries so that it becomes impossible for the legitimate user to detect the presence of the intruder on the system by looking at process tables.

Next we come to social engineering. When a hacker, typically a black hat, is in the second stage of the targeting process, he or she will typically use some social engineering tactics to get enough information to access the network. A common practice for hackers, who use this technique, is to contact the system administrator and play the role of a user who cannot get access to his or her system. Hackers who use this technique have to be quite savvy and choose the words they use carefully, in order to trick the system administrator into giving them information. In some cases only an employed help desk user will answer the phone and they are generally easy to trick. Another typical hacker approach is for the hacker to act like a very angry supervisor and when his authority is questioned they will threaten the help desk user with their job. Social engineering is very effective because users are the most vulnerable part of an organization. All the security devices and programs in the world won't keep an organization safe if an employee gives away a password. Black hat hackers take advantage of this fact.

A Trojan horse is a program which seems to be doing one thing, but is actually doing another. A trojan horse can be used to set up a back door in a computer system such that the intruder can gain access later.

A virus is a self-replicating program that spreads by inserting copies of itself into other executable code or documents. Therefore, a computer virus behaves in a way similar to a biological virus, which spreads by inserting itself into living cells. While some are harmless or mere hoaxes, most computer viruses are considered malicious.

Like a virus, a worm is also a self-replicating program. A worm differs from a virus in that it propagates through computer networks without user intervention. Unlike a virus, it does not need to attach itself to an existing program. Many people conflate the terms “virus” and “worm”, using them both to describe any self-propagating program.

A key logger is a tool designed to record (log) every keystroke on an affected machine for later retrieval. Its purpose is usually to allow the user of this tool to gain access to confidential information typed on the affected machine, such as a user's password or other private data. Some key loggers use methods based on virus, trojans or rootkits to remain active and hidden. However, some key loggers are used in legitimate ways and sometimes to even enhance computer security. As an example, a business might have a key logger on a computer used at a point of sale and data collected by the key logger could be used for catching employee fraud.

**Security exploits**

A security exploit is a prepared application that takes advantage of a known weakness. Common examples of security exploits are SQL injection, Cross Site Scripting and Cross Site Request Forgery which abuse security holes that may result from substandard programming practice. Other exploits would be able to be used through FTP, HTTP, PHP, SSH, Telnet and some web-pages. These are very common in website or domain hacking.

**Vulnerability**

In computer security, a vulnerability is a weakness which allows an attacker to reduce a system's information assurance. All attacks uses some vulnerabilities in system.

Vulnerability is the intersection of three elements: a system susceptibility or flaw, attacker access to the flaw, and attacker capability to exploit the flaw. To exploit a vulnerability, an attacker must have at least one applicable tool or technique that can connect to a system weakness. In this frame, vulnerability is also known as the attack surface.

Security bug (security defect) is a vulnerability that is not related to software. Hardware, site, personnel vulnerabilities are examples of vulnerabilities that are not software security bugs.

A resource may have one or more vulnerabilities that can be exploited by a threat agent in a threat action. The result can potentially compromise the confidentiality, integrity or availability of resources belonging to an organization. OWASP (The Open Web Application Security Project) depicts the same phenomenon in a different words: a threat agent through an attack vector exploits a weakness (vulnerability) of the system and the related security controls causing a technical impact on an IT resource (asset) connected to a business impact.

**Classification**

Vulnerabilities are classified according to the asset class they are related to. There is six different points: hardware, software, network, personnel, site and organizational.

**Causes**

Exists many causes for vulnerabilities. Here is some of them:

Firstly, complexity: large, complex systems increase the probability of flaws and unintended access points.

Secondly, familiarity: using common, well-known code, software, operating systems, or hardware increases the probability an attacker has or can find the knowledge and tools to exploit the flaw.

Thirdly, connectivity: more physical connections, privileges, ports, protocols, and services and time each of those are accessible increase vulnerability.

In the fourth place is a password management flaw: the computer user uses weak passwords that could be discovered by brute force, they stores the password on the computer where a program can access it and they re-use passwords between many programs and websites.

