

Identifying and Analyzing Emerging Cyber Security Risks

Ayushi Thakur Chandel

Assistant professor, Computer Science and Technology Department JB institute of Technology, Dehradun, India
248007

DOI: <https://doi.org/10.51583/IJLTEMAS.2025.1407000062>

Abstract: The exponential growth of the Internet interconnections has led to a significant growth of cyber-attack incidents often with disastrous and grievous consequences. Malware is the primary choice of weapon to carry out malicious intents in the cyberspace, either by exploitation into existing vulnerabilities or utilization of unique characteristics of emerging technologies. The development of more innovative and effective malware defense mechanisms has been regarded as an urgent requirement in the cyber security community. To assist in achieving this goal, we first present an overview of the most exploited vulnerabilities in existing hardware, software, and network layers. This is followed by critiques of existing state-of-the-art mitigation techniques as why they do or don't work. We then discuss new attack patterns in emerging technologies such as social media, cloud computing, smartphone technology, and critical infrastructure. Finally, we describe our speculative observations on future research directions.

Keywords: Cyber security, intrusion detection, deep learning, machine learning.

I. Introduction

Our society, economy, and essential infrastructures have increasingly relied on computer networks and information technology solutions. As our reliance on information technology grows, cyber-attacks become more appealing and potentially more catastrophic. According to the Symantec cybercrime report released in April 2012 [1], the annual cost of cyber-attacks amounts to US\$114 billion. When factoring in the time lost by companies attempting to recover from these attacks, the total financial impact of cyber-attacks could soar to an astonishing US\$385 billion [2]. The number of individuals affected by cyber-attacks is also on the rise. A survey conducted by Symantec, which included interviews with 20,000 individuals across 24 countries, revealed that 69% of respondents reported experiencing a cyber-attack at some point in their lives. Symantec estimated that 14 adults fall victim to a cyber-attack every second, translating to over one million attacks each day [3].

What causes the proliferation of cyber-attacks? The answer lies in the fact that cyber-attacks are more cost-effective, convenient, and less hazardous compared to physical attacks [4]. Cyber criminals need only minimal expenses, primarily a computer and an Internet connection. They operate without geographical limitations and are not hindered by distance. Their anonymous presence on the Internet makes them challenging to identify and prosecute. Considering the allure of targeting information technology systems, it is anticipated that both the frequency and complexity of cyber-attacks will continue to increase.

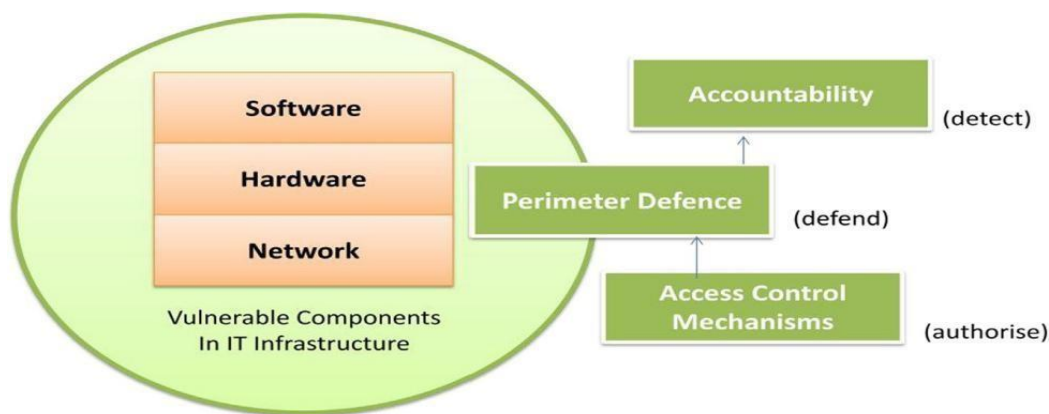


Fig.1. Vulnerabilities and defense strategies in existing systems.

Concerns regarding cyber security involve comprehending the various issues associated with different cyber-attacks and formulating defense strategies (i.e., countermeasures) that maintain the confidentiality, integrity, and availability of all digital and information technologies [18].

- Confidentiality refers to the measures taken to prevent unauthorized individuals or systems from accessing information.
- Integrity pertains to the safeguards implemented to prevent unauthorized modifications or deletions.
- Availability ensures that the systems responsible for delivering, storing, and processing information are accessible when required and by the appropriate users.

Numerous cyber security professionals assert that malware serves as the primary weapon for executing malicious intentions aimed at undermining cyber security measures in cyberspace [12]. Malware encompasses a wide range of attacks that are

installed on a system, usually without the awareness of the rightful owner, to compromise the system for the advantage of an adversary. Notable categories of malware include viruses, worms, Trojan horses, spyware, and bot executable [15]. Malware infiltrates systems through various methods, such as spreading from infected machines, deceiving users into opening compromised files, or enticing users to visit websites that propagate malware. In specific instances of malware

infection, it may transfer itself onto a USB drive connected to an infected device and subsequently infect every other system that the device is later connected to. Malware can also spread from devices and equipment that incorporate embedded systems and computational logic. In summary, malware can be introduced at any stage of the system life cycle. Victims of malware can include a wide array of targets, from end-user systems and servers to network devices (e.g., routers, switches, etc.) and process control systems like Supervisory Control and Data Acquisition (SCADA). The rapid proliferation and increasing sophistication of the ever-growing number of malware pose a significant threat on the Internet today.

Historically, malware attacks occurred at a singular point of vulnerability within hardware devices, software applications, or at the network level, taking advantage of existing design and implementation flaws at each layer. Instead of securing each individual asset, the perimeter defense strategy has primarily been employed to create a barrier around all internal resources, protecting everything within from any unauthorized access from the outside. Most perimeter defense mechanisms rely on firewalls and antivirus software integrated within intrusion prevention and detection systems. Any external traffic is intercepted and scrutinized to ensure that no malware infiltrates the internal resources. The widespread acceptance of this perimeter defense model can be attributed to its relative ease and perceived cost-effectiveness in securing a single perimeter compared to the challenge of protecting a multitude of applications or numerous internal networks. To provide more specific access to certain internal resources, access control mechanisms have been implemented alongside the perimeter defense strategy. In addition to perimeter defense and access control, accountability measures are introduced to identify and penalize any misconduct, as illustrated in Fig. 1. Nevertheless, the combined efforts of the perimeter defense strategy have proven to be increasingly ineffective as malware continues to advance and become more sophisticated. Constantly evolving malware appears to exploit vulnerabilities that allow it to circumvent perimeter defenses entirely. We will detail the most prevalent exploitations across the three distinct layers of the existing information system: hardware, software, and network layers. Following this, we will evaluate the advantages and disadvantages of the most representative defense mechanisms employed within these layers.

Historically, malware attacks occurred at a singular point of vulnerability within hardware devices, software applications, or at the network level, taking advantage of existing design and implementation flaws at each layer. Instead of securing each individual asset, the perimeter defense strategy has primarily been employed to create a barrier around all internal resources, protecting everything within from any unauthorized access from the outside. Most perimeter defense mechanisms rely on firewalls and antivirus software integrated within intrusion prevention and detection systems. Any external traffic is intercepted and scrutinized to ensure that no malware infiltrates the internal resources. The widespread acceptance of this perimeter defense model can be attributed to its relative ease and perceived cost-effectiveness in securing a single perimeter compared to the challenge of protecting a multitude of applications or numerous internal networks. To provide more specific access to certain internal resources, access control mechanisms have been implemented alongside the perimeter defense strategy. In addition to perimeter defense and access control, accountability measures are introduced to identify and penalize any misconduct, as illustrated in Fig. 1. Nevertheless, the combined efforts of the perimeter defense strategy have proven to be increasingly ineffective as malware continues to advance and become more sophisticated. Constantly evolving malware appears to exploit vulnerabilities that allow it to circumvent perimeter defenses entirely. We will detail the most prevalent exploitations across the three distinct layers of the existing information system: hardware, software, and network layers. Following this, we will evaluate the advantages and disadvantages of the most representative defense mechanisms employed within these layers.

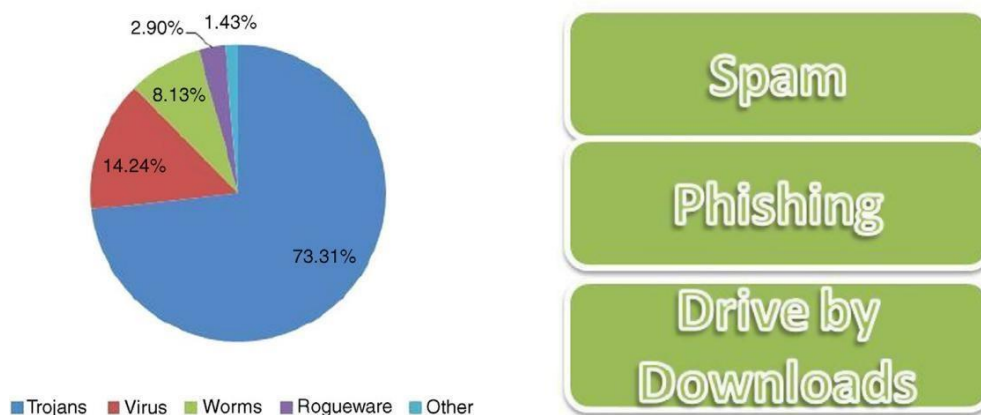


Fig. 2. Types of malware and mediums to spread them [101].

The utilization of computer resources, such as utilities, allows users to pay solely for their usage without incurring any initial costs or requiring expertise in managing intricate computing infrastructures. The increasing accumulation of data within cloud storage services is now drawing the attention of cyber attackers. In June 2012, a Distributed Denial of Service (DDoS) mitigation service on Cloud Flare was compromised by attackers exploiting vulnerabilities in AT&T's voicemail service for mobile users, as well as Google's account-recovery service for Gmail users [19]. With the anticipated rise of 2 billion smartphone users by 2015, there has been a notable surge in mobile malware in recent times. For instance, the global number of unique malware detections for Android rose 17-fold in 2012 compared to the previous year [107]. Additionally, there are increasing concerns regarding cyber threats to critical infrastructure, such as electricity grids and healthcare systems, which could be exploited for terrorism, sabotage, and information warfare. In addition to examining exploitations through the unique characteristics of selected emerging technologies, we also analyze general malware attack patterns that manifest within them to comprehend the methodologies and trends of new attacks.

