

A Project Report
On
**MALWARE IDENTIFICATION IN
CYBERSPACE**

*Project Report submitted in partial
fulfillment for the award of the degree
of*

**Bachelor of Technology in Computer Science and
Engineering**

Submitted by
Gulshan Kumar (19SCSE1010396)
Sneha Mehta (19SCSE1010405)

**IN BRANCH OF
STUDY
Computer Science & Engineering**

Under the Supervision of
Mr. Himanshu Sharma



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA, INDIA**

DECEMBER - 2021



SCHOOL OF COMPUTING SCIENCE & ENGINEERING

PROGRESS REPORT REVIEW-1

Fall 2021-2022

B.Tech., / BCA / B.Sc., / M-Tech., / MCA / M.Sc.,

Project Details:

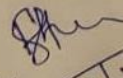
Project ID: **BT3159**

Project Title	MALWARE IDENTIFICATION IN CYBERSPACE
Progress of Project (in words)	Problem statement is defined & ready to implement
Research Paper Title	MALWARE IDENTIFICATION IN CYBERSPACE
Progress of Research Paper	Literature Review is in progress

Student Progress Details (Filled by Guide Only):

S. No	Name	Admission Number	No. Of time Came for Discussion	Performance of Student	Approval for Review 1
1	Gulshan kumar	19SCSE1010 392	4	<input type="checkbox"/> Satisfactory <input checked="" type="checkbox"/> Good <input type="checkbox"/> Poor	<input checked="" type="checkbox"/> Approved <input type="checkbox"/> Not Approved
2	-Sneha Mehta	19SCSE1010 405	4	<input type="checkbox"/> Satisfactory <input checked="" type="checkbox"/> Good <input type="checkbox"/> Poor	<input checked="" type="checkbox"/> Approved <input type="checkbox"/> Not Approved
3				<input type="checkbox"/> Satisfactory <input type="checkbox"/> Good <input type="checkbox"/> Poor	<input type="checkbox"/> Approved <input type="checkbox"/> Not Approved

Guide Signature with Date


13/10/21



SCHOOL OF COMPUTING SCIENCE AND
ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled “ Malware in Identification in Cyberspace ” in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the **School of Computing Science and Engineering** of Galgotias University, Greater Noida, is an original work carried out during the period of JULY-2021 to DECEMBER-2021, under the supervision of **Mr. Himanshu Sharma**, Assistant Professor, Department of Computer Science and Engineering of School, Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

19SCSE1010396 – GULSHAN KUMAR

19SCSE1010405 – SNEHA MENTA

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

Mr. Himanshu Sharma,
Assistant Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **Gulshan Kumar – 19SCSE1010396 & Sneha Mehta – 19SCSE1010405** has been held on _____ and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING.**

Signature of Examiner(s)

Signature of Supervisor(s)

Signature of Project Coordinator

Signature of Dean

Date:

Place:

Statement of Project Report Preparation

1. Thesis title: *MALWARE IDENTIFICATION IN CYBERSPACE*
2. Degree for which the report is submitted: **B.tech CSE**
3. Project Supervisor was referred to for preparing the report.
4. Specifications regarding thesis format have been closely followed.
5. The contents of the thesis have been organized based on the guidelines.
6. The report has been prepared without resorting to plagiarism.
7. All sources used have been cited appropriately.
8. The report has not been submitted elsewhere for a degree.

(Signature of the student)

Name: Gulshan Kumar

Roll No.: 19SCSE1010396

Name: Sneha Mehta

Roll No.: 19SCSE1010405

Abstract

Cyberspace is a new unlimited space where all actors, including countries, share information and communication technology, which is now essential to modern life. Since the beginning of the 21st century, the power of cyberspace has become a very important source of energy. Due to the proliferation of ICT systems in all aspects of life, the importance of knowledge in political matters has increased dramatically. State and non-State actors can use this power to achieve goals online and in the virtual world. The low cost and the great potential impact makes the cyber power appeal to all actors. In fact, cyber threats have grown significantly with the proliferation of cyberspace infrastructure. As a result, cyberspace has become a battleground with the potential to destroy or create irrational, tangible, technological, and virtual infrastructure, damaging critical National capabilities.

This situation forces all national institutions to revise their defense strategies, due to the difficulty of identifying cyber-attack actors. After that it is necessary to get a broad view of the problem to get detailed information, which is useful to find those sources of cyberattacks. This new point of view can be obtained using the analytical method developed by the authors and used in the streaming of data that flows across the communication space. In this way we can collect, detect, classify and analyze the effectiveness of those malware that acts as cyber weapons, using a honeypot-based system such as the one presented in this paper.

Keywords: Cyberspace, Digital Profiling, Malware, Cyber Threat, Honeypot, Cyber Weapon, Digital Behavior.

Table of Content

			Pg. no.
Progress Report			
Candidates Declaration			
Certificate			
Statement of Project Report Preparation			
Abstract			
Acronyms			
	Chapter 1	Introduction	
	Chapter 2	Profiling the Cyberspace	
	Chapter 3	Literary Survey	
	Chapter 4	The method of Analysis	
		<ul style="list-style-type: none"> i. Analysis of properties of cyber weapons ii. Analysis of lifetime of a cyber-weapon iii. Cyber Defense analysis iv. Implementation of filters to monitor data flow 	
	Chapter 5	The Experiment	
		<ul style="list-style-type: none"> i. Honeypot Implementation ii. Experimental Results iii. Analysis of the Activities Performed after the Intrusions 	
	Chapter 6	On the Construction of a Profile of Attack	
	Chapter 7	Conclusion	
	Chapter 8	Reference	

Introduction

Cyberspace is a unique domain that does not occupy a physical space. It happens, however, depending on the physical nodes, servers, and terminals found in the states have the power to control and sometimes become owners, as defined in the definition of the U.S. Department of Defense “A wide range of information, and contains a network of dependencies infrastructure, including telecommunications networks, computer programs, processors and embedded controls ”. Let's break this description down between cyberspace and activities that take place within it. This means that cyberspace, unlike the known physical space, has no national boundaries. In fact, while it is possible to differentiate or disable one or more risk factors the network, its functions and data continue to exist. This unique feature of Cyberspace influences any defense strategy we want to use. In that situation, cyber threats have grown significantly. As a result, cyberspace has become a battleground with the potential to destroy or make senseless infrastructure physically and corruptly and also damage the critical world skills.

The threats within cyberspace are different, they vary, and some may not be equal to the harm they can cause. This means the correct explanation of cyber weapons is very important to explore, the level of threat of cyberattacks is very appropriate, the opposition methods you must take, for the purpose of self-defense and self-defense. Weapons they are all tools where, within a context, a person it can harm another person or thing, or it can protect itself from attack.

Cyberattacks, in the same way as conventional conflicts, are designed to inflict damage only on a particular opponent, usually in the event of a disability or disaster that is already in progress or in connection with birth, in order to get some kind of benefit.

In this paper we create a strategy based on digital expansion behavior from streams of data flowing through the Internet. Total tests are made using a collection of visible, straight-set honeypots with known weaknesses. The purpose is to collect log files, detect malware and ultimately separate those that serve as cyber weapons through use of the information obtained. The purpose of the tests is to find out important details about the characters of the cyber conflict, which provide real-time time for a vision of possible attack situations and the ability to use it effectively a cyber-defense system that is pre-configured for certain threats to be made compared.

Obviously, the problem is that it contains more malware and worm. The Internet is targeted by a number of activities, some through the collaborative distribution of traffic filtering policies, others through automated security testing. In addition, the literature section considers it a good practice to accept access control based on auditing or to use network discovery methods.

Profiling the Cyberspace

One of the biggest problems with cyber security is represented by its anonymity with the possible failure that cyberspace can pose to a cyber-threat person. In that case, it is difficult, if not impossible, to identify the enemy, because many traditional war challenges are highlighted and made online. One of the most important aspects of the situation is awareness of the situation, which is defined as “the continuous dissemination of natural knowledge, the integration of this knowledge and past knowledge to form a coherent picture, and the use of that image to guide further understanding and anticipation”. It is therefore important to get an idea that allows you to get that information. This can be achieved by using the analytical approach proposed in several studies on the Digital Profiling paradigm, which provides a detailed description of the threat has left. This approach is based on ethical testing analysis of major cyber threat characters. We describe their characteristics, in relation to their online life time. This action was carried out from two perspectives: ICT, which analyzes software architecture, and strategy, which exposes their use of strategy / military as real weapons of the crime. The combination of these two features allows us to unveil new additions. Analysis of these new structures, as well as old ones, allows us to ethically unleash a cyber-weapon. Therefore, cyber-profile results are made up of a series of data that can be used as “filters” for monitoring and analysis of data streams, in order to provide effective identification of the characters of the cyber conflict. In fact, this profile type allows real-time recognition of attack situations and facilitates the implementation of an effective cyber defense system.

CHAPTER 3

Literature Survey

Our society, economy, and critical infrastructures have become largely dependent on computer networks and information technology solutions. Cyber-attacks become more attractive and potentially more disastrous as our dependence on information technology increases. According to the Symantec cybercrime report published in April 2012, cyber-attacks cost US\$114 billion each year. If the time lost by companies trying to recover from cyber-attacks is counted, the total cost of cyber-attacks would reach staggering US\$385 billion. Victims of cyber-attacks are also significantly growing. Based on the survey conducted by Symantec which involved interviewing 20,000 people across 24 countries, 69% reported being the victim of a cyber-attack in their lifetime. Symantec calculated that 14 adults become the victim of a cyber-attack every second, or more than one million attacks every day.

Why cyber-attacks flourish? It is because cyber-attacks are cheaper, convenient and less risky than physical attacks. Cyber criminals only require a few expenses beyond a computer and an Internet connection. They are unconstrained by geography and distance. They are difficult to identify and prosecute due to anonymous nature of the Internet. Given that attacks against information technology systems are very attractive, it is expected that the number and sophistication of cyber attacks will keep growing.

Cybersecurity concerns with the understanding of surrounding issues of diverse cyber attacks and devising defense strategies (i.e., countermeasures) that preserve confidentiality, integrity and availability of any digital and information technologies .

Confidentiality is the term used to prevent the disclosure of information to unauthorized individuals or systems.

Integrity is the term used to prevent any modification/deletion in an unauthorized manner.

Availability is the term used to assure that the systems responsible for delivering, storing and processing information are accessible when needed and by those who need them.

Many cybersecurity experts believe that malware is the key choice of weapon to carry out malicious intends to breach cybersecurity efforts in the cyberspace. Malware refers to a broad class of attacks that is loaded on a system, typically without the knowledge of the legitimate owner, to compromise the system to the

benefit of an adversary. Some exemplary classes of malware include viruses, worms, Trojan horses, spyware, and bot executables. Malware infects systems in a variety of ways for examples propagation from infected machines, tricking user to open tainted files, or alluring users to visit malware propagating websites. In more concrete examples of malware infection, malware may load itself onto a USB drive inserted into an infected device and then infect every other system into which that device is subsequently inserted. Malware may propagate from devices and equipment's that contain embedded systems and computational logic. In short, malware can be inserted at any point in the system life cycle. Victims of malware can range anything from end user systems, servers, network devices (i.e., routers, switches, etc.) and process control systems such as Supervisory Control and Data Acquisition (SCADA). The proliferation and sophistication of fast growing number of malware is a major concern in the Internet today.

Traditionally, malware attacks happened at a single point of surface amongst hardware equipment, software pieces or at network level exploiting existing design and implementation vulnerabilities at each layer. Rather than protecting each asset, the perimeter defense strategy has been used predominantly to put a wall outside all internal resources to safeguard everything inside from any unwanted intrusion from outside. The majority of perimeter defense mechanism utilizes firewall and anti-virus software installed within intrusion prevention/detection systems. Any traffic coming from outside is intercepted and examined to ensure there is no malware penetrating into the inside resources. General acceptance of this perimeter defense model has occurred because it is far easier and seemingly less costly to secure one perimeter than it is to secure a large volume of applications or a large number of internal networks. To give more defined access to certain internal resources, the access control mechanisms have been used in conjunction with the perimeter defense mechanism. On top of perimeter defense and access control, accountability is added to identify or punish for any misbehaviors, as represented in Fig. 1. However, the combined efforts of perimeter defense strategy have been found to be increasingly ineffective as the advancement and sophistication of malware improves. Ever evolving malware always seems to find loopholes to bypass the perimeter defense altogether. We describe in details the most common exploitations in the three distinct layers of existing information system at hardware, software and network layers. We then discuss the pros and cons of the most representative defense mechanisms that have been used in these layers.