In the fifth place is a fundamental operating system design flaw: the operating system designer chooses to enforce suboptimal policies on user management. For example operating systems with policies such as default permit grant every program and every user full access to the entire computer. This operating system flaw allows viruses and malware to execute commands on behalf of the administrator.

In the sixth place is an internet website browsing: some internet websites may contain harmful spyware or adware that can be installed automatically on the computer systems. After visiting those websites, the computer systems become infected and personal information will be collected and passed on to third party individuals.

In the seventh place is a software bugs: the programmer leaves an exploitable bug in a software program and as the result the software bug may allow an attacker to misuse an application.

Finally, another cause is an unchecked user input: the program assumes that all user input is safe. Programs that do not check user input can allow unintended direct execution of commands or SQL statements (known as Buffer overflows, SQL injection or other non-validated inputs).

Not learning from past mistakes also causes troubles: for example most vulnerabilities discovered in IPv4 protocol software were discovered in the new IPv6 implementations. But this cause is related to absolutely everything in our world.

The research has shown that the most vulnerable point in most information systems is the human user, operator, designer, or other human: so humans should be considered in their different roles as asset, threat, information resources. Social engineering is an increasing security concern.

**Vulnerability disclosure**

A responsible disclosure first alerts the affected vendors confidentially before alerting CERT (Computer Emergency Response Team) two weeks later, which grants the vendors another 45 day grace period before publishing a security advisory.

Full disclosure is done when all the details of vulnerability is publicized, perhaps with the intent to put pressure on the software or procedure authors to find a fix urgently.

Security researchers catering to the needs of the cyberwarfare or cybercrime industry have stated that this approach does not provide them with adequate income for their efforts. Instead, they offer their exploits privately to enable “Zero day attacks” (nobody can resist them because they are unknown).

The never ending effort to find new vulnerabilities and to fix them is called Computer insecurity.

**Vulnerability disclosure date**

The time of disclosure of a vulnerability is defined differently in the security community and industry. It is most commonly referred to as a kind of public disclosure of security information by a certain party. Usually, vulnerability information is discussed on a mailing list or published on a security web site and results in a security advisory afterward.

The time of disclosure is the first date a security vulnerability is described on a channel where the disclosed information on the vulnerability has to fulfill the following three requirements: firstly, the information is freely available to the public, secondly the vulnerability information is published by a trusted and independent source, and finally, the vulnerability has undergone analysis by experts such that risk rating information is included upon disclosure.

**Identifying and removing vulnerabilities**

Many software tools exist that can aid in the discovery (and sometimes removal) of vulnerabilities in a computer system. Though these tools can provide an auditor with a good overview of possible vulnerabilities present, they cannot replace human judgment. Relying solely on scanners will yield false positives and a limited-scope view of the problems present in the system.

Vulnerabilities have been found in every major operating system including Windows, Mac OS, various forms of UNIX and Linux, and others. The only way to reduce the chance of a vulnerability being used against a system is through constant vigilance, including careful system maintenance (e.g. applying software patches), best practices in deployment (e.g. the use of firewalls and access controls) and auditing (both during development and throughout the deployment lifecycle).

**Malware**

Malware includes computer viruses, worms, trojan horses, rootkits, keyloggers, dialers, spyware, adware, rogue security software and other malicious programs; the majority of active malware threats are usually worms or trojans rather than viruses. In law, malware is sometimes known as a computer contaminant, as in the legal codes of several U.S. states. Malware is different from defective software, which is a legitimate software but contains harmful bugs that were not corrected before release. However, some malware is disguised as genuine software, and may come from an official company website in the form of a useful or attractive program which has the harmful malware embedded in it along with additional tracking software that gathers marketing statistics.

Software such as anti-virus, anti-malware, and firewalls are relied upon by users at home, small and large organizations to safeguard against malware attacks which helps in identifying and preventing the further spread of malware in the network.

**Antimalware strategies**

As malware attacks become more frequent, attention has begun to shift from viruses and spyware protection, to malware protection, and programs that have been specifically developed to combat malware.