Ultimately, we present our speculative insights regarding the future directions of research. These insights encompass: (1) privacy concerns aimed at protecting the growing amounts of personal information shared on the Internet, (2) the necessity for a new generation of secure Internet developed from the ground up, taking into account the observed growth and usage patterns that were not considered in the current Internet, (3) the establishment of trustworthy systems with fundamentally different architectures from their inception to resist the continuously evolving malware, (4) the capability to identify and trace the origins of attacks, supported by the advancement of global identity management systems and trace back techniques, and (5) a strong focus on usable security to provide individuals with security controls that they can comprehend and manage. The remainder of the article is organized as follows. Section 2 provides an insight of the malware. Section 3 provides an overview on how malware penetrates in exiting systems and efforts to mitigate any existing vulnerabilities exploited by adversaries. Section 4 reviews emerging approaches to malware infiltration and discusses the general attack patterns and methods. Section 5 discusses future research directions we identified; this will be followed by concluding remarks in Section 6.

Malware as attack tool

In the early days, malware was primarily developed as experiments, often intended to demonstrate security vulnerabilities or, in some instances, to showcase technical skills. In contrast, contemporary malware is predominantly utilized to illicitly acquire sensitive personal, financial, or business information for the advantage of others [129,131]. For instance, malware frequently targets government or corporate websites to extract confidential information or to disrupt their operations. Additionally, malware is also employed against individuals to obtain personal data such as social security numbers or credit card details. With the advent of affordable and faster broadband Internet access, malware has increasingly been crafted not only for the stealthy acquisition of information but also explicitly for profit-driven motives [130]. For example, a significant portion of prevalent malware has been engineered to seize control of users' computers for black market exploitation, including sending email spam or monitoring users' web browsing activities to display unsolicited advertisements. According to a report by the Anti-Phishing group [101], there were a total of 26 million new malware instances reported in 2012. Figure 2 illustrates the relative proportions of the various types of new malware samples identified in the latter half of 2012, as reported by the Anti-Phishing group.

This report indicates that Trojans have consistently constituted the largest portion of malware threats, with a significant increase in their prevalence. In 2009, Trojans made up 60 percent of all malware. By 2011, this percentage had risen to 73 percent. Current statistics reveal that almost three out of four new malware variants developed in 2011 were Trojans, suggesting that they have emerged as the favored instrument for cybercriminals to carry out network intrusions and steal data.

	Hardware	Software	Network
Common attacks	- Hardware Trojan - Illegal clones - Side channel attacks (i.e. snooping hardware signals)	- Software programming bugs (e.g. memory management, user input validation, race conditions, user access privileges, etc.) - Software design bugs - Deployment errors	- Networking protocol attacks - Network monitoring and sniffing
Examples of countermeasures	- Tamper-Resistant Hardware (e.g. TPM) - Trusted Computing Base (TCB) - Hardware watermarking - Hardware obfuscation	- Secure coding practice (e.g. type checking, runtime error, program transformation, etc.) - Code obfuscation - Secure design and development - Formal methods	- Firewall - Intrusion prevention and detection - Virtual Private Network (VPN) - Encryption

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

Malware creators employ various intermediaries to disseminate malware, aiming to compromise a victim's system. Historically, spam, phishing, and web downloads have been the most prevalent channels for this purpose.

Spam is defined as the distribution of irrelevant, inappropriate, and unsolicited messages to thousands or even millions of recipients. It has emerged as a highly lucrative market since spam can be sent anonymously, with minimal costs incurred beyond managing mailing lists.

This low barrier to entry has resulted in a proliferation of spammers, leading to a significant increase in unsolicited emails. In 2011, the estimated volume of spam messages reached approximately seven trillion. This figure accounts for the costs associated with lost productivity, fraud, and the additional resources required managing the spam influx. Currently, the most recognized form of spam is email spam. According to the Message Anti- Abuse Working Group report, between 88% and 92% of email messages sent in the first half of 2010 were identified as spam.

Phishing is a method used to obtain sensitive information, such as usernames, passwords, or credit card details, by pretending to be a trustworthy entity. Most phishing scams aim to trick users into visiting a malicious website that claims to represent legitimate businesses or agencies. An unsuspecting user may enter private information on this malicious site, which is then exploited by cybercriminals. Many phishing techniques employ some form of technical deception to make a link in an email (and the spoofed website) appears to belong to a legitimate organization, such as a well-known bank. Common tactics used by phishers include misspelled URLs or the use of sub-domains. The Anti-Phishing technical report indicated a noticeable trend in 2011 where phishers sought to conceal their intentions by avoiding the use of obvious IP addresses to host their fraudulent login pages. Instead, they preferred to utilize compromised domains to evade detection. It was reported that there was a 16 percent decrease 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 publish.

Drive-by Downloads refer to the unintentional acquisition of malware from the Internet and have increasingly been utilized by attackers to disseminate malware rapidly. Drive-by downloads can occur in various scenarios; for instance, when a user accesses a website, while reading an email message, or when users interact with a misleading pop-up window. Nevertheless, the most prevalent instances of drive-by downloads occur when users visit websites. A growing number of web pages have been compromised with different forms of malware. According to a survey conducted by Osterman Research [3], 11 million malware variants were identified by 2008, with 90% of these malware originating from concealed downloads on popular and frequently trusted websites. Prior to any download, a user must first navigate to the malicious site. To entice users into visiting a site laden with harmful content, attackers often dispatch spam emails that include links to the site. Once an unsuspecting user visits the malicious website, malware is downloaded and installed on the victim's machine without the user's awareness. For instance, the notorious Storm worm utilizes its own network of numerous infected computers to distribute spam emails containing links to such attack pages [102].

Exploiting existing vulnerabilities

Once malware infiltrates the victim's system, cybercriminals can exploit various existing vulnerabilities within that system to further their illicit activities. We analyze the most frequently targeted vulnerabilities in hardware, software, and network systems. This is succeeded by a discussion on the current initiatives proposed to alleviate the adverse effects resulting from these exploitations. A summary of common attacks across the hardware, software, and network layers is provided, accompanied by examples of countermeasures illustrated in Fig. 3.

Hardware

Although anti-virus scanners are designed to periodically identify malicious attacks, numerous hardware-based attacks possess the capability to evade such detection. Exploiting the lack of support tools for hardware detection, reports indicate a rise in hardware-based attacks [23].

Among the various forms of hardware misuse, hardware Trojans represents the most nefarious and prevalent hardware exploits [24]. These hardware Trojans involve malicious and intentionally covert modifications to electronic devices, such as Integrity Circuits (IC) within the hardware [25]. Hardware Trojans can manifest in various degrees, leading to different types of adverse effects. For instance, a hardware Trojan may manipulate an error detection module to accept inputs that should otherwise be rejected. Additionally, a Trojan could introduce extra buffers in the chip's interconnections, resulting in increased power consumption, which may subsequently deplete the battery rapidly. In more severe instances, Denial-of-Service (DoS) Trojans can obstruct the operation of a function or resource. A DoS Trojan has the potential to deplete limited resources such as bandwidth, computational power, and battery life. Furthermore, it could physically damage, disable, or modify the device's configuration, such as causing the processor to disregard interrupts from specific peripherals.

The existence of illegal hardware clones serves as a source of hardware-based exploitation, as the likelihood of counterfeit hardware containing malicious backdoors or hardware Trojans escalates. The probability of producing inauthentic hardware has surged due to a new trend among IT companies aiming to cut costs by outsourcing and purchasing untrusted hardware from online platforms. Karri et al. [26] 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 [22] that examine a number of ways cryptographic algorithm's secret key leaked as a result of analyzing radio frequency.

Several methods have been suggested to counteract attacks at the hardware level. Tamper-resistant hardware devices have emerged as a significant factor due to their importance as a potential entry point for overall system security. The Trusted Platform Module (TPM) offers cryptographic primitives and secures storage, along with the capability to exchange tamper-resistant evidence with remote servers [29–31, 28]. The concept of Trusted Computing Base (TCB) is defined to encompass the components of a system, including all hardware and software elements that are essential to the system's overall security. It is crucial that the TCB is free from any bugs or vulnerabilities, as these could compromise the security of the entire system. A thorough and meticulous review of its code base is performed through computer-assisted software audits or program verification to guarantee the security of the TCB. In hardware watermarking, ownership information is embedded and concealed within the description of a circuit, thereby preventing the host object from being illegally counterfeited. Hardware Obfuscation is a method used to alter the description or structure of electronic hardware to deliberately hide its functionality [22]. These techniques are employed to deter adversaries from accessing the original design or counterfeiting critical components of the hardware, such as integrated circuit (IC) units. Some countermeasures against side-channel attacks include introducing noise to obscure physical information, filtering certain aspects of physical data, and employing blinding techniques to eliminate any correlation between input data and side-channel emissions [23, 24].

Software defects

A software bug refers to a common term that denotes an error, flaw, mistake, or fault within a computer program, which may include internal operating systems, external I/O interface drivers, and applications [103]. Cyber-attacks take advantage of software bugs to manipulate systems in unintended ways that deviate from their original purpose. A significant number of cyber-attacks today continue to arise from the exploitation of software vulnerabilities stemming from software bugs and design flaws [104].

Software-based exploitation occurs when specific features of the software stack and interface are manipulated. The most prevalent software vulnerabilities typically result from the exploitation of software bugs related to memory, user input validation, race conditions, and user access privileges [40, 39, 42]. Memory safety violations are executed by attackers to alter the contents of a memory location. A prime example of this technique is buffer overflow. Buffer overflow happens when a program attempts to store more data in a buffer than it was designed to accommodate. Given that buffers are intended to hold a limited amount of data, any excess information can overflow into adjacent buffers, thereby corrupting or overwriting the valid data contained within them. This situation enables attackers to interfere with existing process code. Input validation is the procedure of confirming that the input data adheres to specific rules. Inadequate data validation can result in data corruption, as exemplified by SQL injection. SQL injection is one of the most recognized techniques that exploit a program bug within a website's software. An attacker injects SQL commands through a web form, either to modify the database content or to extract database information, such as credit card details or passwords, for their own use. An adversary exploits a flaw in a process where the output is unexpectedly and critically reliant on the timing of other events. The time of check to time of use represents a bug that arises from changes in a system between the verification of a condition and the application of the results from that verification. This is also referred to as exploiting a 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.