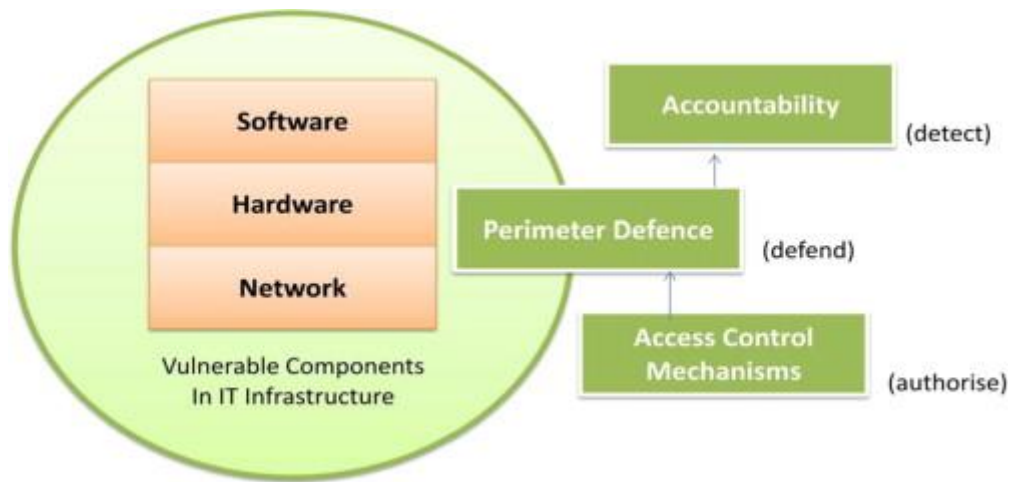


Fig. 1. Vulnerabilities and defense strategies in existing systems.

Malware evolves through time capitalizing on new approaches and exploiting the flaws in the emerging technologies to avoid detection. We describe a number of new patterns of malware attacks present in the emerging technologies. In choosing emerging technologies for illustration, we focus a few that have changed the way we live our daily life. These include social media, cloud computing, smartphone technology, and critical infrastructure. We discuss unique characteristics of each of these emerging technologies and how malware utilizes the unique characteristics to proliferate itself. For example, social media, such as social networking sites and blogs, are now an integral part of our life style as many people are journaling about their life events, sharing news, as well as making friends. Realizing its potential to connect millions people at one go, adversaries use social media accounts to befriend unsuspecting users to use as vehicles for sending spam to the victim's friends while the victim's machine is repurposed into a part of botnet. Cloud computing paradigm allows the use of computer resources like utilities where the users pay only for the usage without having to set up any upfront expense or requiring any skills in managing complex computing infrastructure. The growing trove of data concentrated in the cloud storage services is now attracting attackers. In June 2012, attackers compromised Distributed Denial of Service (DDoS) mitigation service on CloudFlare by using flaws in AT&T's voicemail service for its mobile users; similarly, Google's account-recovery service for its Gmail users . With the subjected growth by 2 billion smartphone users by 2015, a significant growth in mobile malware has been witnesses in recent times. For example, the number of unique detections of malware for Android increased globally by 17 times in 2012 from the previous year. There is also growing concerns in cyber threats to critical infrastructure such as electricity grids and healthcare systems to use in terrorism, sabotage and information warfare. Apart from investigating exploitations through unique characteristics in the selected emerging

technologies, we also discuss general malware attack patterns appear in them to understand the methods and trends of the new attacks.

Finally, we provide our speculative observations as where future research directions are heading. These include:

- (1) privacy concerns to safeguard increasing volumes of personal information entered in the Internet,
- (2) requirement to have a new generation of secure Internet from scratch with careful consideration of the subjected growth and usage patterns which was not the case with the internet we use today,
- (3) trustworthy system whose fundamental architecture is different from their inception to withstand from ever evolving malware,
- (4) being able to identify and trace the source of attacks assisted by the development of global scale identity management system and traceback techniques, and
- (5) a strong emphasis on usable security to give individuals security controls they can understand and control.

2. Malware as attack tool

In early days, malware was simply written as experiments often to highlight security vulnerabilities or in some cases to show off technical abilities. Today, malware is used primarily to steal sensitive personal, financial, or business information for the benefit of others. For example, malware is often used to target government or corporate websites to gather guarded information or to disrupt their operations. In other cases, malware is also used against individuals to gain personal information such as social security numbers or credit card numbers. Since the rise of widespread broadband Internet access that is cheaper and faster, malware has been designed increasingly not only for the stealth of information but strictly for profit purposes. For example, the majority of widespread malware have been designed to take control of user's computers for black market exploitation such as sending email spam or monitoring user's web browsing behaviors and displaying unsolicited advertisements. Based on Anti-Phishing group report, there was a total of 26 million new malware reported in 2012. [Fig.2](#) describes relative proportions of the types of new malware samples identified in the second half of 2012 reported by the Anti-Phishing group.

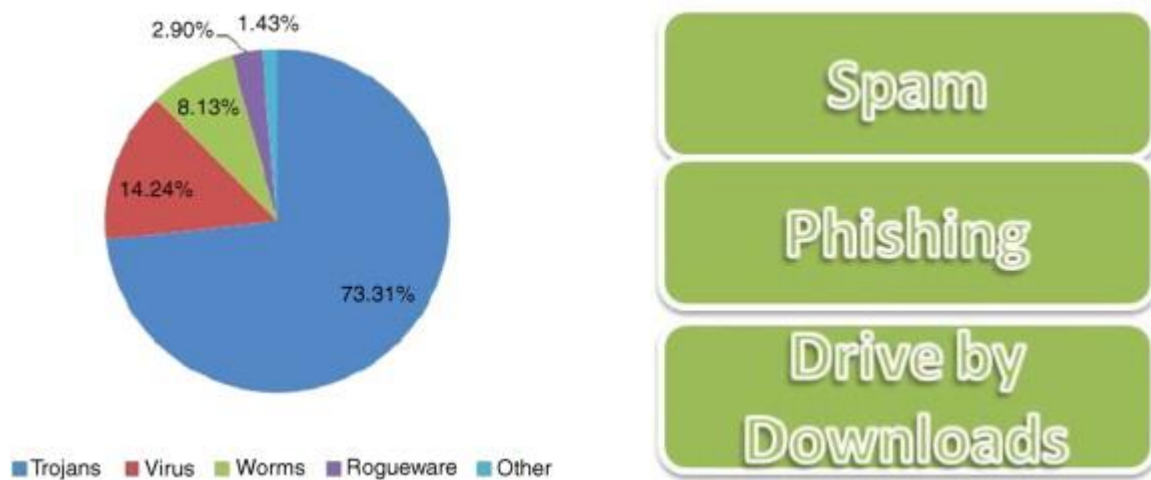


Fig. 2. Types of malware and mediums to spread them

According to this report, Trojans continued to account for most of the threats in terms of malware counting as the number grows spectacularly. In 2009, Trojans were reported to have made up 60 percent of all malware. In 2011, the number has jumped up to 73 percent. The current percentage indicates that nearly three out of every four new malware strains created in 2011 were Trojans and shows that it is the weapon of choice for cyber-criminals to conduct network intrusion and data stealing.

Malware authors use a number of different intermediaries to spread malware to infect a victim's system. Traditionally, spam, phishing and web download have been the most commonly used mediums for the purpose.

Spam refers to sending irrelevant, inappropriate and unsolicited messages to thousands or millions of recipients. Spam has turned out to be a highly profitable market since spam is sent anonymously with no costs involved beyond the management of mailing lists. Due to such low barrier to entry, spammers are numerous, and the volume of unsolicited mail has grown enormously. In the year 2011, the estimated figure for spam messages is around seven trillion. This figure includes the cost involved in lost productivity and fraud, and extra capacity needed to cope with the spam. Today, most widely recognized form of spam is email spam. According to the Message Anti-Abuse Working Group report, between 88–92% of email messages sent in the first half of 2010 carried spam.

Phishing is a way of attempting to acquire sensitive information such as username, password or credit card details by masquerading as a trustworthy entity. Most phishing scams rely on deceiving a user into visiting a malicious web site claiming to be from legitimate businesses and agencies. Unsuspecting user enters private information in the malicious web site which is then subsequently used by malicious criminals. Most

methods of phishing use some form of technical deception designed to make a link in an email (and spoofed website) appear to belong to a legitimate organization, such as well-known bank. Misspelled URLs or the use of sub-domains are common tricks used by phishers. The Anti-Phishing technical report [101] stated that, there was a visible trend of phishers in 2011 to hide their intentions by avoiding the use of obvious IP host to host their fake login pages. Instead the phishers preferred to host on a compromised domain to avoid detection. It is reported that there was 16 percent drop in the number of phishing URLs containing the spoofed company name in the URL. These combined trends show how phishers are adapting as users becoming more informed and knowledgeable about the traits of a typical phish.

Drive-by Downloads concerns the unintended downloads of malware from the Internet and have been increasingly used by the attackers to spread malware fast. Drive-by downloads happen in a variety of situations; for example, when a user visits a website, while viewing an email message by user or when users click on a deceptive pop-up window. However, the most popular drive-by downloads occur by far when visiting websites. An increasing number of web pages have been infected with various types of malware. According to Osterman Research survey, 11 million malware variants were discovered by 2008 and 90% of these malware comes from hidden downloads from popular and often trusted websites. Before a download takes place, a user is first required to visit the malicious site. To lure the user into visiting a website with malicious content, attackers would send spam emails that contain links to the site. When unsuspecting user visits the malicious website, malware is downloaded and installed in the victim's machine without the knowledge of the user. For example, the infamous Storm worm makes use of its own network, multiple of infected computers, to send spam emails containing links to such attack pages.

3. Exploiting existing vulnerabilities

Once malware is carried out to the victim's system, cyber criminals could utilize many different aspects of existing vulnerabilities in the victim's system further to use them in their criminal activities. We examine most commonly exploited existing vulnerabilities in hardware, software, and network systems. This is followed by the discussion on existing efforts that have been proposed to mitigate negative impacts from the exploitations. The summary of the common attacks in the hardware, software and network layers are presented along with the examples of countermeasures in [Fig. 3](#).

	Hardware	Software	Network
Common attacks	<ul style="list-style-type: none"> • Hardware Trojan • Illegal clones • Side channel attacks (i.e. snooping hardware signals) 	<ul style="list-style-type: none"> • Software programming bugs (e.g. memory management, user input validation, race conditions, user access privileges, etc.) • Software design bugs • Deployment errors 	<ul style="list-style-type: none"> • Networking protocol attacks • Network monitoring and sniffing
Examples of countermeasures	<ul style="list-style-type: none"> • Tamper-Resistant Hardware (e.g. TPM) • Trusted Computing Base (TCB) • Hardware watermarking • Hardware obfuscation 	<ul style="list-style-type: none"> • Secure coding practice (e.g. type checking, runtime error, program transformation, etc.) • Code obfuscation • Secure design and development • Formal methods 	<ul style="list-style-type: none"> • Firewall • Intrusion prevention and detection • Virtual Private Network (VPN) • Encryption

Fig. 3. Common attacks and examples of countermeasures in existing system.

3.1. Hardware

Hardware is the most privileged entity and has the most ability to manipulate a computing system. This is the level where it has the potential to give attackers considerable flexibility and power to launch malicious security attacks if the hardware is compromised. Compare to software level attacks where many security patches, intrusion detection tools, and anti-virus scanners exist to detect malicious attacks periodically, many of the hardware-based attacks have the ability to escape such detection. Taking advantage in lack of tools support in hardware detection, the hardware-based attacks have been reported to be on the rise.

Among different types of hardware misuse, hardware Trojan is the most hideous and common hardware exploits. The hardware Trojans are malicious and deliberately stealthy modification made to electronic devices such as Integrity Circuits (IC) in the hardware. The hardware Trojans have a variety of degrees which cause different types of undesirable effects. A hardware Trojan might cause an error detection module to accept inputs that should be rejected. A Trojan might insert more buffers in the chip's interconnections and hence consume more power, which in turn could drain the battery quickly. In more serious case, Denial-of-Service (DoS) Trojans prevent operation of a function or resource. A DoS Trojan can cause the target module to exhaust scarce resources like bandwidth, computation, and battery power. It could also physically destroy, disable, or alter the device's configuration, for example, causing the processor to ignore the interrupt from a specific peripheral.

Illegal clones of hardware become source of hardware-based exploitation since the chances of illegally counterfeited hardware to contain malicious backdoor or hardware Trojans increase. The chance to produce unauthentic hardware has increased with a new trend in IT companies trying to reduce their IT expense via outsourcing and buying off untrusted hardware from online sites. Karri et al. discusses how today's IT model of outsourcing has contributed to the increased chance of producing tampered hardware components from untrusted

factories in the foreign countries. Similarly, it is also pointed out that IT companies often buy untrusted hardware such as chipsets and routers from online auction sites or resellers which in turn may contain harmful hardware-based Trojans. These practices are not only problematic for IT companies operated on the tampered hardware with potential backdoor entry, it also increases the chance that the original design and the details of internal states of system to be leaked to unauthorized personnel.

Side channel attacks occur when adversaries gain information about a system's internal states by the examination of physical information of device such as power consumption, electromagnetic radiation and timing information of data in and out of CPU. Sensitive data can be leaked via the results of such side channel attacks. An approach has been reported in that examines a number of way cryptographic algorithm's secret key leaked as a result of analyzing radio frequency.