**Grayware**

Grayware (or greyware) is a general term that refers to applications or files that are not directly classified as malware (like worms or trojan horses), but can still negatively affect the performance of computers and involve significant security risks.

It describes applications that behave in an annoying or undesirable manner, and yet are less serious or troublesome than malware. Grayware encompasses spyware, adware, dialers, joke programs, remote access tools and any other program apart from a virus, which is designed to harm the performance of computers.

**Computer security approaches**

Exists many approaches to create system secure, and some of them can be used separately from others, but to achieve the best result, developers should use all of them. These approaches are: security by design, security architecture, security operating systems, secure coding, capabilities and access control list, hardware mechanisms that protect computers and data.

**Security by design**

The main approach to think of computer security is to reflect security as one of the main features. This approach is called security by design. Some of the techniques in this approach include seven different rules. Firstly, the principle of least privilege, where each part of the system has only the privileges that are needed for its function. That way even if an attacker gains access to that part, they have only limited access to the whole system. Second principle is automated theorem proving to prove the correctness of crucial software subsystems. Thirdly, code reviews and unit testing are approaches to make modules more secure where formal correctness proofs are not possible. Fourthly defense in depth, where the design is such that more than one subsystem needs to be violated to compromise the integrity of the system and the information it holds. In the fifth place is safety engineering, default secure settings, and design to "fail secure" rather than "fail insecure". Ideally, a secure system should require a deliberate, conscious, knowledgeable and free decision on the part of legitimate authorities in order to make it insecure. In the six place is audit trails, which tracks system activity, so that when a security breach occurs, the mechanism and extent of the breach can be determined. Storing audit trails remotely, where they can only be appended to, can keep intruders from covering their tracks. And finally full disclosure to ensure that when bugs are found the time of not fixed vulnerability is kept as short as possible.

**Security architecture**

The next approach in providing security is to think of security architecture. It can be defined as the design artifacts that describe how the security controls (security countermeasures) are positioned, and how they relate to the overall information technology architecture. These controls serve to maintain the system's quality attributes: confidentiality, integrity, availability, accountability and assurance services.

**Secure operating system**

One use of the term computer security refers to technology to implement a secure operating system. The technology is in limited use today, primarily because it imposes some changes to system management and also because it is not widely understood. Such ultra-strong secure operating systems are based on operating system kernel technology that can guarantee that certain security policies are absolutely enforced in an operating environment. An example of such a computer security policy is the Bell-LaPadula model. The strategy is based on a coupling of special microprocessor hardware features, often involving the memory management unit, to a special correctly implemented operating system kernel. This forms the foundation for a secure operating system which, if certain critical parts are designed and implemented correctly, can ensure the absolute impossibility of penetration by hostile elements. This capability is enabled because the configuration not only imposes a security policy, but in theory completely protects itself from corruption. Ordinary operating systems, on the other hand, lack the features that assure this maximal level of security. The design methodology to produce such secure systems is precise, deterministic and logical.

In USA parlance, the term High Assurance usually suggests the system has the right security functions that are implemented robustly enough to protect very important information. Medium assurance suggests it can protect less valuable information, such as income tax information. Secure operating systems designed to meet medium robustness levels of security functionality and assurance have seen wider use within both government and commercial markets. Medium robust systems may provide the same security functions as high assurance secure operating systems but do so at a lower assurance level. Lower levels mean we can be less certain that the security functions are implemented flawlessly, and therefore less dependable. These systems are found in use on web servers, guards, database servers, and management hosts and are used not only to protect the data stored on these systems but also to provide a high level of protection for network connections and routing services.

**Secure coding**

If the operating environment is not based on a secure operating system capable of maintaining a domain for its own execution, and capable of protecting application code from malicious subversion, and capable of protecting the system from subverted code, then high degrees of security are understandably not possible. While such secure operating systems are possible and have been implemented, most commercial systems fall in a “low security” category because they rely on features not supported by secure operating systems (like portability, and others). In low security operating environments, applications must be relied on to participate in their own protection. There are “best effort” secure coding practices that can be followed to make an application more resistant to malicious subversion.