Within the programming community, numerous projects have been launched with the primary aim of enhancing security [36–38]. These initiatives not only focus on rectifying common inherent security flaws but also strive to introduce innovative concepts to foster a secure computing environment. In secure coding practices based on code reviews, software engineers pinpoint prevalent programming mistakes that can lead to software vulnerabilities, establish standardized secure coding guidelines, educate software developers, and elevate the overall practice of secure coding. In contrast, language-based secure coding practices involve the development of techniques that ensure programs adhere to critical security policies. The most commonly employed techniques encompass analysis and transformation. A prominent type of analysis is 'type is checking,' which allows the program to identify any unsafe object types prior to execution. Another recognized form of program transformation involves the incorporation of runtime checks, which instrument the program to prevent any transformations that would violate established policies [42]. Code obfuscation refers to the process of creating source or machine code that is intentionally made difficult for humans to comprehend [43, 44]. Programmers frequently obfuscate code intentionally to hide its purpose or logic, thereby mitigating the risk of reverse engineering. Additionally, a secure design and

development cycle has been proposed in [41, 27], offering a set of design techniques that enable efficient verification of system components to ensure they are free from potential defects stemming from their original design. Although these methods may not be straightforward, formal approaches allow for a thorough exploration of design and the identification of complex security vulnerabilities. Tools [34, 35] and techniques [32, 33] have been created to assist in verifying mission-critical security properties. These resources aid in translating higher-level security objectives into a cohesive framework. These tools and techniques help to translate higher-level security objectives into a collection of atomic properties to be verified.

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 [46, 47]. 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.

The initial network protocol was designed to accommodate a vastly different environment than what we experience today, operating on a significantly smaller scale, and often fails to function correctly in numerous contemporary scenarios. The vulnerabilities present in network protocols become more complex when both system administrators and users possess limited understanding of the networking infrastructure [46, 47]. For instance, system administrators may neglect to implement efficient encryption schemes, fail to apply recommended patches promptly, or overlook the application of security filters or policies.

A prevalent type of network attack takes advantage of the shortcomings inherent in widely utilized network protocols such as Internet Protocol (IP), Transmission Control Protocol (TCP), or Domain Name System (DNS) [14]. The IP serves as the primary protocol at the network layer, supplying the necessary information for routing packets between routers and computers within the network. The original IP protocol lacked any mechanisms to verify the authenticity and confidentiality of the data being transmitted, which permitted the interception or alteration of data while it traversed an unknown network between two devices. To address this issue, IPsec was developed to facilitate the encryption of IP traffic. Over the years, IPsec has become a fundamental technology for establishing a virtual private network (VPN), which creates a secure channel over the Internet between a remote computer and a trusted network (e.g., a company intranet). TCP operates above the IP layer to ensure reliable transmission of packets (i.e., by retransmitting lost packets) and orderly delivery of those packets. SSL was initially created to provide end-to-end security, as opposed to merely a layer-based protocol, between two computers that operate over the transmission control protocol (TCP). SSL/TLS is frequently employed alongside http to create https for secure web pages. The domain name server (DNS) is the protocol responsible for converting human-readable host names into 32-bit Internet protocol (IP) addresses. It essentially functions as a directory for the Internet, directing routers to the appropriate IP address. It 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 serves as a vital instrument for safeguarding the data exchanged between users by encrypting it, ensuring that only authorized users possessing the correct keys can decrypt the information. It is the most prevalent method employed for data protection. A survey carried out by the Computer Security Institute in 2007 [132] indicated that 71% of organizations implemented encryption for their data during transmission. In response to the sophisticated threats posed by attackers who exploit the weaknesses of current cryptographic algorithms, various initiatives are gaining momentum. The US National Institute of Standards and Technology (NIST) have recently declared the discontinuation of SHA-1, advocating for the adoption of the Advanced Hash Standard (ASH) since 2012 [15]. The exploration of identity-based encryption remains a significant area of research for applications necessitating high-speed encryption, thereby circumventing the limitations of the slow 2048-bit RSA key length and the impractical involvement of a trusted certifying authority [15]. Quantum cryptography represents a developing technology wherein two parties concurrently create shared, secret cryptographic key material through the transmission of quantum states of light [48].

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 [53]. The visual presentation of the data is then analyzed by network experts with in-depth domain knowledge in networking system.

II. Discussion

Although numerous distinct techniques and proposals are available to address vulnerabilities across hardware, software, and

network layers, the traditional approach has favored bundled security protection techniques that shield all internal components from external threats. The vast majority of organizations utilize a perimeter defense security model to protect their networks from potential external intrusions [47]. This strategy emphasizes 'layered defense' or 'defense in depth,' wherein critical internal IT assets, such as servers and mission-critical data, are safeguarded by various barriers and fortifications.

Common perimeter defenses include technologies such as firewalls and intrusion detection systems (IDS). The firewall is the most widely used technology for protecting internal assets. Its primary function is to manage both incoming and outgoing network traffic by analyzing data packets and determining whether to allow them based on a predetermined set of rules. A firewall can be deployed at various layers within the network infrastructure. Network layer firewalls, often referred to as packet filters, operate at a relatively low level of the network layer, preventing packets from passing through unless they adhere to the established rule set (i.e., configurations) defined by network administrators. Despite the progress made in modern firewalls, network layer firewalls are incapable of filtering out undesirable traffic, such as malware payloads that exploit legitimate IP addresses and ports. Application layer firewalls function by monitoring and potentially blocking input, output, or system service calls that do not comply with the configured policies of the network layer firewall. A proxy server can act as a firewall by responding to incoming packets (for example, connection requests) in a manner similar to an application while blocking other packets. Both application layer firewalls and proxies make unauthorized access to internal systems more difficult. However, with the... But with the enhanced capabilities and sophistication, attackers today have developed more advanced methods to transmit malicious packets to a target network. For instance, intruders may take control of a publicly accessible system and utilize it as a proxy for their own objectives. By using the compromised proxy, the intruder generates packets with a forged IP address to obscure the sender's identity or to impersonate another computing system.

The intrusion detection systems are designed to filter any suspicious or anomalous activities occurring over the network [19, 47]. These detection systems are valuable as they aim to identify the early stages of an attack (for instance, an attacker probing a machine or network for specific vulnerabilities) and can subsequently assist in safeguarding a machine from the later stages of the attack.

Additionally, these systems strive to recognize telltale signs of suspicious activities or behavioral patterns, whether they originate from a user, an application, or a piece of malicious code that firewalls or other protective tools may overlook or disregard. Numerous variants of detection systems exist to identify malicious network payloads. These detections can be categorized as either signature-based or anomaly-based. In the signature-based approach, the detection system identifies attack packets by their well-known fingerprints or signatures as they traverse the network's gateway threshold. Conversely, in the anomaly-based approach, the detection system lacks prior knowledge of what constitutes bad packets. Instead, it determines what normal traffic is by analyzing patterns, often in real-time, and reports abnormal traffic behaviors based on this pattern analysis.

The signature-based detection system has been deemed ineffective due to the increasing proliferation and sophistication of malware writers in recent years [16, 17]. It is believed that keeping pace with the ever-evolving malware signatures using the pattern recognition methods commonly employed in the signature-based approach is nearly impossible. Consequently, proposing advanced anomaly-based detection methods has become a prominent area of research [133]. In this approach, the system learns by example (self-learning) what constitutes normal behavior by monitoring traffic over an extended period and constructing a model of the underlying process. This process evolves (self-adaptive) as the signatures of malware continue to develop. 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 [49–51]. The evidences are used for legal action or to understand the network traffic attack patterns. E Mail Tracker Pro [5] 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 Smart Who Is [6] 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 [7] assists users in reviewing web site URLs that are stored in the history files. The Index.dat analyzer [8] is a forensic tool to investigate index.dat files to examine the browsing history, the cookies and the cache. WinPcap [9] captures the packets intercepted at the network interface of a system running the Windows Operating System while AirPcap [10] 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 [52]. Any information captured by honeypots is 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 honey net which connects two or more honeypots on a network, is used for monitoring a larger and more diverse network. These honey nets are often implemented as parts of larger network intrusion detection systems. A honey farm is a centralized collection of honeypots and analysis tools [54].

Since providing uniform access to resources is not feasible, access control mechanisms have been implemented to allow an authority to manage access to specific resources. In access control, the entities capable of executing actions within the system are referred to as "subjects," while the entities that represent resources requiring access control are termed "objects." In

capability-based access control, a subject is permitted to access an object if it possesses a reference or capability. For instance, when a user inputs the correct user ID and password, they are authorized to view their bank statement. Access can be transferred to another party by sending such a capability through a secure channel. For example, a certificate is generated for the user to present for verification purposes. In an access control list-based method, a subject's access to an object is contingent upon whether its identity appears on a list associated with that object. For example, if Alice is included on the list of doctors, she is allowed to view patient records. Access is managed by modifying the list. For instance, when Alice departs from the hospital, she is removed from the list of doctors and will no longer have the ability to view patient records. The three most commonly acknowledged models are Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-Based Access Control (RBAC). In DAC, the owner determines who is permitted to access certain objects and what privileges they possess. In the MAC approach, the operating system limits the ability of subjects (e.g., processes or threads) to access or perform operations on objects (e.g., files, directories, TCP/UDP ports, or shared memory segments), either through a rule that specifies particular conditions or through a mathematical structure that defines the greatest lower bound and least upper bound values. RBAC is a more recent alternative to MAC and DAC that restricts system access solely to authorized users. It is utilized by the majority of enterprises, and most IT vendors provide RBAC in one or more products.