A number of techniques have been proposed to thwart attacks on hardware level. Tamper-resistant hardware devices have become an important consideration due to its criticality as an entry point to the overall system security. Trusted Platform Module (TPM) provides cryptographic primitives and protected storage along with the functionality to exchange tamper resistant evidence with remote servers. The term Trusted Computing Base (TCB) has been defined to refer to parts of a system, the set of all hardware and software components, to be critical to the overall security of the system. The TCB must not contain any bugs or vulnerabilities occurring inside because this might jeopardize the security of the entire system. An exhaustive and rigorous examination of its code base is conducted through computer-assisted software audit or program verification to ensure the security of TCB. In a hardware watermarking, the ownership information is embedded and concealed in the description of a circuit preventing the host object from illegal counterfeit. Hardware Obfuscation is a technique to modify the description or the structure of electronic hardware to intentionally conceal its functionality. These techniques are used to prevent adversaries from obtaining the original design or counterfeiting /cloning important parts of the hardware such as IC units. Some of the countermeasures to count against side channel attacks includes introducing noises so that the physical information cannot be directly displayed, filtering some parts of physical information, and making/blinding which seeks to remove any correlation between the input data and side channel emission.

3.2. Software defects

A software bug is the common term used to describe an error, flaw, mistake, or fault in a computer program such as internal OS, external I/O interface drivers, and applications. Cyber-attacks utilize the software bugs in their benefits to cause the systems to behave unintended ways that are different from their original

intent. The majority of cyber-attacks today still occur as a result of exploiting software vulnerabilities caused by software bug and design flaws.

Software-based exploitation occurs when certain features of software stack and interface is exploited. Most common software vulnerabilities happen as a result of exploiting software bugs in the memory, user input validation, race conditions and user access privileges. Memory safety violations are performed by attackers to modify the contents of a memory location. Most exemplary technique is buffer overflow. The buffer overflow occurs when a program tries to store more data in a buffer than it was intended to hold. Since buffers are created to contain a finite amount of data, the extra information can overflow into adjacent buffers, corrupting or overwriting the valid data held in them. It allows attackers to interfere into existing process code. Input validation is the process of ensuring that the input data follows certain rules. Incorrect data validation can lead to data corruption such as seen in SQL injection. SQL injection is one of the most well-known techniques that exploit a program bug in a website's software. An attacker injects SQL commands from the web form either to change the database content or dump the database information like credit cards or passwords to the attacker. Adversary exploits a flaw in a process where the output of the process is unexpectedly and critically dependent on the timing of other events. The time of check to time of use is a bug caused by changes in a system between the checking of a condition and the use of the results of that check. It is also called exploiting race condition error. Privilege confusion is an act of exploiting a bug by gaining elevated access to resources that are normally protected from an application or user. The result is that adversaries with more privileges perform unauthorized actions such as accessing protected secret keys.

In the programming community, a number of projects have been initiated that are devoted to increasing the security as a major goal. Not only attending to fix inherent common set of security flaws, the primary concern of these projects is to provide new ideas in an attempt to create a secure computing environment. In a code review-based secure coding practice, software engineers identify common programming errors that lead to software vulnerabilities, establish standard secure coding standards, educate software developers, and advance the state of the practice in secure coding. In a language-based secure coding practice, techniques are developed to ensure that programs can be relied on not to violate important security policies. The most widely used techniques include analysis and transformation. A well-known form of analysis is “type checking” where the program detects any unsafe type of objects before the program is run. Another well-known form of program transformation is the addition of runtime checks where the program is instrumented in a way that prevents the program from making any policy-violating transformation. Code obfuscation is a process of producing source or machine code that has been made difficult to understand for humans. Programmers often deliberately obfuscate code to conceal its purpose or its logic to prevent any possibility with reverse engineering. Secure design and

development cycle has also been proposed in which provides a set of design techniques enabling efficient verification that a piece of system component is free of any potential defects from its original design. Though they are not straightforward approaches, formal methods provide the ability to comprehensively explore the design and identify intricate security vulnerabilities. Tools and techniques have been developed to facilitate the verification of mission critical security properties. These tools and techniques help to translate higher-level security objectives into a collection of atomic properties to be verified.

3.3. Network infrastructure and protocol vulnerabilities

The early network protocol was developed to support entirely different environment we have today in a much smaller scale and often does not work properly in many situations it is used today. Weaknesses in network protocols are complicated when both system administrators and users have limited knowledge of the networking infrastructure. For example, the system administrators do not use efficient encryption scheme, do not apply recommended patches on time, or forget to apply security filters or policies.

One of the most common network attacks occurs by exploiting the limitations of the commonly used network protocols Internet Protocol (IP), Transmission Control Protocol (TCP) or Domain Name System (DNS). The IP is the main protocol of the network layer. It provides the information needed for routing packets among routers and computers of the network. The original IP protocol did not have any mechanism to check the authenticity and privacy of data being transmitted. This allowed the data being intercepted or changed while they are transmitted over unknown network between two devices. To prevent the problem, IPsec was developed to provide encryption of IP traffic. In many years, IPsec has been used as one of the main technology for the creation of a virtual private network (VPN) which creates a secure channel across the Internet between a remote computer and a trusted network (i.e., company intranet). TCP sits on top of the IP to transmit the packets in reliable (i.e., retransmitting lost packets) and ordered delivery of the packets. SSL was originally developed to provide end-to-end security, as oppose to only layer-based protocol, between two computers which sits over the transmission control protocol (TCP). SSL/TLS is commonly used with http to form https for secure Web pages. The domain name server (DNS) is the protocol that translates the human-readable host names into 32-bit Internet protocol (IP) addresses. It is essentially works as a directory book for the Internet telling routers to which IP address to direct packets when the user gives a url. Because DNS replies are not authenticated, an attacker may be able to send malicious DNS messages to impersonate an Internet server. Another major concern about DNS is its availability. Because a successful attack against the DNS service would create a significant communication disruption in the Internet, DNS has been the target of several Denial-of-Service (DoS) attacks.

Cryptography is an essential tool to protect the data that transmits between users by encrypting the data so that only intended users with appropriate keys can decrypt the data. Cryptography is the most commonly used mechanism in protecting data. A survey conducted by Computer Security Institute in 2007 revealed that 71% of companies utilized encryption for their data in transit. Further to protect today's sophisticated attackers exploiting the limitations of existing cryptography algorithms, a number of movements are on the rise. The US National Institute of Standards and Technology (NIST) recently announced discontinuation of SHA-1 and to use the Advanced Hash Standard (ASH) from 2012. The potential to use identity-based encryption is an active research agenda for applications that require high-speed encryption to avoid the use of slow 2048 bit RSA key length along with impractical involvement of the trusted certifying authority. Quantum cryptography is an emerging technology in which two parties simultaneously generate shared, secret cryptographic key material using the transmission of quantum states of light.

Skilled adversaries today use a sophisticated technique that disguises malicious traffic payloads that look more like legitimate traffic payloads. In addition, the large volume of data flow on high capacity networks requires new analysis techniques to calculate and also visualize the uncertainty attached to data sets. This challenge has created a new area of research where the combined skill sets from network practitioners and visualization community is required to capture the network traffic with better visualization techniques. The visual presentation of the data is then analyzed by network experts with in-depth domain knowledge in networking system.

3.4. Discussion

Though many separate techniques and proposals exist to remedy vulnerabilities in hardware, software and network layers, rather than focusing on each layer, bundled security protection techniques that protect everything inside from outside attacks have been adopted in the traditional approach. The overwhelming majority of companies employ a perimeter defense security model to guard the company's network from any potential intrusion from outside. This approach focuses on “layered defense” or “defense in depth” strategies in which important internal IT assets, such as servers or mission critical data, are protected by walls and fortifications.

Typical perimeter defenses include technologies such as firewalls and intrusion detection systems (IDS). The firewall has been the most widely used technology to protect the internal assets. Its primary objective is to control the incoming and outgoing network traffic by analyzing the data packets and determining whether it should be allowed through or not, based on a predetermined rule set. A firewall can be placed in different layer in the network infrastructure. Network layer

firewalls, also called packet filters, operate at a relatively low level of the network layer and prevent packets to pass through the firewall unless they match the established rule set (i.e., configurations) defined by network administrators. Though many modern firewalls are more sophisticated, the network layer firewalls cannot filter undesired traffic, such as malware payload, that utilizes legitimate IP addresses and ports. Application layer firewall operates by monitoring and potentially blocking the input, output, or system service calls which do not meet the configured policy of the network layer firewall. A proxy server may act as a firewall by responding to the input packets (for example, connection requests) in the manner of an application while blocking other packets. Both application layer firewall and proxies make tampering with an internal system more difficult. But with the increased capability and sophistication, attackers today have devised more advanced attack methods to pass malicious packets to a target network. For example, intruders may hijack a publicly-reachable system and use it as a proxy for their own purposes. Using the intercepted proxy, the intruder creates the packets with a forged IP address with the purpose of concealing the identity of the sender or impersonating another computing system.

The intrusion detection systems filter any suspicious or anomalous activity over the network. These detect systems are valuable in a way that they seek to detect the early stages of an attack (e.g., an attacker's probing of a machine or network for specific vulnerabilities) and can then aid in protecting a machine from the subsequent stages of the attack. Also, these systems seek to detect telltale signs of suspicious activities or patterns of behavior whether by a user, an application, or a piece of malicious code that firewalls or other protection tools might miss or ignore. Many detect system variants exist to identify malicious network payloads. Such detections are either signature-based or anomaly-based. In signature-based, the detection system recognizes attack packets due to their well-known fingerprints or signatures as those packets cross the network's gateway threshold. In anomaly-based, the detection system has no prior knowledge of what bad packets are. The detection system determines what normal traffic is by examining the pattern, often in real-time, and reports abnormal traffic behaviors based on the analysis on the pattern. The signature-based detection system has been considered ineffective as the proliferation and sophistication of malware writer have improved in recent years. It is considered that it is almost impossible to catch up ever evolving malware signature with pattern recognition methods popularly used in the signature-based approach. Proposing advanced anomaly-based detections have been an active research area. In this method, the system learns by example (self-learning) what constitutes normal by observing traffic for an extended period of time and building some model of the underlying process. The process is evolved (self-adaptive) as the signature of malware evolves.

Rather than focusing on fixing specific aspects of firewalls and IDS, more general approaches to understand the network attack patterns are needed in order to

devise better mechanisms to thwart undesired traffic coming from external sources. The area of network forensic involves the study of monitoring and analysis of network traffic by eavesdropping to Ethernet, TCP/IP, or the Internet including web browser, email, newsgroup, synchronous chat, and peer-to-peer traffic. The evidences are used for legal action or to understand the network traffic attack patterns. eMailTrackerPro analyzes the header of an email to detect the IP address of the machine that sent the message so that the sender can be tracked down. For web browser traffic forensic, the tools such as SmartWhoIs allow to look up all the available information about an IP address, hostname or domain, including country, state or province, city, name of the network provider, administrator and technical support contact information. Web Historian assists users in reviewing web site URLs that are stored in the history files. The Index.dat analyzer is a forensic tool to investigate index.dat files to examine the browsing history, the cookies and the cache. WinPcap captures the packets intercepted at the network interface of a system running the Windows Operating System while AirPcap is the packet capture tool for the IEEE 802.11b/g Wireless LAN interfaces. In research, honeypots are used to gather information about the motives and tactics of the cyber criminals. A honeypot is a trap set to detect, deflect, or in some manner counteract attempts at unauthorized use of resources. Any information captured by honeypots are used to research the threats organizations face and to learn how to better protect against those threats. Virtualization techniques are often employed to host multiple honeypots on a single physical machine. Therefore, even if the honeypot is compromised, there is a chance for quicker recovery with less expense. A large scale honeypots, such as honeynet which connects two or more honeypots on a network, is used for monitoring a larger and more diverse network. These honeynets are often implemented as parts of larger network intrusion detection systems. A honeyfarm is a centralized collection of honeypots and analysis tools.

As it is not possible to give uniform access to resources, access control mechanisms have been used to enable an authority to control access to only certain resources. In the access control, the entities that can perform actions in the system are called “subjects” and the entities representing resources to which access may need to be controlled are called “objects”. In the capability based access control, a subject is granted to access an object if the subject holds a reference or capability. For example, if a user provides a correct userID and password, the user is granted to view his/her bank statement. Access is conveyed to another party by transmitting such a capability over a secure channel. For example, a certificate is created for the user to present it for the verification purpose. In an access control list based approach, a subject's access to an object depends on whether its identity is on a list associated with the object. For example, if Alice is on the list of doctors, she is granted to view patient's records. Access is conveyed by editing the list. For example, when Alice leaves the hospital, she is no longer on the list of the doctors and won't be able to view the

patient's records. The three most widely recognized models are Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role Based Access Control (RBAC). In DAC, the owner decides who is allowed to access certain objects and what privileges they have. In MAC approach, it is the operating system constrains the ability of subjects (e.g. process or thread) to access or perform operation on objects (e.g. files, directories, TCP/UDP ports, or shared memory segments), either by a rule that defines specific conditions or by a mathematical structure that defines greatest lower-bound and least upper-bound values. RBAC is a newer alternative approach to MAC and DAC which restricts the system access only to authorized users. It is used by the majority of enterprises and most IT vendors offer RBAC in one or more products.