In commercial environments, the majority of software subversion vulnerabilities result from a few known kinds of coding defects. Common software defects include buffer overflows, format string vulnerabilities, integer overflow, code or command injection and dangling pointers, which until recently was known but considered to be academic and not practically exploitable. These defects can be used to cause the target system to execute putative data. However, the data contain executable instructions, allowing the attacker to gain control of the processor.

Some common languages such as C and C++ are vulnerable to all of these. Other languages, such as Java, are more resistant to some of these defects, but are still prone to code or command injection and other software defects which facilitate subversion.

Unfortunately, there is no theoretical model of secure coding practices, nor is one practically achievable, insofar as the code (ideally, read-only) and data (generally read/write) generally tends to have some form of defect.

**Capabilities and access control list**

Within computer systems, two security models capable of enforcing privilege separation are access control lists (ACLs) and capability-based security. The semantics of ACLs have been proven to be insecure in many situations, for example, the confused deputy problem. It has also been shown that the promise of ACLs of giving access to an object to only one person can never be guaranteed in practice. Both of these problems are resolved by capabilities. This does not mean practical flaws exist in all ACL-based systems, but only that the designers of certain utilities must take responsibility to ensure that they do not introduce flaws.

Capabilities have been mostly restricted to research operating systems and commercial OSs still use ACLs. Capabilities can, however, also be implemented at the language level, leading to a style of programming that is essentially a refinement of standard object-oriented design. An open source project in the area is the E language.

The most secure computers are those not connected to the Internet and shielded from any interference. In the real world, the most secure systems are operating systems where security is not an add-on.

**Hardware mechanisms that protects computers and data**

Hardware based or assisted computer security offers an alternative to software-only computer security. Devices such as dongles, case intrusion detection, drive locks, or disabling USB ports, or CD ROM Drives may be considered more secure due to the physical access required in order to be compromised.

**Network security**

Network security is one of the orientations in computer security.

Network security covers a variety of computer networks, both public and private, that are used in everyday jobs conducting transactions and communications among businesses, government agencies and individuals. Networks can be private, such as within a company, and others which might be open to public access. Network security is involved in organizations, enterprises, and other types of institutions. It does as its title explains. It secures the network, as well as protecting and overseeing operations being done.

**Network security concepts**

Network security starts with authenticating, commonly with a username and a password. Since this requires just one detail authenticating the user name — i.e. the password — this is sometimes termed one-factor authentication. With two-factor authentication, something the user “has” is also used (e.g. a security token, an ATM card, or a mobile phone), and with three-factor authentication, something the user “is” is also used (e.g. a fingerprint or retinal scan).

Once authenticated, a firewall enforces access policies such as what services are allowed to be accessed by the network users. Though effective to prevent unauthorized access, this component may fail to check potentially harmful content such as computer worms or trojans being transmitted over the network. Anti-virus software or an intrusion prevention system (IPS) helps detect and inhibit the action of such malware. An anomaly-based intrusion detection system may also monitor the network and traffic for unexpected (or suspicious) content or behavior and other anomalies to protect resources, e.g. from denial of service attacks or an employee accessing files at strange times. Individual events occurring on the network may be logged for audit purposes and for later high-level analysis.

Communication between two hosts using a network may be encrypted to maintain privacy.

[Honeypots](http://en.wikipedia.org/wiki/Honeypot_%28computing%29), essentially decoy network-accessible resources, may be deployed in a network as surveillance and early-warning tools, as the honeypots are not normally accessed for legitimate purposes. Techniques used by the attackers that attempt to compromise these decoy resources are studied during and after an attack to keep an eye on new exploitation techniques. Such analysis may be used to further tighten security of the actual network being protected by the honeypot.

**Security management**

Security management for networks is different for all kinds of situations. A home or small office may only require basic security while large businesses may require high-maintenance and advanced software and hardware to prevent malicious attacks from hacking and spamming.