Traditional access control systems deliver fundamental services such as authentication, authorization, and accountability. Authentication and authorization involve the process of confirming that a subject is associated with an object. Traditional mechanisms for authentication and authorization utilize three distinct factors to identify a subject and ascertain whether the subject possesses the appropriate capability to access the object. The first factor is something you know, such as a password or a personal identification number (PIN). This presumes that only the account owner is aware of the password or PIN required accessing the account. The second factor is something you have, which may include a smart card or security token. This assumes that only the account owner possesses the necessary smart card or token to unlock the account. The third factor is something you are, which encompasses characteristics like fingerprints, voice, or iris patterns. The prevailing trend in authentication is a layered approach, often referred to as strong authentication, which relies on the presentation of two or more authentication factors [55]. Various types of compact authentication tokens have been proposed. These tokens house a cryptographic key within tamper-resistant storage. Leveraging the ubiquitous nature of modern computers equipped with USB ports, USB-based tokens have also been suggested, either simply as a storage medium for an X.509 certificate or frequently executing a challenge/response protocol [56,13,86]. To capitalize on the rapidly expanding population of mobile users, several authentication mechanisms aimed at mobile users have been proposed [56–58]. Biometric technology has found limited applications. While some PCs and workstations have become increasingly sophisticated with audio-visual interfaces, there is a renewed interest in utilizing biometric authentication technology within the network environment [59, 60, 62]. 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 [20]. There is a variety of technique used by researchers in the accountability; most notable ones are logging, auditing and conflict resolution [21, 94]. 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 [93].

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 included: 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.

Social media

Social media, such as Facebook and Twitter, has shown explosive growth in recent years. At the end of 2012, there are more than 450 million active user accounts in Twitter while the number grows exponentially in Facebook reaching almost

Common characteristics	Common attack patterns
• 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	• 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.

1 billion users [108]. Social networking platforms have gained immense popularity and have become the favored means of communication for the majority of younger generations. Each of these social media sites generally offers tools that allow users to share their personal details (such as name, address, gender, date of birth, and preferences in music and movies), photos, stories, and links. Malicious actors are exploiting the social media phenomenon as a new avenue for launching harmful attacks. By the conclusion of 2008, the Kaspersky Lab database included over 43,000 malicious files associated with social media platforms [109]. A report released by the IT security and data protection company Sophos has indicated a concerning increase in attacks targeting users of social media sites. According to their findings [111], approximately 60% of users on social networks have encountered spam. Given the unrestricted access to user profiles, attackers can also acquire information regarding corporate and commercial secrets. In a survey conducted by Sophos [111], about 60% of companies expressed concern that their employees disclose excessive information on social networks, while around 66% of companies believe that the use of social networks presents a significant threat to their operations. The Koobface worm [110], which propagated through social media platforms in 2009, is particularly recognized as a prominent malware case that capitalized on the widespread use of social media. Utilizing its network of compromised machines, the Koobface botnet automates the creation of new social media accounts to befriend unsuspecting users, subsequently spamming them with enticing links that lead to malware. Victims who fall victim to these social engineering schemes find their own social networking accounts transformed into tools for sending spam to their friends, while their machines are converted into zombies. Thomas and Nicol [110] developed a zombie emulator capable of infiltrating the Koobface botnet, identifying fraudulent and compromised social network accounts that were used to disseminate malicious links to over 213,000 social network users, generating further concern about the security of these platforms.

Social networking platforms have heightened the importance of privacy protection due to the centralization of vast amounts of user data, the personal nature of the information collected, and the continuous availability of updated data that is consistently tagged and formatted [109]. This situation renders social networking sites appealing targets for various organizations aiming to aggregate substantial quantities of user data, some for legitimate reasons and others for malicious intents. In many instances, the extraction of data infringes upon users' expectations of privacy. The safeguarding of users' private data stored by social networking service providers has been a subject of investigation. Lucas et al [112] introduced a Facebook application designed for encrypting and decrypting sensitive data through client-side JavaScript. This framework guarantees that data does not reach social network service providers in an unencrypted state, thereby preventing them from monitoring and accumulating the information that users transmit across the network. Additionally, issues related to privacy awareness and tools that assist users in configuring their privacy settings more intuitively have also been suggested. For instance, Fang and LeFevre [113] proposed a privacy wizard. This wizard iteratively prompts the user to assign privacy 'labels' to chosen friends, utilizing this input to develop a classifier based on a machine learning model, which can subsequently be employed to automatically allocate privileges to the remaining friends of the user. The rationale behind this design stems from the observation that actual users formulate their privacy preferences regarding which friends should have access to specific information, based on an implicit set of rules they establish and consistently apply in most friend settings.

Cloud computing

The efficiencies associated with migrating data and applications to the cloud continue to entice consumers who utilize services such as Drop Box and I Cloud for data storage, employ Gmail and Live Mail for email management, and monitor their activities through platforms like Ever note and Mint.com. Cloud computing is arguably one of the most significant technological transformations in recent history [16]. The very concept of utilizing computing in a manner akin to using a utility is transforming the IT services landscape and presents substantial potential. Clients, whether they are large corporations or small enterprises, are attracted to the cloud's promises of flexibility, lower capital expenditures, and improved IT resources. IT firms are transitioning from offering their own IT infrastructure to leveraging the computational services available through the cloud to meet their information technology requirements [66].

Cloud computing offers distinct features that set it apart from traditional methods. The five primary characteristics of cloud computing are on-demand self-service, ubiquitous network access, location- independent resource pooling, rapid elasticity, and measured service, all designed to facilitate seamless and transparent cloud usage [67]. Resource pooling refers to the capability where resources are not exclusively allocated to a single user but are instead aggregated to serve multiple consumers. Resources,

whether at the application, host, or network levels, are allocated and reallocated as required by these consumers. On-demand self-service allows users to automatically assign themselves additional resources, such as storage or processing power, without the need for human intervention. This concept is similar to autonomic computing, where the computer system is capable of self-management. In addition to self-provisioning of resources, cloud computing is characterized by the ability to quickly locate and release resources, a feature commonly referred to as "elasticity." This enables consumers to increase the resources they require at any moment to handle heavy loads and usage spikes, and subsequently decrease them by returning the resources to the pool when they are no longer needed [66]. Measured service, often referred to as pay-as-you-go, allows the cloud to be utilized as a utility, where users are charged based on their consumption, similar to how they pay for utilities such as electricity, gas, and water.

Cloud computing represents a model of integration that provides various resources to clients across different layers of the system while utilizing diverse resources. In general, the architecture of a cloud computing environment can be categorized into four layers: the hardware layer (which includes data centers), the infrastructure layer, the platform layer, and the application layer [68].

The hardware layer: This layer is tasked with managing the physical resources of the cloud, which encompass physical servers, routers, switches, as well as power and cooling systems. In practice, the hardware layer is usually realized in data centers. A data center typically houses thousands of servers that are organized in racks and interconnected via switches, routers, or other networking fabrics. Common challenges at the hardware layer include hardware configuration, fault tolerance, traffic management, and the management of power and cooling resources.

The infrastructure layer: This layer is also referred to as the virtualization layer. The infrastructure layer generates a pool of storage and computing resources by partitioning the physical resources through virtualization technologies such as Xen, Kernel-based Virtual Machine, and VMware. The infrastructure layer is a crucial element of cloud computing, as many essential features, including dynamic resource allocation, are only accessible 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.

Nonetheless, in practice, cloud services can be categorized into three main types: software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS) [68]. Applications that operate on or are being developed for cloud computing platforms encounter various security and privacy issues, which vary based on the underlying delivery and deployment models. In the case of IaaS, the cloud provider offers a range of virtualized infrastructure components, including virtual machines (VMs) and storage, which customers can utilize to build and run their applications. Ultimately, the application will reside on the VM along with the virtual operating system. PaaS provides programming environments that can access and leverage additional application building blocks. These programming environments significantly influence the application architecture, imposing constraints on the services that the application can request from an operating system. Lastly, in the SaaS model, cloud providers deliver application software as on-demand services. Multi-tenancy is a distinctive feature of cloud computing that enables providers to optimize resource utilization by partitioning a shared, virtualized infrastructure among multiple customers. For instance, to ensure the isolation of data among various tenants, Salesforce.com utilizes a query rewriter at the database level, while Amazon implements hypervisors at the hardware level. Virtualization serves as a crucial enabling technology in this domain, facilitating the abstraction of infrastructure and resources, which can be made available to clients as isolated VMs. Ensuring robust isolation, mediated sharing, and secure communication between VMs remains an active area of research. The implementation of a flexible access control mechanism that regulates the control and sharing capabilities of VMs within a cloud host has been proposed as a potential solution [67]. Given that clients obtain and utilize software components from various providers, significant concerns arise regarding the secure composition of these components and the protection of information managed by these integrated services [67]. For instance, a PaaS environment. Might limit access to well-defined parts of the file system, thus requiring a fine-grained authorization service. The management of trust and the integration of policies represent a significant field of study within cloud computing. This is primarily due to the outsourcing model inherent in cloud services, where providers are responsible for the control and management of users' data and services, necessitating a high level of trust in the technical abilities of these providers [67]. In cloud computing settings, the interactions among various service domains, influenced by service requirements, are characterized as dynamic, transient, and intensive. Consequently, a trust framework has been proposed to effectively capture a comprehensive set of parameters essential for establishing trust, as well as for managing the evolving requirements related to trust and interaction/sharing [115,116]. Additionally, the integration of policies within the cloud is another prominent research area aimed at tackling issues such as semantic heterogeneity, secure interoperability, and the management of policy evolution [117]. Moreover, the behaviors of customers can change rapidly, which in turn can impact the trust values that have been established. This indicates a necessity for a cohesive, trust-oriented, secure interoperation framework that facilitates the establishment, negotiation, and maintenance of trust, thereby adaptively supporting policy integration [114,117].

Smartphones

Smartphones, along with advancements in wireless technologies, have evolved into highly sophisticated devices for computing and communication that individuals carry with them throughout the day. The combination of enhanced computing power, personalization, and mobility renders them an appealing tool for managing both work and personal life. According to [16], the vast number of mobile phone users globally highlights an urgent requirement for proactive mobile security measures. It is estimated that over 4.5 billion people utilize a cell phone daily, with projections indicating that 2 billion smartphones will be in use by 2013. Beyond basic SMS messaging, smartphones now store an increasing amount of sensitive information. In corporate settings, these technologies are instigating significant transformations in the organization of information systems, thereby introducing new risks. As smartphones gather and store more sensitive data, access to this information must be regulated to safeguard user privacy and protect the intellectual property of the organization. The remarkable growth in mobile technology has made it an appealing target for cybercriminals. Security issues in mobile devices differ from traditional security challenges faced in PC and enterprise computing due to their embedded characteristics and distinct operational environments. Milliner [119] identified several features that are unique to mobile computing.