Traditional Access control systems provide the essential services such as authentication, authorization, and accountability. Authentication and Authorization is the process of verifying that a subject is bound to an object. Traditional authentication and authorization mechanisms use three different factors to identity a subject to verify if the subject has a right capability to access the object. First factor is something you know, for example, password or a personal identification number (PIN). This assumes that only the owner of the account who knows the password or PIN needed to access the account. Second factor is something you have which includes smart card or security token. This assumes that only the owner of the account has the necessary smart card or token needed to unlock the account. Third factor is something you are, such as fingerprint, voice, or iris characteristics. The current trend in authentication is layered approach, often called as strong authentication, replying on the presentation of two or more authentication factors. A number of different types of pocket-sized authentication tokens have been proposed. These tokens contain a cryptographic key in tamper-resistant storage. Taking advantage of ubiquitous nature of today's computer with USB ports, USB-based tokens have also proposed either simply as a storage of a X.509 certificate or often running challenge/response protocol. To take advantage of ever fast growing population of mobile users, a number of authentication mechanisms targeting mobile users have been proposed. Biometric technology has been used in limited applications. Some PC and workstations have become more sophisticated with audio-visual interfaces, there is renewed interest in employing biometric authentication technology in the network environment.

Accountability is another aspect of access control which involves the study that ensures anyone or anything that has access a system component, such as a computing device, an application, a network, can be held accountable for the results of such access. Accountability offers techniques and tools which can identify or punish for any misbehaviors. There is a variety of technique used by researchers in the accountability, most notable ones are logging, auditing and conflict resolution. The study of accountability typically starts with logging. A number of automated log files are created to record any access information, for

example, log files to record user logons and logoffs, application started, or files accessed. Such history logs should be sufficient enough to provide evidence for any later disputes. Identification of critical information for logging is one of the focus areas. Tamper resistant logging techniques are being proposed. Audits are performed to ascertain the validity and reliability of information typically by examining logging files when a misuse case is detected (also used for detecting problems). Monitoring tools are commonly used in the audit process to analyze system's states and operations. Conflict resolution is offered as a way to deal with the root cause, for example, forbidding the violating services from further interaction or the inclusion of the violators into a blacklist. Privacy is an increasing concern in the area of accountability. How much data can be captured to use as evidence without violating the privacy of a user has been the question a number of researchers have tried to address.

4. Emerging threats

Cyber-attacks on cyberspace evolve through time capitalizing on new approaches. Most times, cyber criminals would modify the existing malware signatures to exploit the flaws exist in the new technologies. In other cases, they simply explore unique characteristics of the new technologies to find loopholes to inject malware. Taking advantages of new Internet technologies with millions and billions active users, cyber criminals utilize these new technologies to reach out to a vast number of victims quickly and efficiently. We select four such up and coming technology advancements which include: social media, cloud computing, smartphone technology, and critical infrastructure, as illustrative examples to explore the threats in these technologies. We discuss unique characteristics of each of these emerging technologies and analyze a number of common attack patterns presented in them, as summarized in [Fig. 4](#).

Common characteristics	Common attack patterns
<ul style="list-style-type: none"> • Millions and billions of active users • Became part of our daily life • No geographical boundaries • Accessed 24/7 from anywhere at anytime • Services are available via Internet connection using Web Browsers • Services offered by many different devices such as mobiles and tablets 	<ul style="list-style-type: none"> • Increased Attack through Web Browser • Increased attacks through social engineering websites • Increasing attacks coming from non-PC-based devices (e.g. mobiles, tablets, VoIP) • Increasing number of more organized attacks through botnet • Increasing number of attacks through the attackers with internal knowledge (i.e. insider threats)

Fig. 4. Emerging Technologies: Their common characteristics and common attack patterns.

4.2. Cloud computing

The efficiencies of moving data and applications to the cloud continue to attract consumers who store their data in DropBox and iCloud, use Gmail and Live mail

to handle email, and track their lives using services such as Evernote and Mint.com. Cloud computing is arguably one of the most significant technological shifts in recent times. The mere idea of being able to use computing in a similar manner to using a utility is revolutionizing the IT services world and holds great potential. Customers, whether large enterprises or small businesses, are drawn towards the cloud's promises of agility, reduced capital costs, and enhanced IT resources. IT companies are shifting from providing their own IT infrastructure to utilizing the computation services provided by the cloud for their information technology needs.

Cloud computing provides unique characteristics that are different from the traditional approaches. The five key characteristics of cloud computing include on-demand self-service, ubiquitous network access, location-independent resource pooling, rapid elasticity, and measured service, all of which are geared towards using clouds seamlessly and transparently. Resource pooling refers to the ability where no resources are dedicated to one user but instead are pooled together to serve multiple consumers. Resources, whether at the application, host or network levels, are assigned and reassigned as needed to these consumers. On demand self-service refers where the users can assign themselves additional resources such as storage or processing power automatically without human intervention. This is comparable with autonomic computing where the computer system is capable of self-management. Along with self-provisioning of resources, cloud computing is characterized with the ability to locate and release resources as rapidly as needed, the term often called as “elasticity”. This allows consumers to scale up the resources they need at any time to address heavy loads and usage spikes, and then scale down by returning the resources to the pool when finished. Measured service, also often called as pay as you go, enables the cloud to be offered as a utility where users pay on a consumption basis, much the same way it is done to pay utilities like electricity, gas and water.

Cloud computing is also a model of integration that delivers various resources to clients at different layers of the system and utilizes different resources. Generally speaking, the architecture of a cloud computing environment can be divided into 4 layers: the hardware layer (including data centers), the infrastructure layer, the platform layer and the application layer.

-

The hardware layer: This layer is responsible for managing the physical resources of the cloud, including physical servers, routers, switches, power and cooling systems. In practice, the hardware layer is typically implemented in data centers. A data center usually contains thousands of servers that are organized in racks and interconnected through switches, routers or other fabrics. Typical issues at hardware layer include hardware configuration, fault tolerance, traffic management, power and cooling resource management.

-

The infrastructure layer: This layer is also known as the virtualization layer. The infrastructure layer creates a pool of storage and computing resources by partitioning the physical resources using virtualization technologies such as Xen, Kernel based Virtual Machine and VMware. The infrastructure layer is an essential component of cloud computing, since many key features, such as dynamic resource assignment, are only made available through virtualization technologies.

-

The platform layer: Built on top of the infrastructure layer, the platform layer consists of operating systems and application frameworks. The purpose of the platform layer is to minimize the burden of deploying applications directly into VM containers. For example, Google App Engine operates at the platform layer to provide API support for implementing storage, database and business logic of typical web applications.

-

The application layer: At the highest level of the hierarchy, the application layer consists of the actual cloud applications. Different from traditional applications, cloud applications can leverage the automatic-scaling feature to achieve better performance, availability and lower operating cost.

However, in practice, clouds offer services that can be grouped into three categories: software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS). Applications running on or being developed for cloud computing platforms pose various security and privacy challenges depending on the underlying delivery and deployment models. In IaaS, the cloud provider supplies a set of virtualized infrastructural components such as virtual machines (VMs) and storage on which customers can build and run applications. The application will eventually reside on the VM and the virtual operating system. PaaS enables programming environments to access and utilize additional application building blocks. Such programming environments have a visible impact on the application architecture, such as constraints on which the application can request services from an OS. Finally, in SaaS, the cloud providers enable and provide application software as on-demand services.

Multi-tenancy is a feature unique to clouds which allows cloud providers to manage resource utilization more efficiently by partitioning a virtualized, shared infrastructure among various customers. For example, to isolate multiple tenants' data, Salesforce.com employs a query rewriter at the database level, whereas Amazon uses hypervisors at the hardware level. Virtualization is an important enabling technology in this area that helps abstract infrastructure and resources to be made available to clients as isolated VMs. Providing strong isolation,

mediated sharing, and secure communications between VMs are active research areas. Using a flexible access control mechanism that governs the control and sharing capabilities of VMs within a cloud host has been suggested as a potential solution. Because clients acquire and use software components from different providers, crucial issues include securely composing them and ensuring that information handled by these composed services is well protected. For example, a PaaS environment might limit access to well-defined parts of the file system, thus requiring a fine-grained authorization service.

Trust management and policy integration is an active area of research in cloud computing as the outsourcing model of the cloud, where the cloud providers control and manage user's data and services, forces the clients to have significant trust in their provider's technical competence. In cloud computing environments, the interactions between different service domains driven by service requirements are also dynamic, transient, and intensive. Thus, a development of trust framework has been proposed to allow efficient capturing of a generic set of parameters required for establishing trust and to manage evolving trust and interaction/sharing requirements. The cloud's policy integration is another active area of research to address challenges such as semantic heterogeneity, secure interoperability, and policy-evolution management. Furthermore, customers' behaviors can evolve rapidly, thereby affecting established trust values. This suggests a need for an integrated, trust-based, secure interoperation framework that helps to establish, negotiate, and maintain trust to adaptively support policy integration.

4.4. Critical infrastructure

The critical infrastructure systems that form the lifeline of a modern society and their reliable and secure operation are of paramount importance to national security and economic vitality. In most sense, the cyber system forms the backbone of a nation's critical infrastructures, which means that a major security incident on cyber systems could have significant impacts on the reliable and safe operations of the physical systems that rely on it. The recent findings, as documented in government reports, indicate the growing threat of physical and cyber-based attacks in numbers and sophistication on electric grids and other critical infrastructure systems. Cybersecurity related to critical infrastructure seeks to limit vulnerabilities of these structures and systems to:

- - *Terrorism* – person or groups deliberately targeting critical infrastructure for political gain. In the November 2008 Mumbai attack, the Mumbai central station and Taj hotel were deliberately targeted.

Sabotage – person or groups such as ex-employee, political groups against governments, environmental groups in defense of environment, for example seizure of Bangkok's International airport by protestors.

- *Information warfare* – private person hacking for private gain or countries initiating attacks to glean information and also damage a country's infrastructure. For example, a series of cyber-attacks that swamped website of Estonian organizations including Estonian parliament, banks, ministries, newspapers and broadcasters, amid the country's row with Russia about the relocation of an elaborate Soviet-era grave market and war graves.

- *Natural disaster* – hurricane or natural events which damage critical infrastructure such as oil pipelines, water and power grids.

Critical infrastructure protection is harder to address than information and communication technology (ICT) protection because of these infrastructures' interconnection complexity, which can lead to different kinds of problems. Consider the power grid, in which geographically dispersed production sites distribute power through different voltage level stations (from higher to lower voltage) until energy eventually flows into our houses. Both the production and distribution sites are typically controlled by supervisory control and data-acquisition (SCADA) systems, which are remotely connected to supervision centers and to the corporate networks (intranets) of the companies managing the infrastructures. The intranets are linked to the Internet to facilitate, for example, communication with power regulators and end clients. These links create a path for external attackers. Operators' access SCADA systems remotely for maintenance operations, and sometimes equipment suppliers keep links to the systems through modems. The prevalence of proprietary solutions and use of older versions plagued with vulnerabilities are sought to add another dimension to propose solutions to protect nation's crucial infrastructure.

As the research into the critical system is quite new, researchers are still trying to understand the nature of critical infrastructure systems. This includes understanding criticality in system, understanding interdependencies among systems and infrastructures, and identifying and quantifying consequences of attacks on the critical systems. Because of the tight dependency of these systems and millions of users in their daily life, it is important that the critical infrastructure operates on *24*7 bases without any downturn. Self-diagnostic techniques using heartbeats, challenge-response, built-in monitoring of critical functions and detection of process anomalies which can capture any signs of non-operative functions have been proposed. Another relevant topic of interest is the development of self-healing systems to pursue automated and coordinated attack response and recovery.

5. Future research direction

With the tremendous growth in the Internet availability and the advancement of Internet enabled devices, an increasing number of populations use the Internet in all wakes of their lives, often exposing highly sensitive personal information without realizing the consequences of data misuse. We speculate that the issues surrounding the end-user privacy will continuously grow into the future in accordance to the growing volume of personal information over the Internet. In addition, usability issues are gaining more attention as a way to provide end-user focused security mechanism where the users can intuitively learn and use them, without complexity or deep learning curve, to protect their data.

Traditionally the practice in the cybersecurity community has been based on incremental patches which rectify the current security and privacy issues and then moves onto next step. Some believe that this incremental approach has not worked well and will not be able to accommodate future needs since the original Internet was invented for a very different environment than how it is used today. An approach to think “outside box” without relying on the current computing system and the Internet but starting something afresh has been suggested to make a better use of the fast growing demands of the Internet.

Anonymous nature of the Internet has been defined as a source of the increasing cyber-attack and difficult to trace the offender. The global scale identity management and traceback techniques have become an active area of research as a strategic plan to thwart increasing number of cyber attackers in the future, especially when the critical infrastructure is involved. We delve into more detailed of these speculated future research directions in the following sections. The summary of the research questions and future research directions is illustrated in [Fig. 5](#).

Areas of research	Research question	Research directions
Privacy	How to enable users of the Internet to better express, protect, and control confidentiality of their private information?	Selective disclosure of data [65], protection of shared data [67], data sanitization [141], privacy policy [88]
Next generation secure internet	Is this possible to design the current Internet system from scratch without being restrained by the existing system?	Internet-scale validation [123,124], security from beginning [142], new rich content delivery [125], energy efficient protocol design [126], federation of heterogeneous networking environment [126]
Trustworthy systems	How to develop a computing system that is inherently secure, available, and reliable, despite environmental disruption, human errors, and attacks by hostile parties?	Development of secure hardware and software [45], architecture design [143], evaluation of trustworthiness [144], self-testing and self-diagnosing [145,147], self-reconfiguring [147], compromise resilient [146,148], and automated remediation [148]
Global-scale identity management and traceback techniques	What are approaches to develop a global-scale identity management which can identify and authenticate entities when accessing critical information systems from anywhere?	Federated Identity beyond single organization [148–151], Attack Attribute [96–98], Open Provenance Model [136], Data provenance and annotation [135], Provenance-aware storage [137]
Usable security	How to develop a security system that can be actually managed and controlled by users with all different levels of computer skills?	Integration with HCI (human-computer interaction) [139], security interface design [138], evaluation of usable security [140]

Fig. 5. Future research questions and approaches.