Homes and small businesses security requirements:

1. A basic firewall or a unified threat management system.
2. For Windows users, basic Antivirus software. An anti-spyware program would also be a good idea. There are many other types of antivirus or anti-spyware programs available.
3. When using a wireless connection, use a robust password. Also try to use the strongest security supported by your wireless devices, such as WPA2 with AES. TKIP may be more widely supported by your devices and should only be considered in cases where they are not compliant with AES.
4. If you are using wireless network, change the default SSID network name, also disable SSID Broadcast, because this function is unnecessary for home use. Security experts consider this to be easily bypassed with modern technology and some knowledge of how wireless traffic is detected by software.
5. Enable MAC Address filtering to keep track of all home network MAC devices connecting to your router. This is not a security feature per se, however it can be used to limit and strictly monitor your DHCP address pool for unwanted intruders.
6. Assign “static” IP addresses to network devices. This measure may be used, in conjunction with other features, to make your AP less desirable to would-be intruders.
7. Disable ICMP ping on router.
8. Review router or firewall logs to help identify abnormal network connections or traffic through your router.
9. Use passwords for all accounts.
10. If you use windows operating system, have multiple accounts per family member and use non-administrative accounts for day-to-day activities.
11. Raise awareness about information security to children.

Medium businesses security requirements:

1. A fairly strong firewall or Unified Threat Management System.
2. Strong antivirus software and internet security software.
3. Use strong passwords for authentication and change them on a weekly or monthly basis.
4. Using a wireless connection, use a robust password.
5. Raise awareness about physical security to employees.
6. Use an optional network analyzer or network monitor.
7. An enlightened administrator or manager.
8. Use a VPN (Virtual Private Network), to communicate between a main office and satellite offices using the Internet as a connectivity medium. A VPN offers a solution to the expense of leasing a data line while providing a secure network for the offices to communicate. Although the Internet is used, it is private because the link is encrypted and convenient to use. A medium sized business needing a secure way to connect several offices will find this a good choice.
9. Clear employee guidelines should be implemented for using the Internet, including access to non-work related websites, sending and receiving information.
10. Individual accounts to log on and access company intranet and Internet with monitoring for accountability.
11. Have a back-up policy to recover data in the event of a hardware failure or a security breach that changes, damages or deletes data.
12. Disabling messenger also helps to improve business security.
13. Assign several employees to monitor a group like CERT which studies Internet security vulnerabilities and develops training to help improve security.

Large businesses security requirements:

1. A strong firewall and proxy, or network Guard, to keep unwanted people out.
2. A strong antivirus software package and internet security software package.
3. For authentication, use strong passwords and change them on a weekly basis.
4. Using a wireless connection, use a robust password.
5. Exercise physical security precautions to employees.
6. Prepare a network analyzer or network monitor and use it when needed.
7. Implement physical security management like closed circuit television for entry areas and restricted zones.
8. Security fencing to mark the company's perimeter.
9. Fire extinguishers for fire-sensitive areas like server rooms and security rooms.
10. Security guards can help to maximize physical security.

School security requirements:

1. An adjustable firewall and proxy to allow authorized users access from the outside and inside.
2. Strong Antivirus software and Internet Security Software packages.
3. Wireless connections that lead to firewalls.
4. Supervision of network to guarantee updates and changes based on popular site usage.
5. Constant supervision by teachers, librarians, and administrators to guarantee protection against attacks from internet sources.
6. An enforceable and easy to understand acceptable use policy which differentiates between school owned and personally owned devices.

Large government security requirements:

1. A strong firewall and proxy to keep unwanted people out.
2. Strong antivirus software and internet security software suites.
3. Strong encryption of all information.
4. White list authorized wireless connection, block all else.
5. All network hardware is in secure zones.
6. All hosts should be on a private network that is invisible from the outside.
7. Host web servers in a DMZ, or a firewall from the outside and from the inside.
8. Security fencing to mark perimeter and set wireless range to this.
9. Inventory controls of government owned mobiles.