- **Mobility:** This is the most important characteristic of the mobile phones. Since mobile users can take them to anywhere, the chances of getting stolen, lost, or physically tampered increase as compared to stationary devices.
- **Strong Personalization:** As a personal device, mobile devices usually are not shared among multiple users.
- **Strong Connectivity:** Mobile phones are commonly used to connect to other devices over the wireless networks (or wireless Internet) for data exchanges.
- **Technology Convergence:** Today numerous functional features are integrated in the mobile phones, for example gaming, video and data sharing, and Internet browsing.
- **Limited Resources and Reduced Capabilities:** Comparing with stationary devices, mobile devices have four major limitations:

a) Limited battery life, b) limited computing power, c) very small display screen size, and d) very small sized keys for inputs. These limits bring the challenges in building mobile security technology.

There are various styles of attacks designed to exploit the widespread use of mobile computing. Communication-related attacks stem from vulnerabilities in the design and management of mobile communication infrastructure. An attacker may attempt to compromise the encryption of the mobile network. The GSM (Global System for Mobile Communication) network currently employs two types of algorithms known as A5/1 and A5/2, with the latter recognized as being less secure. Since the encryption algorithm was disclosed, it has been demonstrated that breaking the encryption can be accomplished in approximately 6 hours [118]. An attacker might also attempt to intercept Wi-Fi communications to extract information (e.g., username, password). These attacks are not exclusive to smartphones; however, smartphones are particularly susceptible to them, as Wi-Fi is often their sole means of accessing the Internet. Security concerns regarding Bluetooth on mobile devices have been investigated and have revealed several issues. For instance, Cabir is a worm that propagates through Bluetooth connections [120]. This worm seeks out nearby phones that have Bluetooth enabled and in discoverable mode, sending itself to the targeted device. The user must accept the incoming file and install the application. Once installed, the worm infects the device. To mitigate communication-related attacks, the network traffic exchanged by phones can be monitored, such as through surveillance at network routing points or by overseeing the utilization of mobile network protocols.

Another category of attacks arises from the weaknesses present in mobile software applications, particularly those that exploit mobile web browsers. Similar to conventional web browsers, mobile web browsers extend beyond mere web navigation by incorporating widgets and plug-ins, which many attackers utilize as a means to disseminate malware. The act of jail breaking the iPhone was entirely predicated on vulnerabilities within the web browser, specifically a stack-based buffer overflow in a library utilized by the web browser [121]. Vulnerability in the Android web browser was identified in October 2008, taking advantage of an outdated and susceptible library [121]. Malicious attackers frequently target mobile phones as a conduit for spreading malware [106]. Reports on emerging cyber security threats from Georgia Tech [16] and Symantec [17] over the past few years have highlighted the increasing prevalence of malware specifically designed for mobile devices, such as those targeting Google Android-based phones and Apple iPhones. To mitigate the spread of malware, mobile companies provide a centralized public marketplace complemented by an approval process prior to hosting applications. This centralized marketplace facilitates the removal of any applications deemed suspicious before they can be downloaded by users. For instance, Apple employs a vetting process to ensure that all applications adhere to its guidelines before they are made available through the App Store. Apple grants approval for an application through code signing with encryption keys. Accessing applications via the App Store is the sole method for iPhone devices to install software. In a similar vein, Android also maintains a public marketplace for hosting applications. However, in contrast to Apple, Android applications can be self-signed. Android leverages crowdsourcing to evaluate applications based on user ratings. In response to user complaints, applications can be removed from the marketplace and uninstalled from devices as well. Another strategy employed by mobile companies to safeguard their platforms is the concept of sandboxing. Sandboxing compartmentalizes different processes to prevent them from interacting and damaging each other therefore effectively limiting any chance for malicious code to be implanted and overtaking the running processes from doing harmful activities. Apple iOS focuses

on limiting access to its API for applications from the Apple Store while Android uses its sandboxing on underlying legacy Linux kernel.

Critical infrastructure

The essential infrastructure systems that constitute the foundation of contemporary society, along with their dependable and secure functioning, are crucial for national security and economic health. In many respects, the cyber system serves as the backbone of a nation's critical infrastructures, implying that a significant security breach in cyber systems could greatly affect the reliable and safe functioning of the physical systems that depend on it. Recent findings, as outlined in government reports [15], reveal an increasing threat of both physical and cyber-based attacks, both in frequency and complexity, targeting electric grids and other vital infrastructure systems. Cyber security concerning critical infrastructure aims to reduce the vulnerabilities of these structures and systems to [77]:

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.

Protecting critical infrastructure poses greater challenges than safeguarding information and communication technology (ICT) due to the intricate interconnections of these infrastructures, which can result in various issues [78]. Take the power grid as an example, where production sites located in different geographical areas distribute electricity through stations that manage varying voltage levels (from higher to lower) until the energy reaches our homes. Typically, both production and distribution facilities are overseen by supervisory control and data acquisition (SCADA) systems, which maintain remote connections to supervision centers and the corporate networks (intranets) of the companies that operate these infrastructures. The intranets are connected to the Internet to enable communication with power regulators and end users, among other functions. These connections create vulnerabilities that can be exploited by external attackers. Operators' access SCADA systems remotely for maintenance tasks, and in some cases, equipment suppliers maintain connections to these systems via modems. The widespread use of proprietary solutions and older versions that are rife with vulnerabilities further complicates the development of effective strategies to safeguard the nation's essential 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 [79,77]. Another relevant topic of interest is the development of self-healing systems to pursue automated and coordinated attack response and recovery [77].

Other emerging areas of concern

Cyber security concerning embedded systems and sensors has garnered significant attention from both industry and academia in recent years, primarily due to their growing integration into various aspects of our daily lives. For instance, small embedded devices are increasingly found in automobiles, household appliances, mobile phones, and audio/video equipment, becoming integral to our existence. Likewise, sensors are experiencing expanded research and commercial applications in military, scientific, and commercial sectors, including the monitoring of biological habitats, agriculture, and industrial processes. The security challenges in these domains differ from traditional security issues encountered in personal computers and enterprise computing, owing to their unique embedded characteristics and operational environments [72,73]. Embedded systems and sensors are often highly cost-sensitive, necessitating the use of smaller processors that offer limited capacity for security measures, such as storing large cryptographic keys. Consequently, most enterprise security solutions are not applicable in the realm of embedded systems. These systems and sensors face constraints in energy, memory, computational speed, and communication bandwidth due to their compact size. They also possess a notably weak physical trust boundary. For example, they are installed in residential and commercial properties, situated in outdoor fields, or carried by individuals in their hands or pockets, making them susceptible to various physical attacks. Furthermore, they rely on a close integration of hardware and software, frequently lacking the protection of an operating system. The distinct embedded nature of the embedded systems and sensors have created different sets of security vulnerabilities [74, 75]. For example, limited battery power in embedded systems makes them vulnerable to attacks that drain this

resource [73]. The proximity of embedded systems to a potential attacker creates vulnerabilities for attacks where physical access to the system is necessary. This allows the attackers to perform attacks that are involved examining the usage of physical system, for example, power analysis attacks or snooping attacks on the system bus. Embedded systems need to operate within a reasonable environmental condition. Due to the highly exposed operating environment of embedded systems, there is a potential vulnerability for attacks that overheat the system (or cause other environmental damage). Attackers reprogram a stolen embedded system to use them for further misuse. The usual security countermeasures to prevent unauthorized access through user authentication, techniques to preserve data integrity through cryptographies and network defense mechanisms are active area of interest in the field.

However, preventing attacks done by examining or altering the physical system are quite unique, for example techniques such as masking, window methods and dummy instruction insertion in the code/algorithm have been proposed [72]. Since Network connectivity via wireless or wired access is increasingly common for embedded systems to enhance remote control data collection and update, the vulnerabilities that exploiting such network connectivity, such as spread of viruses and wiretapping, have become another source of growing concern in the field.

Cyber warfare pertains to hacking driven by political motives aimed at executing sabotage and espionage. In the book Cyber Ware [92], cyber warfare is characterized as "actions by a nation-state to infiltrate another nation's computers or networks with the intent of inflicting damage or disruption." The primary concerns surrounding cyber warfare revolve around breaches of national security and the sabotage of a nation's critical infrastructure [76]. The former scenario relates to international espionage, where classified information that could compromise national security is unlawfully accessed or modified by unauthorized individuals. The latter scenario pertains to any possible disruption of a nation's critical infrastructure, including systems such as the power grid and transportation networks. In 2008, the Department of Homeland Security conducted a simulated exercise known as "cyber storm." The objective of this exercise was to evaluate the nation's defenses against digital espionage. The Cyber Storm exercise underscored the deficiencies and vulnerabilities within the nation's cyber defenses. Since that time, researchers have suggested several new priorities concerning the nation's cyber defense strategy [77, 78]. One proposal includes the Identification of Nation's Critical Systems [80], aimed at recognizing the Internet-enabled systems vital to the nation's cyber defense and understanding the interdependencies among these systems. Various strategies have been proposed to safeguard the nation's critical infrastructures through vulnerability identification and remediation, as well as threat mitigation and response [79, 77].

III. Discussion

A number of common patterns appear in the malware found in the emerging technologies we described above.

Increased Attack through Web Browser – the services provided by emerging technologies are typically rendered by the use of web browsers. The web browsers are arguably one of the most commonly used applications providing users the interface to perform a wide range of activities connecting them to outside world. Web browsers are thus becoming increasingly important tool for millions of today's computer users. Like any other software piece, web browsers contain a number of vulnerabilities [70, 71]. The attackers use these vulnerabilities to take control of user's computer, steal user information, destroy the files, and use user's compromised computer to attack other computers. According to Osterman Research survey [3], 11 million malware variants were discovered by 2008 and 90% of these malware comes from hidden downloads from popular and often trusted websites.