5.1. Focus on privacy

In recent years, privacy has become a critical issue in the development of IT systems with the widespread of networked systems and the Internet. Now, the Internet is used in all wakes of our lives demanding increasing volume of personal information to be entered in the cyberspace. According to JP Morgan's annual report, global ecommerce sales has been increased at an annual rate of 19.4% reaching \$963 billion sales by 2013. This increase in online shopping suggests that the Internet users are becoming more comfortable sharing their sensitive financial information, such as credit card numbers and shipping addresses. Similarly, professional and social networking sites that connect people with similar interests online have seen an exponential growth in last decade. LinkedIn, a professional networking site founded in May 2003, have 200 million users by January 2013. Facebook, launched in February 2004, have reached 1 billion active users as of September 2012. These numbers indicate that people increasingly feel comfortable putting personal information about themselves online. Individuals also appear more willing to speak out about what they perceive as invasion of privacy when engaging in online activities.

As increasing volume of information is being put in the Internet, the chances of occurrence of compromise of privacy also increase. For example, individual's online visits are watched to infiltrate the information and send advertising based on one's browsing history. The methods of compromise can range from gathering of statistics on users, to more malicious act such as the spreading of spyware. Cyber criminals use the social networking sites to steal personal information to use in fraud and identity theft. To prevent such privacy leakage, several social networking sites provide privacy measures. For example, Facebook provides a privacy setting for all registered users. The settings available on Facebook include the ability to block certain individuals from seeing one's profile, the ability to choose one's "friends", and the ability to limit who has access to one's pictures and videos. Privacy settings are also available on other social networking sites such as Google Plus and Twitter. Children and adolescents are very susceptible to misusing the Internet and ultimately risking their privacy. There is a growing concern among parents whose children are now starting to use Facebook and other social media sites on a daily basis. Website information collection practices is another growing concern as young individuals are more vulnerable and unaware of the fact that all of their information and browsing can and may be tracked while visiting a particular site.

The goal of privacy-aware security is to enable users and organizations to better express, protect, and control the confidentiality of their private information, even when they choose to (or require to) share it with others. One stream of research in this field concerns with the way data is accessed and disclosed while protecting

privacy. A number of researches are conducted to investigate how to selectively disclose the data, how to protect the data that are shared by people, and how to sanitize the data. Another stream of research conducted in this area concerned with the development of specification framework to build and reinforce privacy policy. Development of building a number of specifications for providing privacy guarantees such as languages for specifying privacy policies, specifications for violations of privacy, and detecting violations of privacy is an active research area. Building techniques for data policy for data collection, data sharing and transmission, and dealing with privacy violations are other active areas of research in this category.

5.2. Next generation secure internet

There is no doubt that the Internet has been a social phenomenon that has changed, and continues to change how humans communicate, businesses work, how emergencies are handled, and the military operates among many other things. Despite the Internet's critical importance, some portions of the Internet is fragile and the constantly under incessant attacks that range from software exploits to denial-of-service. One of the main reasons for these security vulnerabilities is that the Internet architecture and its supporting protocols were primarily designed for a benign and trustworthy environment, with little or no consideration for security issues [125], [126]. This assumption is clearly no longer valid for today's Internet, which connects millions of people, computers, and corporations in a complex web that spans the entire globe.

In the past 30 years, the Internet has been very successful using an incremental approach where a system is moved from one state to another with incremental patches [123]. However, some believe that the entire Internet technology has now reached a point where people are unable to experiment new ideas on the current architecture. For example, a best effort delivery model of IP is no longer considered adequate without added security assurance. Routing is no longer based on algorithmic optimization, but rather has to deal with policy compliance to accommodate a wide range of applications. Protocols designed without concern for energy efficiency cannot integrate energy conscious embedded system networks such as sensor networks. Initial projections about the scale of the Internet have long since been invalidated, leading to the current situation of IP address scarcity.

A new paradigm of architectural design described as “clean-slate design” has been suggested [123], [124]. The theme of “clean-slate design” is to design the system from scratch without being restrained by the existing system, providing a chance to have an unbiased look at the problem space [142]. However, the scale of the current Internet forbids any changes, and it is extremely difficult to convince the stakeholders to believe in a clean-slate design and adopt it. There is simply too much risk involved in the process. The only way to mitigate such risks

and to appeal to stakeholders is through actual Internet-scale validation of such designs that show their superiority over the existing systems [123]. Despite the risk, research funding agencies all over the world have realized this pressing need and a world-wide effort to develop the next generation Internet is being carried out [123], [124]. The National Science Foundation (NSF) was among the first to announce a GENI (Global Environment for Networking Innovations) program for developing an infrastructure for developing and testing futuristic networking ideas developed as part of its FIND (Future Internet Design) program. The NSF effort was followed by the FIRE (Future Internet Research and Experimentation) program which support numerous next generation networking projects under the 7th Framework Program of the European Union, the AKARI program in Japan, and several other similarly specialized programs in China, Australia, Korea, and other parts of the world.

The “clean state design” idea can be approached in a number of areas. In the area of Internet security aspect, security mechanisms are placed as an additional overlay on top of the original architecture rather than as part of the Internet architecture. This includes proposals and projects related to security policies, trust relationships, names and identities, cryptography, anti-spam, anti-attacks, and privacy. Concerning on new mechanisms for content delivery over the Internet as the next generation Internet is set to see a huge growth in the amount of content delivered over the Internet, newer paradigms for networking with content delivery at the center of the architecture is proposed [123] rather than connectivity between hosts, as in the current architecture. Challenged network research focuses specifically on heterogeneous networking environments where continuous end-to-end connectivity cannot be assumed such as seen in the wireless ad hoc networks. The discussions in this area relate to two important perspectives of the future Internet design requirements: Energy efficient protocol design and implementation and federation of heterogeneous networking environments. Another area is the management and control framework. The current Internet works on a retro-fitted management and control framework that does not provide efficient management and troubleshooting. The proposals for the future Internet in this area vary from completely centralized ideas of management to more scalable and distributed ideas.

5.3. Towards trustworthy systems

Most of today's systems are built out of untrustworthy legacy systems using inadequate architectures, development practices, and tools. Hence, they are typically not well suited to deal with the attacks in cyberspace. Matters get worse as the modern devices are themselves networks of systems and components. They need to interact in complex ways with other components and systems, sometimes producing unexpected and potentially adverse behavior.

Historically, many systems claimed to have a trustworthy computing base (TBC) that was supposed to provide a suitable security foundation to safeguard the critical components. For example, error-correcting codes were developed to overcome unreliable communications and storage media. Encryption has been used to increase confidentiality and integrity despite insecure communication channels. Similarly, firewalls have been used to protect inside assets from outside attacks. However, the idea of having one specific solution to a particular problem has not been successful due to the continuous evolution of attacks.

The term trustworthy systems have been defined by the Department of Homeland Security (DHS) in US [15] as a long-term goal to indicate a computing system that is inherently secure, available, and reliable, despite environmental disruption, human user and operator errors, and attacks by hostile parties. Towards this goal, the author [45] advocates the requirement for secure hardware and software combinations as essential building block towards trustworthy system. In the proposal, systems and devices share provable and standard trust information confirming their trustworthiness, generic security-assured commodity hardware solutions at all levels, and systems able to determine whether to trust a device, software package, or network based on dynamically acquired trust information rooted in hardware and user defined security policies. Towards this goal, a several threads of research work have been carried away in the areas of trustworthy isolation technique [143], separation and virtualization in hardware and software [134], [143], analyzes that could greatly simplify evaluation of trustworthiness before putting applications into operation [144], robust architectures that provide self-testing and self-diagnosing [145], [147], self-reconfiguring [147], compromise resilient [146], [148], and automated remediation [148].

5.4. Global-scale identity management and traceback techniques

Identity management is the task of controlling information about users on computers. Such information includes information that authenticates the identity of a user, information that describes information and actions they are authorize to access and/or perform. It also includes the management of descriptive information about the user and how and by whom that information can be accessed and modified. Managed entities typically include users, hardware and network resources and even applications [15].

There are many current approaches to identity management. For example, many websites employ logging in process with username and password combination to screen only eligible users to enter into the service. However, many of these are not yet fully interoperable with other services across different organizations and scalable. They are only for single-use or limited in other ways. It has been pointed

out [15] that due to the lack of adequate identity management it is often extremely difficult to trace identity theft.

Global-scale identity management concerns identifying and authenticating entities such as people, hardware devices, distributed sensors and software applications when accessing critical information technology systems from anywhere. The term global-scale is intended to emphasize the pervasive nature of identities, due to increasing use of mobile phones and embedded sensors in everywhere of our daily life. This also implies the existence of identities in federated systems that may be beyond the control of any single organization [11], [148], [149], [150], [151].

Combined with the development of the global-identity management, an attack attribution technique could assist in determining the identity or location of an attacker or an attacker's intermediary. In the Ingress filtering technique [96], [97], [98], the source IP addresses of all inbound packets into the company's router are analyzed. Any packets containing suspected illegal source IP addresses are blocked or recorded. Similarly, Egress filtering techniques filters any outbound attack traffic. Marking [96], [97] is another commonly used traceback technique. A mark, typically an IP address or the edges of the path that the packet traversed to reach the router, is inserted into a packet and then used to trace the source of the attack. However, it is criticized that most current traceback methods only work well for a single cooperative defense and skilled attackers easily evade most currently deployed traceback systems by tweaking the header IP addresses [96], [15]. The development of global scale traceback system with a defense mechanism which can trace and block evolving packet signatures are listed as solutions required for the future computing environment.

Provenance technique is another notable one that has been emerging and provides an ability to trace the life time changes and transformation of computer related resources such as hardware, software, documents, database, data, and other entities [136]. The provenance aims to provide a good knowledge about the sources and intermediate processors of the data. This is to assist to access the data's trustworthiness and reliability at the decision-making process. Toward this goal, a number of ideas have been proposed. In the area of data pedigree, researchers suggest the use of directed graphs to make connection between the historical dependencies of data through the life cycle of data [137]. Tool developments are also proposed to assist the trace and identification of where resources went and how they have been used. In other area, researchers suggest that there require the development of techniques to assist the following up the original sources of any subsequent changes such as modifications made to resources throughout the life cycle of data. It is suggested [135] that current version control systems or the techniques used in the natural language translation and file compression could be useful to develop required techniques in this area.

5.5. Usable security

As the range of potential threats over the Internet expands, end users are increasingly find themselves in a position having to make security decisions, for example through configuring security-related settings, responding to security-related events and messages, or enforced to specify security policy and access rights [128]. Unfortunately, experience suggests that although security features are often provided, they are conveyed in a manner that is not understandable or usable for many members of the target audience. As most users unable to comprehend the security features on offer, many security enhancements remain unused leaving the end users in a vulnerable position from malicious attacks. The need for usable security and the difficulties inherent in realizing adequate solutions are increasing being recognized [99], [100].

Many security technologies have tried to improve the usability aspects; most of which fall short in terms of usability. Password schemes have been believed to be one important parts of usable security. Therefore, several elaborate procedures have been progressed such as frequency of changing, inclusion of non-alphabetic characters, or visual and biometric based passwords that users do not have to remember. Despite these attempts, security pitfalls of poorly implemented password schemes have been extensively documented over the years. Users resort to writing them on slips of paper or storing them unencrypted on handheld devices [15]. Mail authentication is another active area where usable security has been studied in a form to authenticate senders of valid emails. Security pop-up dialogs and SSL lock icons also have been proposed. Another issue that makes it difficult to devise an effective usable security scheme is that usability of systems tends to decrease as attempts are made to increase security. For example, some email system requires users to re authenticate in a regular time to assure that they are actually the authorized person. In another example, some web browsers warn users before any script is run. But users may still browse a web server that has scripts on every page causing pop-up alerts to appear on each page. The potential impacts of security that is not usable include increase susceptibility and vulnerable from social engineering type of cyber-attacks.

The research conducted in the field of HCI (human-computer interaction) to develop techniques for interface design, evaluation for usable security, and tool development have been discussed in [95], [99]. However, only a small fraction of this research has focused on usability related to security. At the same time, security research tends to focus on specific solutions to specific problems, with little or no regard whether they are practical to use and transparent to all different types of users. The authors [127], [139] argue that there needs research into the question of how to evaluate usability as it relates to security. A significant contribution can be made from HCI research that has already developed methodologies for evaluating usability [138], [139], [140].