**Hackers classification**

Several subgroups of the computer underground with different attitudes use different terms to demarcate themselves from each other, or try to exclude some specific group with which they do not agree. According to (Clifford R.D. 2006) a cracker or cracking is to “gain unauthorized access to a computer in order to commit another crime such as destroying information contained in that system”. These subgroups may also be defined by the legal status of their activities.

**White hat**

A white hat hacker breaks security for non-malicious reasons, perhaps to test their own security system or while working for a security company which makes security software. The term “white hat” in Internet slang refers to an ethical hacker. This classification also includes individuals who perform penetration tests and vulnerability assessments within a contractual agreement. The EC-Council, also known as the International Council of Electronic Commerce Consultants, is one of those organizations that have developed certifications, courseware, classes, and online training covering the diverse arena of Ethical Hacking.

**Black hat**

A “black hat” hacker is a hacker who “violates computer security for little reason beyond maliciousness or for personal gain”. Black hat hackers form the stereotypical, illegal hacking groups often portrayed in popular culture, and are “the epitome of all that the public fears in a computer criminal”. Black hat hackers break into secure networks to destroy data or make the network unusable for those who are authorized to use the network. They choose their targets using a two-pronged process known as the “pre-hacking stage”.

**Grey hat**

A grey hat hacker is a combination of a Black Hat and a White Hat Hacker. A Grey Hat Hacker may surf the internet and hack into a computer system for the sole purpose of notifying the administrator that their system has been hacked, for example. Then they may offer to repair their system for a small fee.

**Elite hacker**

A social status among hackers, elite is used to describe the most skilled. Newly discovered exploits will circulate among these hackers. Elite groups such as Masters of Deception conferred a kind of credibility on their members.

**Script kiddie**

A script kiddie (or skiddie) is a non-expert who breaks into computer systems by using pre-packaged automated tools written by others, usually with little understanding of the underlying concept.

**Neophyte**

A neophyte, "newbie" is someone who is new to hacking or phreaking and has almost no knowledge or experience of the workings of technology, and hacking.

**Blue hat**

A blue hat hacker is someone outside computer security consulting firms who is used to bug test a system prior to its launch, looking for exploits so they can be closed. Microsoft also uses the term BlueHat to represent a series of security briefing events.

**Hacktivist**

A hacktivist is a hacker who utilizes technology to announce a social, ideological, religious, or political message. In general, most hacktivism involves website defacement or denial-of-service attacks.

**Bots**

Bots are automated software tools, some freeware, that are available for the use of any type of hacker.

**Conclusion**

In conclusion I would like to summarize the main points.

Not all hackers are bad. There exist white hackers that do it not for malicious reasons. On the other hand not all hackers are really dangerous. Developers must seriously fear only elite hackers, but they must not forget about others.

We have also considered different types of attacks and vulnerabilities, so now we know from what we are defending ourselves. Developers can defend their software from many different attacks, but from some attacks, like fishing, only user by itself can defend. Vulnerabilities, like attacks, also divide on software and human vulnerabilities.

We have looked at automated means of doing attacks, this is malware. It can be extremely various types, using different vulnerabilities and producing different kinds of attacks

The next point is that computer security has many approaches that must be combined in different, very well-considered proportions, because in other case product will not be secure.

Another peculiarity of computer security is that it has special branch called network security. The problem with this direction is that it must have other methods for defense from attacks. And the biggest problem in it is that this defense often must provide users by themselves.

**References**

1. <http://en.wikipedia.org/wiki/Computer_security>
2. <http://en.wikipedia.org/wiki/Network_security>
3. [http://en.wikipedia.org/wiki/Attack\_(computing)](http://en.wikipedia.org/wiki/Attack_%28computing%29)
4. [http://en.wikipedia.org/wiki/Vulnerability\_(computing)](http://en.wikipedia.org/wiki/Vulnerability_%28computing%29)
5. [http://en.wikipedia.org/wiki/Hacker\_(computer\_security)](http://en.wikipedia.org/wiki/Hacker_%28computer_security%29)
6. <http://en.wikipedia.org/wiki/Malware>