Some of the common attacks that exploit the browser security are through extensions, often also called "plug-in" or "add-on" and scripting languages such as JavaScript or VBScript. Extensions are reusable software components which can be plugged into a browser to provide a new functionality or to customize user experience. Anyone, even with a little experience and software development train, can develop an extension and it can be downloaded freely by many unsuspecting users. Such extensions often contain software bugs which greatly increase the attack surface for the attackers to exploit them. The ability to run a scripting language such as JavaScript or VBScript allows web page authors to add a significant amount of features and interactivity to a web page. However, this same capability can be abused by attackers. A well-known vulnerability of exploiting scripting language is Cross-site Scripting (XSS). XSS enables attackers to inject malicious script into web pages. When unsuspecting clients view the web pages, the malicious code is executed to perform malicious activities on user's computer.

Most common defense mechanism against web browser vulnerabilities is done by the form of strengthened user authentication to ensure the web page is accessed only by authorized users. Content filtering is another popular technique [69] that has been used to detect any malicious scripts embedded in web pages. Some browsers or browser extensions can be configured to disable client-side scripts on a per-domain basis to toughen up the browser security. By nature, the Internet provides anonymity. Many attacks over the Internet have grown significantly to exploit such nature of the Internet. A requirement not only users authenticating themselves to a server but also the server authenticating itself to users in such a way that both parties are assured of the others' identity has been raised [61]. The technique termed as "mutual authentication" has gained a popularity to address this requirement. Some of the mutual attestation techniques studied today includes password authenticated key exchange (PAKE), Dynamic Security Skins (DSS), and the remote attestation proposed by Trusted Computing Group (TCG) [87].

Platform Switch – Cybercrime is switching its battle ground from desktop to other platforms, including mobile phones, tablet computers, and VoIP. With the expansion of the global mobile devices penetration and their expanded capacity to offer many online services, malwares created specific to mobile platforms are on the rise. According to the report [1], the number of unique detections of malware for Android has increased globally by 17 times in 2012 compared to the previous year. Other similar security trend reports [16, 17, 111] have similar concerns on the widening of malware spread on pervasive computing technologies. For example, the massively successful banking Trojan Zeus is already being adapted for the mobile platform [111]. Smashing, or SMS phishing, is another method cyber criminals are using to exploit mobile devices. VoIP technology continues to improve in terms of reliability and call quality. Now major vendors offer VoIP as a part of their integrated multimedia experience seen in applications like Skype, MSN, Facebook, etc. As the Internet telephony handles more and more data, it has become more frequent target of cyber-crime. Especially, the attacks on VoIP infrastructure are on the rise. When voice is digitized, encoded, compressed into packets and exchanged over IP networks, it is susceptible to misuse. It is concern that cyber criminals will be drawn to the VoIP medium to engage in voice fraud, data theft and other scams. VoIP systems are being used to support vishing (i.e. telephone-based phishing) schemes, which are now growing in popularity.

Social Engineering Scams – Industry experts [81, 89, and 90] have shown that popular social networking sites like Facebook or Twitters have been increasingly used as a delivery mechanism to get unsuspecting users to install or spread malware. There are virtually no limits to the creativity of adversaries spreading malware when social engineering is involved. For example, an adversary, often under a false pretense, would befriend a naive user over a social network and lure the user into deliberately executing malicious code on the victim's machine. The adversary asks a user to install a provided "codec" to view the movie or to click an image file attached to a spam email, but in fact these turn out to be malware. In addition, social networking becomes an increasingly important tool for cyber criminals to recruit money mules to assist their money laundering operations around the globe. Spammers are not only spoofing social networking messages to persuade targets to click on links in emails, they are taking advantage of users' trust of their social networking connections to attract new victims.

More Organized Attacks using Botnets – Bots (short for "robots") are malware programs that are covertly installed on a user's machine which allows an unauthorized user to remotely control the compromised computer for a variety of malicious purposes. Botnets are networks of machines that have been compromised by bot malware so that they are under the control of an adversary. Among the various forms of malware, botnets are emerging as the most serious threat against cyber security as they provide a distributed platform for major illegal activities in the cyber space including distributed denial of service attacks (DDoS) [15–17]. When bots are spread through the emerging technologies with hundreds and billions of registered users, the negative impact would be disastrous.

Earlier bots used covert channels on standard Internet Relay Chat (IRC) protocol to communicate with remotely located adversary who controls Command and Control (C&C) server. This form of communication is visible in plaintext and is fairly easy to detect. With the increase awareness of botnets, bot masters have modified their techniques and methods to avoid detection. Most commonly used methods are encrypted traffic, using Internet protocol, domain flux, and rootkits [82–85]. Newer bots use alternative protocols such as http or https. Http is useful for bot master because the protocol is the most commonly used in network traffic which makes it difficult to filter bot traffic. In addition, the use of https encrypts the commands making it difficult to detect or monitor the traffic between the adversary and botnets. Domain Flux is another method used by many botnets to avoid detection. The bot master generates a list of domain names and change the point of contact frequently. This makes it difficult to shut down, or block the C&C server even their locations are detected at a certain time which might change later after utilizing domain flux strategy. Bot masters also utilize rootkits that hide the fact that a system has been compromised. Rootkits, such as mebroot [4], avoid detection from antivirus software by using various techniques such as modifying boot records so that, when the computers boot up, it makes itself start before the antivirus can.

Inherently defense against botnet has been the signature-based defense mechanisms. Other than capturing the malware by understanding the signature of malware, the idea of Internet-scale simulated test beds and containment technology has been proposed [15]. The idea behind internet-scale emulator is to capture botnets via a large network of honeypots that lure any malware for further analysis. This allows observation of botnets behavior to devise new techniques to combat them. The use of the combination of virtualization and honey net techniques has been suggested for the test beds [84]. In these techniques, a virtualized network environment is deployed on unused address space to interact with malware with an aim of capturing its copy to enable further analysis. The research idea in the containment technology recognizes that bots (and other malware) are part of the computing environment. It advocates the idea to secure only a part of the system rather than trying to make a whole system secure allowing trusted transaction to run from a potentially untrusted system. There is industry research advancing virtual machines to Trusted Platform Module (TPM) and hypervisor technology in hardware and software to capture botnets using a small part of trusted compartment in an untrusted computing environment [15].

Insider Threats – previous research on cyber security has focused on protecting valuable resources from attacks by outsiders. However, statistics [16,17] show that a large amount of security and privacy breaches are due to insider attacks.

Protection from insider threats is challenging because insiders may have access to many sensitive resources and high-privileged system accounts. Similar style of exploitation is reported in [63, 64] when the authentication is compromised by insider.

Monitoring has been a central in insider threat research to examine if there are any different patterns of access that is done by insiders. The number of visualization tools have been proposed to monitor requested and granted privileges,

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.

Relative to each user and each object [63,64]. Multidisciplinary detection mechanisms are also popularly used to identify insider abuse and suspected anomalies. New detection techniques that are emerging in this area are: using data mining techniques [152], behavior- based detection to find intends of insider threats [15], and integrated model that combined prediction and detection techniques are suggested [63,64]. New breed of access control mechanisms have been attempted to protect the systems from over escalated privileges that are done by insiders who know the system. Some new techniques in access control especially targeting insider threats include splitting up the privileges and multi-level access control. Different anti-tampering technologies depending on whether insiders have illegal physical access or logical access have been investigated [63]. The protection that preserves the integrity of multi-layers at hardware, software and data with finer- grained controls has been explored in [63]. Audit trails have been also used to find any clues of who have accessed the system by examining system log files [15]. However, because insiders might have a high privilege over most system files including logs, some techniques are suggested to produce audit trails that are unalterable (e.g., once-writable) and non by-passable.

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 cyber security 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 [126].

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 trace back 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.

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 [122], 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 [16,17]. 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 [65]. One stream of research in this field concerns with the way data is accessed and disclosed while protecting privacy [65]. 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 [141]. Another stream of research conducted in this area concerned with the development of specification framework to build and reinforce privacy policy [88]. 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.

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.

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].

Global-scale identity management and trace back 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 [11,148–151].

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 [96–98], 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 trace back 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 trace back methods only work well for a single cooperative defense and skilled attackers easily evade most currently deployed trace back systems by tweaking the header IP addresses [96,15].

The development of global scale trace back 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.

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 [138–140].

IV. Conclusion

This survey focused on two aspects of information system: understanding vulnerabilities in exiting 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 trace back techniques to enable tracking down adversaries has also gained an attention as an important issue to address in the future.

References

1. <http://www.maawg.org/>, last accessed: June 2013.
2. <http://www.antiphishing.org/>, last accessed: June 2013.
3. <http://www.ostermanresearch.com/downloads.htm>, last accessed: June 2013.
4. <http://en.wikipedia.org/wiki/Mebroot>, last accessed: June 2013.
5. <http://www.emailtrackerpro.com>, last accessed: June 2013.
6. <http://www.tamos.com>, last accessed: June 2013.
7. <https://www.mandiant.com/resources/download/web-historian>, last accessed: June 2013.
8. http://www.majorgeeks.com/index.dat_analyzer_d5259.html, last accessed: June 2013.
9. <http://www.winpcap.org/>, last accessed: June 2013.
10. <http://www.riverbed.com/products-solutions/products/performance-management/wireshark-enhancement-products/Wireless-Traffic-Packet-Capture.html>, last accessed: June 2013.
11. <http://shibboleth.internet2.edu/>, last accessed: June 2013.
12. Australian Parliament the report of the inquiry into Cyber Crime, http://www.aph.gov.au/house/committee/coms/cybercrime/report/full_report.pdf.
13. www.it2trust.com/pdf/Aladdin.SafeWord_PO_SafeWord.pdf, last accessed: June 2013.
14. A. Cardenas, T. Roosta, G. Taban, S. Sastry, Cyber security basic defenses and attack trends, Fujitsu Lab., <http://www.flacp.fujitsulabs.com/~cardenas/Papers/Chap4v2.pdf>, last accessed: June 2013.
15. DHS S&T, Roadmap for cybersecurity research, Jan. 2009, <http://www.cyber.st.dhs.gov/docs/DHS-Cybersecurity-Roadmap.pdf>, last accessed: June 2013.
16. Annual Emerging Cyber Threats Report, Georgia Tech Information Security Center, <http://www.gtisc.gatech.edu/>, last accessed: June 2013.
17. Internet Security Threats Report. Symantec, <http://www.symantec.com/threatreport/>, last accessed: June 2013.
18. S.E. Goodman, H.S. Lin (Eds.), Toward a Safer and More Secure Cyberspace, The Nat'l Academics Press, 2007.
19. R.C. Newman, Computer Security: Protecting Digital Resources, first edition, Jones & Bartlett Publishers, February 20, 2009.
20. B.W. Lampon, Privacy and security – Usable security: how to get it, Commun. ACM 52 (11) (2009) 25–27.
21. A. Haeberlen, P. Kouznetsov, P. Druschel, Practical accountability for distributed systems, in: SOSP 2007, pp. 175–188.
22. M. Tehranipoor, C. Wang, Introduction to Hardware Security and Trust, Springer, 2011.
23. N. Potlapally, Hardware security in practice: Challenges and opportunities, in: HOST 2011, pp. 93–98.
24. Q. Li, H. Gao, B. Xu, Z. Jiao, Hardware threat: The challenge of information security, in: ISCSCT 2008, pp. 517–520.