6. Conclusion

This survey focused on two aspects of information system: understanding vulnerabilities in existing technologies and emerging threats in up and coming advancement in the telecommunication and information technologies. Growing threats have been found in emerging technologies, such as social media, cloud computing, smartphone technology and critical infrastructure, often taking advantage of their unique characteristics. We described characteristics of each of emerging technologies and various ways malware being spread in these new technologies. Then, we discuss common set of general attack patterns found in the emerging technology. For example, as most of these emerging technologies offer services through online, some of the common attacks increasingly exploit the browser security through malware hidden inside extensions or vulnerabilities exist in scripting languages to access confidential data. Adversaries are also switching their battle ground from desktop to other platforms including mobile phones, tablet PCs and VoIP to avoid detection. Especially mobile malware has risen sharply in the last few years with the growing number of mobile users and the sophistication of mobile applications. Scams using social engineering are on the rise. Popular social networking sites like Facebook, Twitters and others have been increasingly used as delivery mechanisms to get unsuspecting users to install or spread malware. More organized attacks through the use of botnets have been reported. As the impact of such damage is much bigger than individual attacks, there is a growing concern to thwart botnets. Recent statistics also show there is an increasing number of cyber-attacks tailored to a specific system, for example command and control system, using inside knowledge and personnel.

We also illustrated potential future research directions. As more and more people are connected over the Internet, understanding all levels of users including both experts and non-experts in computing system and devising security mechanisms corresponding to their confidence levels have been suggested. Preserving user privacy has been emphasized by many security experts as an important future research to carry out as the amount of personal information over the Internet has expanded rapidly in recent years. Rather than trying to fix a specific problem on existing Internet and computing systems incrementally, more innovative approaches to see “a bigger picture” or think “outside of the box” have been suggested, as some evidences suggest that the capacity of today's modern technology saturates and do not scale well any more using traditional incremental approaches. The developments of next generation secure Internet and trustworthy systems have been suggested as important areas of research to look into the future. The development of global scale identity management and traceback techniques to enable tracking down adversaries has also gained an attention as an important issue to address in the future.

The Method of Analysis

How to increase the effectiveness of a cyber-weapon consists of the following four steps. The first is to analyze the properties of the cyber weapon, which provides detailed information. This is followed by an analysis of the timeline of cyberattacks. It uses features derived from the previous step to extract detailed information that helps to explain the behavior of the attacker. The third step is to analyze cyber security, taking into account the data collected in the second step. Finally, the fourth step is the implementation of filters to monitor and analyze cyber threats, using the profile obtained in all previous steps.

a. Analysis of properties of cyber weapons

The study of the characteristics of cyber weapons is based on architecture from experiments performed using two different theories. The concept of ICT, which defines malware as any set of computer programs designed to hack a computer system illegally. Cyber weapons are actually the emergence of malware and all its properties. A strategic / military concept that highlights the impact of cyberattacks and the expected damage inflicted on an enemy target. This view adds further details about the target, such as critical infrastructure, data or programs contained in or related to it, using standard military strategic methods. A cyber weapon is a set of commands compiled into a programming language, and thus can be fragmented, analyzed and modified. Unlike the usual malware that affects any computer system, except for any kind of control or profit, it is specially designed for the features of the operating system, for the purpose of gaining a certain advantage. The cyber weapons code differs from one attack to another and is able to deal with different attacks at the same time.

The impact of the damage caused is revealed publicly by lag: similarly in all cases, the victim is unwilling to expose the risks. In addition, the source and track are difficult to locate, as their authors can take advantage of the anonymity offered by the Internet. A cyber weapon can injure itself after an attack, leaving no trace in the infected system. Any clue that is eventually left behind after an attack can be easily created by a deceptive advertisement or any attempt to identify it.

Cyber weapons are often used as part of a larger general attack to support it within a war, gaining more than an enemy. They can work for a while, stay “quiet” until the right moment for the attack actions, adapt to the programming environment in which it is presented, and change with the flexible response they encounter. These structures make them smarter, like "fire and forgettable" weapons. Usually, they have a very short life span, just a time of attack. Its availability indicates an immediate response to correct any exploited vulnerabilities. For that reason, cyber weapons should not be used over time without major changes. Implementing a cyber-weapon is a very difficult task. Unlike standard malware that can be created and delivered by a single person, it requires a C4ISTAR command & control (C&C) structure similar to those in other advanced botnet formats.

b. Analysis of lifetime of a cyber-weapon

The above features enable us to explain in detail where a cyber-weapon is presented in the field. To analyze the following six steps, we can use the methods that reflect the life of a cyber-weapon.

Target choice – Often the formation of cyberattacks occurs in a strategic way, from motives from the management of all attacks. Initially, the choice of target is related to the nature of the enemy and its significance and is closely related to the reasons for the attack. It is possible to explain the target selection in response to the following four questions. Where is the visible area of the target? What are targeted activities? Who the owner and users of the target? Why the attack? In this respect, can be determined by the type of damage you can cause, which can be of digital access with unauthorized access to confidential data, delay or service interruptions, alterations, damage or damage to computer code or caused by the destruction of devices and equipment. In addition, the damage is measured in terms of the magnitude of the effects of the attack and the persistence of the effects of either permanent, temporary or transient.

Acquisition of information – Category related to access to information in terms of the selected target is important to build the weapon itself, because its ability to successfully hit a certain target is equal to the quality and quantity of the information collected. Such information can be obtained primarily from a spy perspective (e.g., details of targeted selection, location, access route, physical defense systems, good timing of attacks,

etc.) or from a technical point of view (e.g., selected target technical features, vulnerability and security systems) hardware and software.

Source Code Analysis - Most cyber-weapons are specifically designed their purpose: we often find certain cyber-weapons in particular striking stones. This makes the cyber weapon more effective. In fact, when cyber a weapon was found, and some combat action was taken. This does it could no longer do so if the quality of the cyber weapon was high. The the configuration code is used depending on the type of login, can be:

- *align: connect to the target system with the transmitting device (USB mass storage, CDROM, etc.);*
- *Direct: sent via network from unspecified location;*
- *Indirect: posted via cyberspace.*

In addition, this "armed code" should use those properties that separate the common malware and make it an effective, anonymous weapon and hard to find. That is, the successful implementation of good cyber-weapon with the above features should consider when it should be introduced (immediate, delayed or duplicated) and must adapt to target system conditions, including self-destruction method and the possibility of connecting to a C&C server online.

Simulation and Testing – Cyberattacks must be successful in the first attempt, otherwise they can be easily mitigated. Its detection should include, as with any other software, the testing phase, before the actual attack. Initially it happens in the visible environment, to test the functionality of the code used, but it needs to be tested online, to correct any errors in the end, and especially to adapt to possible changes in the security measures taken by the target system. The purpose is to obtain information on the effectiveness of the entry methods and the intended damage, in order to ensure the success of the targeted attack. At this stage the type of attack is the same as the real one. In fact, the target program is built with a list of programs similar to those selected.

Attacks – The most important stage in the timeline analysis is the attack, in which all prepared actions, tested in previous phases, are performed. The aim is to successfully hit the target and get the answer as close as possible to the expected result in the right way and at the right time, to avoid any unwanted side effects.

Result Evaluation - The final stage consists of testing, both in the case and in the near future, the success of the attack by comparing the expected results compared to the actual findings. The first step is to ensure the successful attainment of the intended target, followed by an attack time test, the nature, duration, and cost of the damage caused to the target system. Later, it is also considered to be the result of damage to the infected system, the building in which it is located, and any impact in the short, medium and long term, such as adverse effects within and outside the target system, the impact of attacks (military / political / social), and resistance (ultimately / performance) measures. The evaluation of the above categories results in a comprehensive assessment of the attack in terms of cost / profit analysis and the timing of actual gains received.

c. Cyber Defense analysis

In order to have a detailed analysis of cyberattacks, an examination of its features from the point of view of the security building is very important. A study of well-known cyber weapons (from DDoS attacks in Estonia to Stuxnet worm), has confirmed that a computerized computer is often used as part of a larger general attack to support it. This observation led to the development of a monitoring system that could be useful in interpreting those indications that cyberattacks could be detected on sensitive infrastructure, by analyzing available data from a variety of sources. Such sources may be open to the public (such as national reports from antivirus companies, national and international newspapers, websites responsible for political, economic and social analysis), which may not be open if it contains websites of hijackers, opponents, extremists, activists and partisans. of strategic / military documents.

The information obtained should be able to respond to the following seven questions:

- *who: identification of potential attackers;*
- *why: reasons for the attack;*
- *where: identifying critical infrastructure that may be under consideration;*
- *how: access mode;*
- *what: type of damage;*
- *when: time of attack;*
- *consequences: the extent of the damage and possible damage;*

- *reaction: reaction actions.*

d. Implementation of filters to monitor data flow

The data obtained from the aforementioned analysis constitutes the first set of filters that are active in the analysis of data streams to identify those indicators that a cyberattack may be imminent. The main step consists in detecting the presence of a cyber weapon by analyzing the characteristics of its behavior, which distinguishes it from the common malware. Possible target identification may be used to find similar properties between different malware. In particular, the target may be: numerically restricted and restricted to a particular type, distributed locally, by the same processes or sensitive data, with the same OS, with the same policy and security systems and, finally, with the same vulnerabilities.

The table lists the characteristics of malware detected which analyzes and highlight the work of cyber weapons. In addition, the real target (depending on cyber bullets) is highly targeted, attacked multiple times at different times, can be detected over time by intrusion or attack, reveals more fixes in malware code and is related to causes of conflict / difficulty / conflict / controversy, either national or worldwide. Information obtained during the analysis of the content of its source code, may provide the cyber weapon profile found. Possible clues extracted from it, can be used as filters to detect ongoing or imminent cyberattacks. Such filters can be applied to log files related to attempts to enter the domain of interest.

Feature	Meaning
Incomplete code	- developing code
Simultaneous diffusion of the same code in a limited number of objectives	- malware test on a controllable number of objective similar to target - refine tuning of malware code - deception - reduction of target reaction response time
Repeated attacks over time for the same purposes	- code corrections - deception
More attacks on the chosen target	- customize code on actual configuration of real target
No major damage caused as a results of the intrusion or attack	- decrease the possibility of detection by antivirus softwares - reduction of target reaction response time

The Experiments

The main goal of the tests is to demonstrate the application of the method outlined in previous sections. This will be used to implement a system to monitor and analyze the flow of data flowing through the Internet. The tests were performed on a small scale only from a technical point of view, using those filters based on information extracted from the analysis of the symptoms of cyberattacks found. To improve the tests, we used a honeypot network (called honeynet) where it collected, detected, extracted and analyzed malicious codes sent against it. Honeypot is a network-connected machine that removes system crashes to attract, capture and analyze cyberattacks. If a connection occurs, it could be an accidental connection or, which may be an attempt to attack the machine. In short, we can divide honeypots first into two groups, depending on their delivery. Production honeypots are used in the company's internal network to improve the security of the entire network. In addition, research honeypots are much more complex than those produced, and they provide detailed information on attacks and are used by research, military and government agencies.

The second condition separates honeypots based on their naming process. The Honeypots are pure full production systems, so no other software needs to be available installed. High-performance honeypots use multiple OS simulated services that can be used by the attacker. Also, low-performance honeypots mimic part of the system and the most widely used services.

a. Honeypots Implementation

We used the tools contained into the “Mercury Live DVD” [25]. It comprises valuable tools for digital forensics, data recovery, network monitoring, spoofing, reverse engineering, and four different type of honeypots: **Honeyd**, **Nepenthes**, **Dionaea**, and **Kippo**. In particular, Honeyd is a low-interaction honeypot that comprises several components (see [Figure 1\(a\)](#)): configuration database, a central packet dispatcher, protocol handlers and a personality engine. Incoming packets first go through the central packet dispatcher. It is able of dealing with three protocols, TCP, UDP and ICMP. The dispatcher queries the configuration corresponding to the destination address. Then it passes the packet to the protocol specific handler. On receiving a TCP or UDP packet, the handler manages the connections to different services. The framework checks if a specific packet is part of an already started service application. If so, all packets are redirected to the service, otherwise a new service is started. The

handler also helps in connections' redirection. Then the packet is sent to the personality engine which manipulates its content to make it appear similar to the one originated from the network stack. Through Honeyd we implemented a network with three routers and four simulated hosts, as in Figure 1(b). The Honeyd implementation also includes two hosts configured with two different versions of Microsoft Windows, one as a server and the other as a client. What follows is an example of configuration.