25. R.S. Chakraborty, S. Narasimhan, S. Bhunia, Hardware Trojan: Threats and emerging solutions, in: HLDVT 2009, pp. 166–171.
26. R. Karri, J. Rajendran, K. Rosenfeld, M. Tehranipoor, Trustworthy hardware: Identifying and classifying hardware trojans, *IEEE Comput.* 43 (10) (2010) 39–46.
27. H. Mouratidis, Secure by design: Developing secure software systems from the group up, *Intern. J. Secure Software Eng.* 2 (3) (2011) 23–41.
28. A. Sadeghi, Trusted computing — special aspects and challenges, in: V. Geffert, et al. (Eds.), *SOFSEM*, in: *Lect. Notes Comput. Sci.*, vol. 4910, Springer, Berlin, 2008, pp. 98–117.
29. Trusted Computing Group, TPM Main, Part 1, Design Principles, Specification version 1.2. Revision 94, 2006.
30. Trusted Computing Group, TPM Main, Part 2, TPM Structures, Specification version 1.2. Revision 94, 2006.
31. Trusted Computing Group, TPM Main, Part 3, Design Principles, Specification version 1.2. Revision 94, 2006.
32. B. Beckert, R. Hähnle, P.H. Schmitt (Eds.), *Verification of Object- Oriented Software: The KeY Approach*, *Lect. Notes Comput. Sci.*, vol. 4334, Springer, Heidelberg, 2007.
33. C. Hoare, J. Misra, G.T. Leavens, N. Shankar, The verified software initiative: A manifesto, *ACM Comput. Surv.* 41 (2009) 1–22.
34. K.R.M. Lein, An automatic program verifier for functional correctness, in: E.M. Clarke, A. Voronkov (Eds.), *LPAR-16 2010*, in: *Lect. Notes Comput. Sci.*, vol. 6355, Springer, Heidelberg, 2010, pp. 348–370.
35. M. Sitaraman, B. Adcock, J. Avigad, Building a push-button RESOLVE verifier: Progress and challenges, in: *Formal Aspects of Computing*, 2010, pp. 1–20.
36. J. MaManus, *The CERT Sun Microsystems Secure Coding Standard for Java*, CERT, 2009.
37. R. Seacord, *Top 10 Secure Coding Practice*, CERT, 2010.
38. R. Gennaro, J. Katz, H. Krawczyk, T. Rabin, Secure network coding over the integers, in: P.Q. Nguyen, D. Pointcheval (Eds.), *PKC 2010*, in: *Lect. Notes Comput. Sci.*, vol. 6056, Springer, Heidelberg, 2010, pp. 142–160.
39. M. Howard, D. LeBlanc, J. Viega, *19 Deadly Sins of Software Security*, McGraw–Hill, 2005.
40. K. Tsipenyuk, B. Chess, G. McGraw, Seven pernicious kingdoms: A taxonomy of software security errors, *IEEE Secur. Priv.* 3 (6) (2005) 81–84.
41. C.B. Haley, R. Laney, J.D. Moffett, B. Nuseibeh, Security requirements engineering: A framework for representation and analysis, *IEEE Trans. Softw. Eng.* 34 (1) (2008) 133–153.
42. G. McGraw, *Software Security: Building Security In*, Addison– Wesley, 2006.
43. M.I. Sharif, A. Lanzi, J.T. Giffin, W. Lee, Impeding malware analysis using conditional code obfuscation, in: *Network and Distributed System Security Symposium (NDSS)*, 2008.
44. J.-M. Borello, L. Mé, Code obfuscation techniques for metamorphic viruses, *J. Comput. Virol.* 4 (3) (2008) 211–220.
45. F.T. Sheldon, V. Vishik, Moving toward trustworthy systems: R&D essentials, *IEEE Comput. Mag.* (2010) 31–40.
46. W. Stallings, *Cryptography and Network Security Principles and Practices*, third edition, Pearson Educations, 2010.
47. E. Cole, R. Krutz, J. Conley, *Network Security Bible*, second edition, Wiley Publishing, 2011.
48. T. Rubya, N. Prema Latha, B. Sangeetha, A survey on recent security trends using quantum cryptography, *IJCSE* 2 (9) (2010) 3038–3042.
49. E.S. Pilli, R.C. Joshi, R. Niyogi, Network forensic frameworks: Survey and research challenges, *Dig. Investigation (Int’l. J. Dig. Investigation)* (2010), in press.
50. A. Almulhem, I. Traore, Experience with engineering a network forensics system, in: C. Kim (Ed.), *ICOIN 2005*, in: *Lect. Notes Comput. Sci.*, vol. 3391, Springer, Heidelberg, 2005, pp. 62–71.
51. B.J. Nikkel, A portable network forensic evidence collector, *Dig. Investigation (Int’l. J. Dig. Investigation)* 3 (3) (2006) 127–135.
52. A. Mairh, D. Barik, K. Verma, D. Jena, Honeypot in network security: a survey, in: *ICCCS*, 2011, pp. 600–605.
53. F. Fischer, F. Mansmann, D.A. Keim, S. Pietzko, M. Waldvogel, Large-scale network monitoring for visual analysis of attacks, in:
54. J.R. Goodall, G.J. Conti, K.-L. Ma (Eds.), *VizSEC*, in: *Lect. Notes Comput. Sci.*, vol. 5210, Springer, 2008, pp. 111–118.
55. M. Vrabie, J. Ma, J. Chen, D. Moore, E. Vandekieft, A. Snoeren, G. Voelker, S. Savage, Scalability, fidelity and containment in the Potemkin virtual honeyfarm, in: *Proceedings of the 2005 Symposium on Operating Systems Principles*, October 2005.
56. H.K. Lu, A. Ali, Communication security between a computer and hardware token, in: *ICONS 2008*, pp. 220–225.

57. F. Aloul, S. Zahidi, W. El-Hajj, Two factor authentication using mobile phones, in: IEEE International Conference on Computer Systems and Applications (AICCSA), Rabat, Morocco, May 2009.
58. D. Ilett, US bank gives two-factor authentication to millions of customers, available at <http://www.silicon.com/financialservices/0,3800010322,39153981,00.htm>, 2005.
59. D. de Borde, Two-factor authentication, Siemens Enterprise Communications UK-Security Solutions, available at [http://www.insight.co.uk/files/whitepapers/Two-factor_authentication_\(White_paper\).pdf](http://www.insight.co.uk/files/whitepapers/Two-factor_authentication_(White_paper).pdf), 2008.
60. J. Bringer, H. Chabanne, An authentication protocol with encrypted biometric data, in: AFRICACRYPT, in: Lect. Notes Comput. Sci., 2008, pp. 109–124.
61. M.K. Khan, J.S. Zhang, X.M. Wang, Chaotic hash-based fingerprint biometric remote user authentication scheme on mobile devices, *Chaos Solutions & Fractals* 35 (2008) 519–524.
62. G. Coker, J. Guttman, P. Loscocco, J. Sheehy, B. Sniffen, Attestation: Evidence and trust, in: ICICS '08, 2008, pp. 1–18.
63. J. Jang, H. Hwang, S. Nepal, Biometric enabled portable trusted computing platform, in: TurstCom 2011, pp. 436–442.
64. The CERT guide to insider threats: How to prevent, detect, and respond to theft of critical information, sabotage, and fraud, www.cert.org/archive/pdf/insidercross051105.pdf.
65. J. Hunker, C.W. Probst, Insiders and insider threats—An overview of definitions and mitigation techniques, *J. Wireless Mobile Netw. Ubiquitous Comput. Dependable Appl.* 2 (1) (2011) 4–27.
66. P. Guarda, N. Zannone, Towards the Development of Privacy-Aware Systems, *Information and Software Technology*, 2008.
67. K. Dahbur, B. Mohammad, A.B. Tarakji, A survey of risks, threats and vulnerabilities in cloud computing, 2011, pp. 12–18.
68. H. Takabi, J. Joshi, G. Ahn, Security and privacy challenges in cloud computing environments, *IEEE Secur. Priv.* (2010) 24–31.
69. Q. Zhang, L. Cheng, R. Boutaba, Cloud computing: state-of-the-art and research challenges, *J. Internet Serv. Appl.* 1 (2010) 7–18.
70. M.T. Louw, J.S. Lim, V.N. Venkatakrishnan, Extensible web browser security, in: B.M. Hammerli, R. Sommer (Eds.), DIMVA, in: Lect. Notes Comput. Sci., vol. 4579, Springer, 2007, pp. 1–19.
71. C. Soghoian, A remote vulnerability in Firefox extensions, <http://paranoia.dubfire.net/2007/05/remote-vulnerability-in-firefox.html>, last accessed: June 2013.
72. C. Reis, A. Barth, C. Pizano, Browser security: lessons from google chrome, *Commun. ACM* 52 (2009) 45–49.
73. P. Koopman, Embedded system security, *IEEE Comput.* 37 (7) (2004) 95–97.
74. S. Parameswaran, T. Wolf, Embedded systems security – an overview: DAES 2008, vol. 12, pp. 173–183, <http://dx.doi.org/10.1007/s10617-008-9027-x>.
75. J.P. Walters, Z. Liang, Wireless sensor network security: A survey, in: Y. Xiao (Ed.), *Security in Distributed, Grid, and Pervasive Computing*, Auerbach Publications, CRC Press, 2006.
76. Y. Zhou, Y. Fang, Y. Zhang, Securing wireless sensor networks: a survey, *IEEE Commun. Surv. Tutor.* 10 (3) (2008) 6–28.
77. S.J. Collier, A. Lakoff, The vulnerability of vital systems: How “critical infrastructure” became a security problem, in: *Critical Infrastructure, Risk and (In)security*, 2008, pp. 17–39.
78. C.W. Ten, Cybersecurity for critical infrastructures: Attack and defense modeling, *IEEE Trans. Syst. Man Cybern.* 40 (4) (2010) 853–865.
79. Critical infrastructure protection report, Government Accountability Office, Washington, DC, May 2005. [Online]. Available: <http://www.gao.gov/new.items/d05434.pdf>, last accessed: June 2013.
80. J.M. Weiss, Control systems cybersecurity—maintaining the reliability of the critical infrastructure, in: Testimony of Joseph M. Weiss Control Systems Cybersecurity Expert before the House Government Reform Committee’s Subcommittee on Technology, Information Policy, Intergovernmental Relations, and the Census U.S. House of Representatives, Mar. 30, 2004.
81. W.L. McGill, B.M. Ayyub, The meaning of vulnerability in the context of critical infrastructure protection, in: *Critical Infrastructure Protection: Elements of Risk*, School of Laws, George Mason Univ., Arlington, VA, Dec. 2007.
82. S. Abraham, I.S. Chengalur-Smith, An overview of social engineering malware: Trends, tactics, and implications, *Technol. Soc.* 32 (2010) 183–196.
83. M. Feily, A. Shahrestani, S. Ramadass, A survey of botnet and botnet detection, in: SECURWARE 2009, pp. 268–273.
84. M. Bailey, E. Cooke, F. Jahanian, et al., A survey of botnet technology and defense, in: CATCH 2009, pp. 299–304.
85. Z. Zhu, G. Lu, Y. Chen, et al., Botnet research survey, in: COMPSAC 2008, pp. 967–972.
86. C. Li, W. Jiang, X. Zou, Botnet: survey and case study, in: ICICIC 2009, pp. 1184–1187.