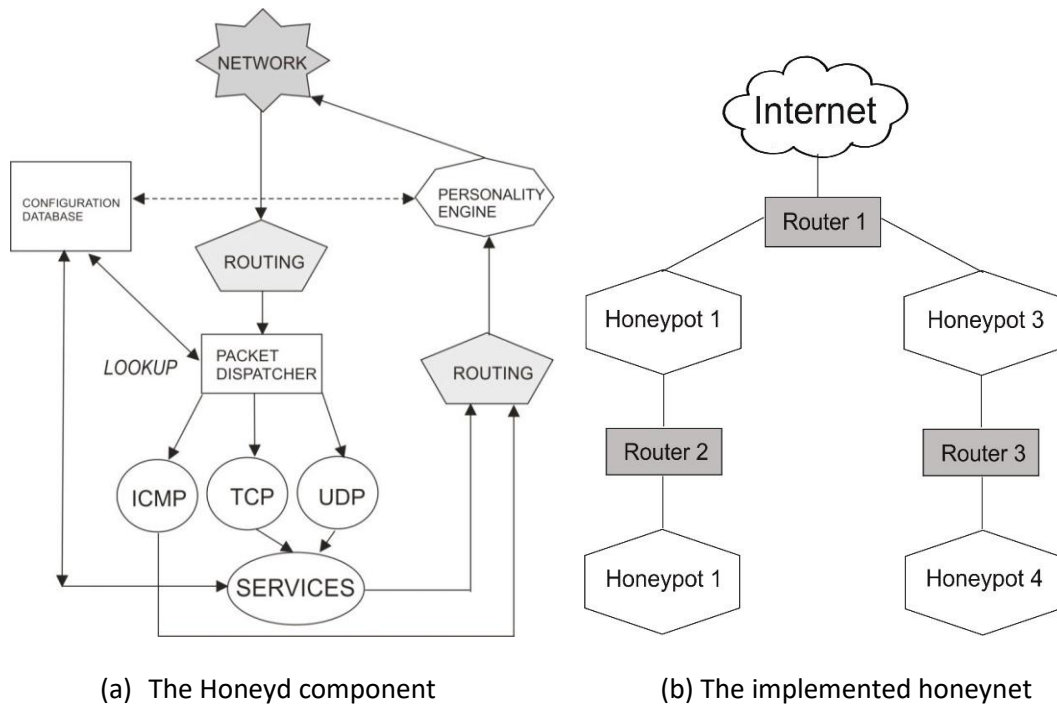


Fig. 1. (a) Architectural sketch of the Honeyd components and (b) the scheme of the implemented honeynet

```
# Windows 2000 Server SP3 WebServer
create windows2000 set windows2000 personality "Microsoft Windows
2000 Server SP3"
add windows2000 tcp port 80 "perl scripts/iis-0.95/iisemul8.pl"
add windows2000 tcp port 139 open
add windows2000 tcp port 137 open
add windows2000 udp port 137 open
add windows2000 udp port 135 open
set windows2000 default tcp action reset
set windows2000 default udp action reset
```

To improve the reality of the implemented honeynet, Honeyd allows to simulate all the standard devices connected to a network, such as “Cisco” routers as shown in the following example:

```
# Cisco Router  
create routerCisco  
set routerCisco personality "Cisco IOS 11.3 - 12.0(11)"  
set routerCisco default tcp action reset  
set routerCisco default udp action reset  
add routerCisco tcp port 23 "/usr/bin/perl scripts/router-telnet.pl"  
set routerCisco uid 32767 gid 32767  
set routerCisco uptime 1327650
```

All configurations are contained in a simple text file (name.conf) that must be read by the program, and according to which all details of the simulated network are created. This is in fact a sort of false digital profile offered to the attackers, to increase the realism of the honeypots. This concept is similar to the one of the false digital alibi in which it is shown how simple is to setup false digital evidence on different systems (such as Mac OS X [26], Android devices [27], and different flavor of the Microsoft Windows Oses [28], [29], [30]) in order to claim a false alibi to be used in several scenarios. In the case of the honeynet, since the attackers often try to remotely fingerprint Oses by using tools like nmap or X probe, Honeyd takes the same fingerprint database used by nmap to spoof the response of any OS it is emulating by providing false evidence about the running OS.

In order to present simple but effective experimental results, we focus our attention on the study of the behavior of malicious attacks performed against the SSH service. Also, the experiments let to inspect the activities performed by the attackers once they gain access to the system and try to progress in their intrusion [31], configuring the machine to record the password along with the account name that was used in the login attempt [32], [33]. In order to better analyze the behavior of the attacks, we implemented two identical honeypots, into two different subnets, with two different SSH user account configurations, in order to obtain two different profiles of the same attack to compare. In the first one (see Figure 2(a)) there exist 8 user accounts and their relative's passwords composed by very common words, in order to offer a high level of vulnerability. On the contrary, the second one ((see Figure 2(b))) also contains the same 8 user accounts, but with 8 complex passwords, composed by letters, digits, and special symbols, to resemble to a more protected system.

Here we present the results of the analysis of captured data in the two honeypots during 30 days, focusing in particular on the log files containing the authentication requests to the SSH server: date, time, the IP

address from which the login attempt originated, the result of the request (failure or success), the account name and the password used for the authentication request as follows

Jan 16 03:36:45 basta sshd[2308]: PW-ATTEMPT: 1234

Jan 16 03:36:45 basta sshd[2308]: Failed password for root from 10.0.160.14 port 39529 ssh2

Jan 16 03:17:11 basta sshd[2310]: Illegal user password from 10.0.160.14

Jan 16 03:17:11 basta sshd[2308]: PW-ATTEMPT: password

Jan 16 03:17:11 basta sshd[2308]: Failed password for illegal user password from 10.0.160.14 port 40444 ssh2

Honeypot1	
<i>Account Name</i>	<i>Password</i>
root	root
admin	1234
user	0000
guest	Password
password	123456
test	qwerty
administrator	654321
webmaster	abc123

(a) Weak account names and passwords

Honeypot2	
<i>Account Name</i>	<i>Password</i>
root	JotCR4E->
admin	mC3bum@:
user	ZR?s25{ _
guest	k6r@bPr6
password	Ea~K^#_
test	{Q};Dced
administrator	:3h!t>VD
webmaster	c)isWAr?

(b) Weak account names with strong passwords

b. Experimental Results: Statistical Aspects and Analysis

In this section are presented the results of our experiments that start with a statistical overview of the activities observed on the two honeypots continuing with the analysis of the activities performed after the intrusions.

In the examined period of 30 days, the two honeypots were contacted by 237 different IP addresses. They recorded 74201 login attempts on SSH, capturing in total 2548 different account names and 4231 passwords. We processed the raw data in order to use it as filters to extract relevant information. Such data range from usernames and passwords, the attack types and also the activities performed after the intrusion. As stated above, honeypot1 contained weak accounts names and passwords, while honeypot2 contained weak accounts names but complex passwords. Referring to the SSH login account of the honeypot1, the first success occurred with the same username and password: “root”, after only 23 attempts by only one attack. Thus the remaining 7 accounts were all detected and used to access the machine after about 50-100 attempts. In relation to honeypot2, only one account was successfully detected after 4452 attempts, the one with username “root” and password JotCR4E->.

Regarding the date and time of the connections, considering the database of the 74201 login attempts on SSH, filtering them by date, we analyzed the distribution of the attacks in the 30 days (see Fig. 3), in which we observed that honeypot1 was hit with 57457 login attempts with a rate of 1915 attempt per day (with an increasing trend), while for honeypot2 there were 16744 login attempts with a rate of 558 attempt per day (showing an initial increase, followed by a decrease, probably due to the complexity of passwords).

Analyzing duration and frequency of the attempts, we can split them into two separate groups. The first one comprises attacks performed without interruptions for a period of time (days), with an high frequency and the same interval of time among them. In addition, the second one is composed by attacks realized from time to time, with a low frequency and different intervals of time among them.

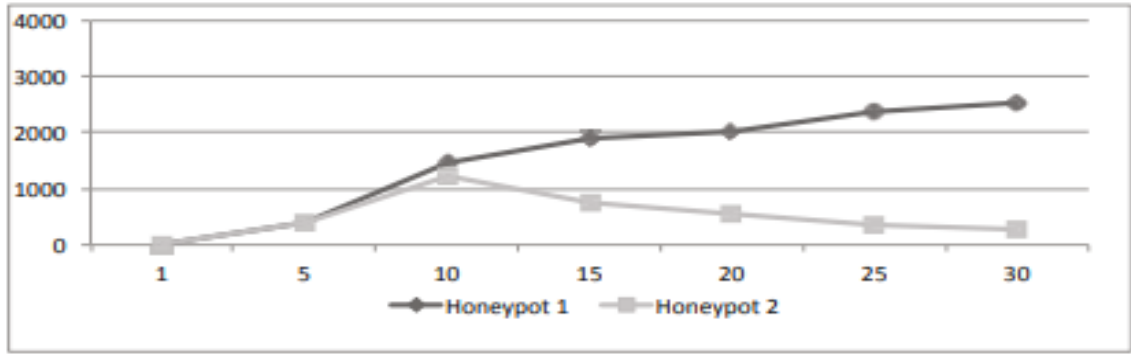


Fig. 3. Trend of access attempts on the two honeypots along the 30 days timeline

Regarding the location (i.e., the IP address) of the attacker, during the examined period of 30 days, the two honeypots were contacted by 437 different IP addresses. All of them attacked the honeypot1, but only 76 attempted to access the honeypot2. Using the tool GeoIP, we could geographically locate the machines performing the attacks, not necessary the real origin of them. We realized that the intrusion tentative come from several countries, that is 21% from USA, 19% from China, 15% from Netherlands 11% from Romania, 9% from the United Kingdom, 7% from Germany, and so on.

Analyzing the total number of login attempts, we recognized from one side a 9% of real-time intrusions, recognized by their behavior, processing username and password in a slow way with different breaks, containing also some typing errors, while on the other site, the 91% were performed by dictionary attacks. Applying “IP addresses” and “honeypot1” as filters to the list of dictionary attacks, we found that 106 of their attempts had these characteristics, which let us to recognize them as performed by automatic scripts. In fact, such connections were only targeted against port 22, thousands of usernames and passwords were processed in a very short time, no pauses were found between attempts and weak usernames and passwords were found in a very short time. The login attempts against the honeypot2 were performed from 76 IP addresses, which used 233 dictionary attacks, 17 real-time intrusions, and 52 scanning activities. Referring to honeypot1, only 8 real-time attacks, performed by two different IP addresses, were able to compromise the system, while all the 106 dictionary attacks violated the machine.

c. Analysis of the Activities Performed after the Intrusions

After a successful entry, a series of tasks were performed by the attackers on the abuser. Attackers first change the password for the hacked account and then try to get root privileges. After that, start scanning the file system and start downloading files using wget and sftp commands. Also, create and hide new clues where you can store malicious software that is often used to scan networks and build backgrounds. Often, such software enables an IRC client to join botnets and other useful tools to perform multiple scanning tasks

d. Sample code with php

```
<?php
//check if form was sent
if($_POST){
    $to = 'some@email.com';
    $subject = 'Testing HoneyPot';
    $header = "From: $name <$name>";

    $name = $_POST['name'];
    $email = $_POST['email'];
    $message = $_POST['message'];

    //honey pot field
    $honeypot = $_POST['firstname'];

    //check if the honeypot field is filled out.
    If not, send a mail.
    if( ! empty( $honeypot ) ){
        return; //you may add code here to echo
an error etc.
    }else{
        mail( $to, $subject, $message, $header
);
    }
}
```

```

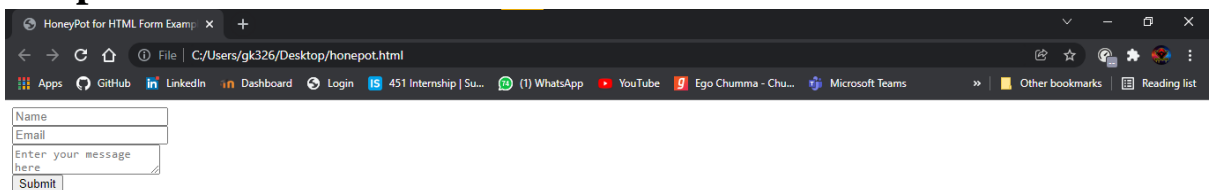
?>
<html>
  <head>
    <title>HoneyPot for HTML Form
Example</title>
    <style>
      .hide-robot{
        display:none;
      }
    </style>
  </head>
  <body>
    <form method="post" action="#my-form"
id="my-form">
      <!-- Create fields for the honeypot
-->
      <input name="firstname" type="text"
id="firstname" class="hide-robot">
      <!-- honeypot fields end -->

      <input name="name" type="text"
id="name" placeholder="Name" required><br>
      <input name="email" type="email"
id="email" placeholder="Email" required><br>
      <textarea name="message"
id="message" placeholder="Enter your message
here" required></textarea><br>
      <input type="submit">
    </form>

  </body>
</html>

```

Output:



On the Construction of a Profile of Attack

Upon performing the analysis of the attacks, it is possible to extract lot of interesting features that can be useful to start constructing the profile of the attack. First of all, the IP address of the attacker machine and the associated “owner” of such IP together with the geographical location of the attacker machine are extracted. Clearly, the type of attack can be a real-time attack or one based on a dictionary. All the temporal information, such as attack lifetime, duration of the attack with the complete hour and day time are easily calculated since the honeypots have been synchronized with a trusted external source of time. From such temporal data it is possible to have the frequency of the attempts of intrusion giving an idea of its regularity and occurrence. Analyzing whether the attack was successful or not, it can be seen the activities performed after the intrusion that may range from (internal/external) network scan, download of files, system exploration, directory creation, malicious software upload and installation. At last, it can be analyzed the type of installed malicious software, the activities performed by the malicious software and, more importantly, the traces and evidence left on the attacked host.