87. S. Nepal, J. Zic, D. Liu, J. Jang, Trusted computing platform in your pocket, in: ECU 2010, pp. 812–817,.
88. R. Sailer, X. Zhang, T. Jaeger, L. van Doorn, Design and implementation of a TCG-based integrity measurement architecture, SSYM 2004,.
89. M. Johnson, S. Egelman, S. Bellovin, Facebook and privacy: it's complicated, in: Proc. SOUPS 2012, ACM, 2012.
90. C. Dwyer, S. Hiltz, K. Passerini, Trust and privacy concern within social networking sites: A comparison of Facebook and MySpace, in: Americas Conference on Information Systems (AMCIS), Keystone, Colorado, USA, 2007.
91. R. Gross, A. Acquisti, Information revelation and privacy in online social networks (the Facebook case), in: Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society, 2005, pp. 71–80.
92. G. Hogben, Security issues and recommendations for online social networks, in: Position Paper. ENISA, European Network and Information Security Agency, 2007.
93. R.C. Clark, Cyber War, Ecco, 2010.
94. D. Weitzner, et al., Information accountability, Commun. ACM 51 (6) (2008) 82–87.
95. J. Yao, S. Chen, C. Wang, Accountability as a service for the cloud, in: SCC 2010, pp. 81–88.
96. S. Egelman, J. King, R.C. Miller, N. Ragouzis, E. Shehan, Security user studies: methodologies and best practices, in: CHI 2007, <http://dx.doi.org/10.1145/1240866.1241089>.
97. J. Wang, J.N. Whitley, R.C.W. Phan, Unified parametrizable attack tree, J. Inform. Security Res. (IJISR) 1 (1) (2011) 20–26.
98. A. John, T. Sivakumar, Ddos: Survey of traceback methods, in: IJRTE, 2009.
99. B. Kordy, M. Pouly, P. Shweitzer, Computational aspects of attack- defense trees, in: SIIS 2011, in: Lect. Notes Comput. Sci., vol. 7053, 2012, pp. 103–116.
100. L.F. Cranor, S. Garfinkel (Eds.), Security and Usability: Designing Secure Systems that People Can Use, O'Reilly Media, 2005.
101. L.F. Cranor, S. Garfinkel, Secure or usable?, IEEE Secur. Priv. 2 (2004) 16–18.
102. global -e-commerce-sales-asia-factors-big/, last accessed: June 2013.
103. S.M. Bellovin, D.D. Clark, A. Perrig, D. Song, A clean-slate design for the next-generation secure internet, National Science Foundation, Tech. Rep., Mar. 2005.
104. A. Feldmann, Internet clean-slate design: What and why?, Comput. Commun. Rev. 37 (3) (2007) 59–64.
105. M. Conti, S. Chong, S. Fdida, W. Jia, H. Karl, Y.-D. Lin, P. Mahonen, M. Maier, R. Molva, S. Uhlig, M. Zukerman, Research challenges towards the Future Internet, Comput. Commun. 34 (18) (2011) 2115–2134.
106. S. Paul, J. Pan, R. Jain, Architectures for the future networks and the next generation Internet: A survey, Comput. Commun. 34 (1) (2011) 2–42.
107. S. Furnell, Making security usable: Are things improving?, Comput. Secur. 26 (6) (2007) 434–443.
108. B.D. Payne, W.K. Edwards, A brief introduction to usable security, IEEE Internet Comput. 12 (2008) 13–21.
109. E.E. Schultz, Where have the worms and viruses gone? New trends in malware, Comput. Fraud Secur. 2006 (7) (2006) 4–8.
110. U. Bayer, I. Habibi, D. Balzarotti, E. Kirda, C. Kruegel, A view on current malware behaviours, in: USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), April 2009.
111. G. Cluley, Sizing up the malware threat-key malware trends for 2010, Netw. Secur. (2010), [http://dx.doi.org/10.1016/S1353-4858\(10\)70045-3](http://dx.doi.org/10.1016/S1353-4858(10)70045-3).
112. http://gocsi.com/sites/default/files/uploads/2007_CSI_Survey_full_color_no_marks.indd_.pdf, last accessed: June 2013.
113. P. García-Teodoroa, J. Díaz-Verdejoa, G. Maciá-Fernández, E. Vázquez, Anomaly-based network intrusion detection: Techniques, systems and challenges, Comput. Secur. 28 (1–2) (2009) 18–28.
114. W. Hasselbring, R. Reussner, Toward trustworthy software systems, Computer 39 (4) (2006) 91–92, <http://dx.doi.org/10.1109/MC.2006.142>.
115. L.M.R. Gadelha Jr., B. Clifford, M. Mattoso, M. Wilde, Provenance management in Swift, Future Gener. Comput. Syst. 27 (6) (2011) 775–780.
116. L. Moreau, J. Freire, J. Futrelle, R.E. McGrath, J. Myers, P. Poulson, The open provenance model, <http://openprovenance.org/>, last accessed: June 2013.
117. K.-K. Muniswamy-Reddy, D.A. Holland, U.B.M.I. Seltzer, Provenance-aware storage systems, in: Proc. USENIX Conf., Usenix, 2006, pp. 43–56.
118. N.R. Mathiasen, S. Bodker, Experiencing security in interaction design, in: Proc. CHI 2011, 2011, pp. 2325–2334.
119. L. Barkhuus, The mismeasurement of privacy: using contextual integrity to reconsider privacy in HCI, in: Proc. CHI 2012, 2012, pp. 367–376.
120. M.Y. Ivory, M.A. Hearst, The state of the art in automating usability evaluation of user interfaces, ACM Comput. Surv. 33 (2001) 470–516.

121. M. Madden, Privacy management on social media sites, <http://pewinternet.org/Reports/2012/Privacy-management-on-social-media.aspx>, February 2012.
122. J. Pan, S. Paul, R. Jain, A survey of the research on future Internet architectures, *IEEE Commun. Mag.* 49 (7) (2011) 26–36.
123. H.P. Bui, J. Cox, S. Theobald, S. Wiegand, Trustworthy IT systems, trustworthy.googlecode.com/svn-history/r27/trunk/tex/doc.pdf, 2012.
124. S. Ding, X.J. Ma, S.L. Yang, A software trustworthiness evaluation model using objective weight based evidential reasoning approach, *Knowl. Inf. Syst.* 33 (2012) 171–189.
125. T. Eze, R. Anthony, C. Walshaw, A. Soper, A new architecture for trustworthy autonomic systems, in: *EMERGING 2012*, 2012, pp. 62–68.
126. A. Casimiro, P. Verissimo, D. Kreutz, TRONE: Trustworthy and resilient operations in a network environment, in: *Proc. Dependable Sys. and Networks Workshops*, 2012, pp. 1–6.
127. R. Maas, E. Maehle, Applying the organic robot control architecture ORCA to cyber-physical systems, in: *Proc. SEAA*, 2012, pp. 250–257.
128. H.S. Lim, G. Ghinita, E. Bertino, A game theoretic approach for high-assurance of data trustworthiness in sensor networks, in: *Proc. ICDE*, 2012, pp. 1192–1203.
129. M.A.P. Leandro, T.J. Nascimento, Multi-tenancy authorization system with federated identity for cloud-based environments using shibboleth, in: *Proc. ICN*, 2012, pp. 88–93.
130. J. Jensen, Federated identity management challenges, in: *Proc. ARES*, 2012, pp. 230–235.
131. D.W. Chadwick, M. Hibbert, Towards automated trust establishment in federated identity management, in: C. Fernandez- Gago, et al. (Eds.), *IFIPTM 2013*, in: *IFIP AICT*, vol. 401, 2013, pp. 33–48.
132. S. Mathew, M. Petropoulos, H.Q. Ngo, A data-centric approach to insider attack detection in database systems, in: *RAID 2010*, in: *Lect. Notes Comput. Sci.*, vol. 6307, 2010, pp. 382–401.