The information extracted from above-mentioned features allow us to build the profile of the attack. First of all, we recognized two main groups: real-time or automatic attacks. We want here to focus attention on the second group, to which we applied, as filters, the following features: high frequency attacks, fast guessing of usernames and passwords, network scanning, creation of new directories/files, successful passwords guessing, upload of files, no errors encountered, success of the intrusion and no traces left.

a. Profile Analysis to Detect the Presence of a Cyber Threat

The obtained results brought to our attention one intruder, the only one able to compromise the honeypot2. Its IP origin seems to be in Shaoxing, located in the province of Zhejiang, China. Here we show a detailed timeline of its activity:

– Jan 6 2013, 02:00 A.M.: *the attacker machine launched a dictionary attack against honeypot1, breaking one login account in 21 minutes (username: test password: 0000).*

– Jan 6 2013, 02:22 A.M.: *it changed the password in N!ka@mikk@2112, then it closed the connection.*

- Jan 7 2013, 02:00 A.M.: the same machine entered the system with its new account and began to scan the network finding honeypot2 and its open port 22.*
- Jan 7 2013, 02:07 A.M.: It began a dictionary attack to SSH login account on the honeypot2, finding the username root in about 10 minutes.*
- Jan 7 2013, 02:18 A.M. it continued its attack against the password, processing thousands of words in a very fast way with high frequency attempts, and stopped at 5:00.*
- Jan 8 2013, 01:00 A.M.: it resumed its attack stopping it at 5:00.*

He attacked the system in a continuous way for 11 days, from 0:00 to 6:00 A.M. until the right password JotCR4E-> was guessed. After the intrusion, it did not change the password, but created into the home directory a new directory named “MY OLD DOCS”, in which a file named “11022012doc old.pdf” was uploaded before stopping the connection. We did not detect other any other connection until the end of the experimentation on the two honeypots. Submitting that file (named “11022012doc old.pdf”) to a forensic analysis, we extracted the MD5 (0xD1E7C8A8D857E097EEF8922F41074E80), the SHA1 (0xA1339C48B7D8A9F8C7358DA6C3C620F63BE25A51) and file size (253.952 bytes). This allowed us to discover that it was a known cyber threat, named IXESHE [34], that is a backdoor/Trojan born in China. This malware communicates with remote servers and receives instructions, acting as in a botnet. It may download and run other malware. The Trend Micro reported [35] that such a trojan is often attached to email messages as a simple PDF file, coming from a compromised or spoofed account. Once opened, the PDF either displays a blank or dummy page, but the code inside it starts the malware. Once installed, IXESHE starts communicating with compromised machines hosted on previously infiltrated networks. Such a dangerous backdoor is the trojan horse named IXESHE, hidden into a PDF file with a very common name capable to connect to a remote C&C server [36] to transmit and receive information to be used during future attacks. It is worth to highlight that very often PDF files are used to convey different kind of malware. The reader can find an interesting study on some security issues that can be exploited by means of PDF files in [37].

Although this type of malware is almost sent by email messages as attachment, here we saw that it was uploaded, but not executed, into a directory with a very ordinary name, probably for several reasons. In fact, spreading such a malware in the ordinary way may not be effective because it could be easily detected by antivirus checking emails and attachments. Also, the attacker did not start the malware immediately for several reasons: to observe how long it will

remain undetected, to wait for some user to open such file resulting in the malware installation, or to test a new infection method on some compromised machines to improve it and use at a later time on different targeted systems. Moreover, due to the massive spread of mobile devices and its ubiquitous nature, particular attention should be also paid in protecting such mobile equipment from malware attacks [38], [39].

Conclusions

In this paper, we have introduced the use of the analysis method in monitor cyberspace data streaming. The test was made using the honeynet system. Results found from a 30-day trial using two honeypots, designed to capture SSH entry attempts in two different ways. Let us analyze the effectiveness of the attack, step by step, from login attempts to post-login activities. Clearly, this represents a simple situation that can be extended to further research. However, these tests, albeit small, have allowed us to detect, extract and analyze the effectiveness of a single cyberattack used to destroy well-protected systems, in a very short time with a dictionary-based attack against the SSH service, operating only at night with a growing invisible detection.

It is important to note that, in order to learn the attacks of a pre-determined system, it can also be helpful to use tools and methods that record the entire package of packets to the target system. Either way, that approach will not help you when you need a broader perspective. The information obtained from the implementation of our approach can be used to implement a series of cybercrime protection measures, improve physical protection of sensitive infrastructure, and digital Disability for critical systems and the implementation of new security data protection policies. From a defensive point of view, you may be aware of, in real time, the presence of an upcoming attack the implementation of an effective cyber defense system in a powerful way.

References

1. U.S. Department of Defense: Joint Publication 1-02, Dictionary of Military and Associated Terms, http://www.dtic.mil/doctrine/new_pubs/jp1_02.pdf (November 2010)
2. Fahrenkrug, D.T.: Countering the Offensive Advantage in Cyber-space: An Integrated Defensive Strategy. In: 4th International Conference on Cyber Conflict, NATO CCD COE Publications, Tallinn, pp. 197–207 (2012)
3. Klimburg, A.: National Cyber Security Framework Manual. NATO CCD COE Publications (December 2012), <http://www.ccdcoe.org/369.html>
4. Saalbach, K.: Cyber-war. Methods and Practice, version 6.0 (January 2013), <http://www.dirk-koentopp.com/downloads/saalbach-cyberwar-methods-and-practice.pdf>
5. Colombini, C., Colella, A., Mattiucci, M.: Cyber-war Profiling, a new Method for the Analysis of a Cyber-Conflict. To appear on NATO CCD COE, Tallinn (January 2013)
6. Palmieri, F., Fiore, U.: Containing large-scale worm spreading in the Internet by cooperative distribution of traffic filtering policies. *Computers & Security* 27(1-2), 48–62 (2008)
7. Palmieri, F., Fiore, U., Castiglione, A.: Automatic security assessment for next generation wireless mobile networks. *Mobile Information Systems* 7(3), 217–239 (2011)
8. Palmieri, F., Fiore, U.: Audit-Based Access Control in Nomadic Wireless Environments. In: Gavrilova, M., Gervasi, O., Kumar, V., Tan, C.J.K., Taniar, D., Lagan´a, A., Mun, Y., Choo, H. (eds.) ICCSA 2006. LNCS, vol. 3982, pp. 537–545. Springer, Heidelberg (2006)
9. Palmieri, F., Fiore, U.: Network anomaly detection through nonlinear analysis. *Computers & Security* 29(7), 737–755 (2010)
10. Fiore, U., Palmieri, F., Castiglione, A., De Santis, A.: Network anomaly detection with the restricted Boltzmann machine. *Neurocomputing* (2013), <http://dx.doi.org/10.1016/j.neucom.2012.11.050>, doi:10.1016/j.neucom.2012.11.050
11. Vidulich, M., Dominguez, C., Vogel, E., McMillian, G.: Situation Awareness: Papers and Annotated Bibliography, U.S. Department of Defense,

Defense Technical Information Center (DTIC) (June 1994),
<http://www.dtic.mil/dtic/tr/fulltext/u2/a284752.pdf>

12. Colombini, C.M., Colella, A.: Digital Profiling: A Computer Forensics Approach. In: Tjoa, A.M., Quirchmayr, G., You, I., Xu, L. (eds.) ARES 2011. LNCS, vol. 6908, pp. 330–343. Springer, Heidelberg (2011)

13. Colombini, C., Colella, A., Castiglione, A., Scognamiglio, V.: The Digital Profiling Techniques Applied to the Analysis of a GPS Navigation Device. In: 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 591–596 (2012)

14. Castiglione, A., De Santis, A., Fiore, U., Palmieri, F.: Device Tracking in Private Networks via NAPT Log Analysis. In: 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 603–608 (2012)

15. Colombini, C.M., Colella, A., Mattiucci, M., Castiglione, A.: Network Profiling: Content Analysis of Users Behavior in Digital Communication Channel. In: Quirchmayr, G., Basl, J., You, I., Xu, L., Weippl, E. (eds.) CD-ARES 2012. LNCS, vol. 7465, pp. 416–429. Springer, Heidelberg (2012)

16. Matrosov, A., Rodionov, E., Harley, D., Malcho, J.: Stuxnet Under the Microscope, rev. 1.31, ESET LLC (2012), [http://ece.wpi.edu/dchasaki/papers/Stuxnet Under the Microscope.pdf](http://ece.wpi.edu/dchasaki/papers/Stuxnet%20Under%20the%20Microscope.pdf)

17. Castiglione, A., De Prisco, R., De Santis, A., Fiore, U., Palmieri, F.: A botnetbased command and control approach relying on swarm intelligence. *Journal of Network and Computer Applications* (2013), <http://dx.doi.org/10.1016/j.jnca.2013.05.002>, doi:10.1016/j.jnca.2013.05.002

18. Ziolkowski, K.: Ius ad bellum in Cyberspace - Some Thoughts on the “SchmittCriteria” for Use of Force. In: 4th International Conference on Cyber Conflict, NATO CCD COE Publications, Tallinn, pp. 295–309 (2012)

19. Fanelli, R., Conti, G.: A methodology for cyber operations targeting and control of collateral damage in the context of lawful armed conflict. In: 2012 4th International Conference on Cyber Conflict (CYCON), pp. 1–13 (2012)

20. CrySys Lab: sKyWIper (a.k.a. Flame a.k.a. Flamer): A complex malware for targeted attacks (May 2012), <http://www.crysys.hu/skywiper/skywiper.pdf>

21. Bencsáth, B., Pék, G., Buttyán, L., Félégyházi, M.: Duqu: A Stuxnet-like malware found in the wild (October 2011), <http://www.crysys.hu/publications/files/bencsathPBF11duqu.pdf>

22. Kaspersky Lab, Global Research and Analysis Team: Gauss: Abnormal Distribution (August 2012), <http://www.securelist.com/en/analysis/204792238/>
23. Kaspersky Lab, Global Research and Analysis Team: The Mahdi Campaign (July 2012), <http://www.securelist.com/en/blog/208193691/> The Madi Campaign Part II
24. Infosec Institute: Honeypots Resources (October 2012), <http://resources.infosecinstitute.com/honeypots/>
25. Moore, J.: Mercury Live DVD (2013), <http://mercurylivedvd.sourceforge.net/>
26. Castiglione, A., Cattaneo, G., De Prisco, R., De Santis, A., Yim, K.: How to Forge a Digital Alibi on Mac OS X. In: Quirchmayr, G., Basl, J., You, I., Xu, L., Weippl, E. (eds.) CD-ARES 2012. LNCS, vol. 7465, pp. 430–444. Springer, Heidelberg (2012)
27. Albano, P., Castiglione, A., Cattaneo, G., De Maio, G., De Santis, A.: On the Construction of a False Digital Alibi on the Android OS. In: Xhafa, F., Barolli, L., K^oppen, M. (eds.) INCoS, pp. 685–690. IEEE (2011)
28. Castiglione, A., Cattaneo, G., De Maio, G., De Santis, A.: Automated Production of Predetermined Digital Evidence. IEEE Access 1, 216–231 (2013)
29. De Santis, A., Castiglione, A., Cattaneo, G., De Maio, G., Ianulardo, M.: Automated Construction of a False Digital Alibi. In: Tjoa, A.M., Quirchmayr, G., You, I., Xu, L. (eds.) ARES 2011. LNCS, vol. 6908, pp. 359–373. Springer, Heidelberg (2011)
30. Castiglione, A., Cattaneo, G., De Maio, G., De Santis, A., Costabile, G., Epifani, M.: The Forensic Analysis of a False Digital Alibi. In: 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 114–121 (2012)
31. Nicomette, V., Ka[^]aniche, M., Alata, E., Herrb, M.: Set-up and deployment of a high-interaction honeypot: experiment and lessons learned. Journal in Computer Virology 7(2), 143–157 (2011)
32. Li, C., Parsioan, T.: Profiling Honeynet Attackers. In: Proceedings of the Class of 2006 Senior Conference, pp. 19–26 (2005)
33. Seifert, C.: Analyzing Malicious SSH Login Attempts (November 2010), <http://www.symantec.com/connect/articles/analyzing-malicious-ssh-login-attempts>

34. Threat Expert Ltd.: Backdoor:Win32/Ixeshe.E (2013), <http://www.threatexpert.com/report.aspx?md5=d1e7c8a8d857e097eef8922f41074e80>
35. Sancho, D., dela Torre, J., Bakuei, M., Villeneuve, N., McArdle, R.: IXESHE An APT Campaign (2012), http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp_ixeshe.pdf
36. Tyugu, E.: Command and control of cyber weapons. In: 2012 4th International Conference on Cyber Conflict (CYCON), pp. 1–11 (2012)
37. Castiglione, A., De Santis, A., Soriente, C.: Security and privacy issues in the Portable Document Format. *Journal of Systems and Software* 83(10), 1813–1822 (2010)
38. Armando, A., Merlo, A., Migliardi, M., Verderame, L.: Would You Mind Forking This Process? A Denial of Service Attack on Android (and Some Countermeasures). In: Gritzalis, D., Furnell, S., Theoharidou, M. (eds.) *SEC 2012. IFIP AICT*, vol. 376, pp. 13–24. Springer, Heidelberg (2012)
39. Armando, A., Merlo, A., Migliardi, M., Verderame, L.: Breaking and fixing the Android Launching Flow. *Computers & Security* (2013)
40. Castiglione, A., Cattaneo, G., De Maio, G., De Santis, A.: Forensically-Sound Methods to Collect Live Network Evidence. In: 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), pp. 405–412 (2013)