Journal of Computer and System Sciences 80 (2014) 973–993

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

A survey of emerging threats in cybersecurity

Julian Jang-Jaccard, Surya Nepal *

CSIRO ICT Centre, Australia

ARTICLE INFO

Article history:
Received 25 September 2012
Received in revised form 15 March 2013
Accepted 27 August 2013
Available online 10 February 2014

Keywords:
Cybersecurity
Malware
Emerging technology trends
Emerging cyber threats
Cyber attacks and countermeasures

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 cybersecurity 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.

1. Introduction

Our society, economy, and critical infrastructures have become largely dependent on computer networks and information technology solutions. Cyber attacks become more attractive and potentially more disastrous as our dependence on information technology increases. According to the Symantec cybercrime report published in April 2012 [17], cyber attacks cost US$114 billion each year. If the time lost by companies trying to recover from cyber attacks is counted, the total cost of cyber attacks would reach staggering US$385 billion [17]. Victims of cyber attacks are also significantly growing. Based on the survey conducted by Symantec which involved interviewing 20,000 people across 24 countries, 69% reported being the victim of a cyber attack in their lifetime. Symantec calculated that 14 adults become the victim of a cyber attack every second, or more than one million attacks every day [105].

Why cyber attacks flourish? It is because cyber attacks are cheaper, convenient and less risky than physical attacks [1]. Cyber criminals only require a few expenses beyond a computer and an Internet connection. They are unconstrained by geography and distance. They are difficult to identity and prosecute due to anonymous nature of the Internet. Given that attacks against information technology systems are very attractive, it is expected that the number and sophistication of cyber attacks will keep growing.

* Corresponding author.
E-mail addresses: julian.jang-jaccard@csiro.au (J. Jang-Jaccard), surya.nepal@csiro.au (S. Nepal).

http://dx.doi.org/10.1016/j.jcss.2014.02.005
0022-0000/

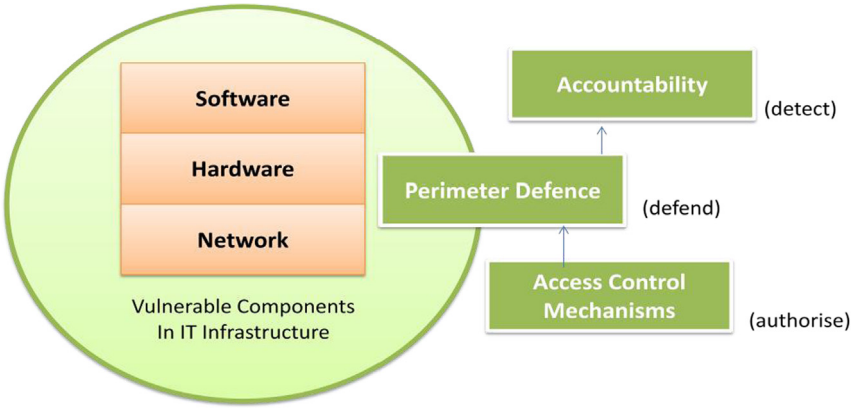

Fig. 1. Vulnerabilities and defense strategies in existing systems.

Cybersecurity concerns with the understanding of surrounding issues of diverse cyber attacks and devising defense strategies (i.e., countermeasures) that preserve confidentiality, integrity and availability of any digital and information technologies [18].

- **Confidentiality** is the term used to prevent the disclosure of information to unauthorized individuals or systems.
- **Integrity** is the term used to prevent any modification/deletion in an unauthorized manner.
- **Availability** is the term used to assure that the systems responsible for delivering, storing and processing information are accessible when needed and by those who need them.

Many cybersecurity experts believe that malware is the key choice of weapon to carry out malicious intends to breach cybersecurity efforts in the cyberspace [12]. Malware refers to a broad class of attacks that is loaded on a system, typically without the knowledge of the legitimate owner, to compromise the system to the benefit of an adversary. Some exemplary classes of malware include viruses, worms, Trojan horses, spyware, and bot executables [15]. Malware infects systems in a variety of ways for examples propagation from infected machines, tricking user to open tainted files, or alluring users to visit malware propagating websites. In more concrete examples of malware infection, malware may load itself onto a USB drive inserted into an infected device and then infect every other system into which that device is subsequently inserted. Malware may propagate from devices and equipments that contain embedded systems and computational logic. In short, malware can be inserted at any point in the system life cycle. Victims of malware can range anything from end user systems, servers, network devices (i.e., routers, switches, etc.) and process control systems such as Supervisory Control and Data Acquisition (SCADA). The proliferation and sophistication of fast growing number of malware is a major concern in the Internet today.

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

Malware evolves through time capitalizing on new approaches and exploiting the flaws in the emerging technologies to avoid detection. We describe a number of new patterns of malware attacks present in the emerging technologies. In choosing emerging technologies for illustration, we focus a few that have changed the way we live our daily life. These include social media, cloud computing, smartphone technology, and critical infrastructure. We discuss unique characteristics of each of these emerging technologies and how malware utilizes the unique characteristics to proliferate itself. For example, social media, such as social networking sites and blogs, are now an integral part of our life style as many people are journaling about their life events, sharing news, as well as making friends. Realizing its potential to connect millions people at one go, adversaries use social media accounts to befriend unsuspecting users to use as vehicles for sending spam to the victim's friends while the victim's machine is repurposed into a part of botnet. Cloud computing paradigm allows the

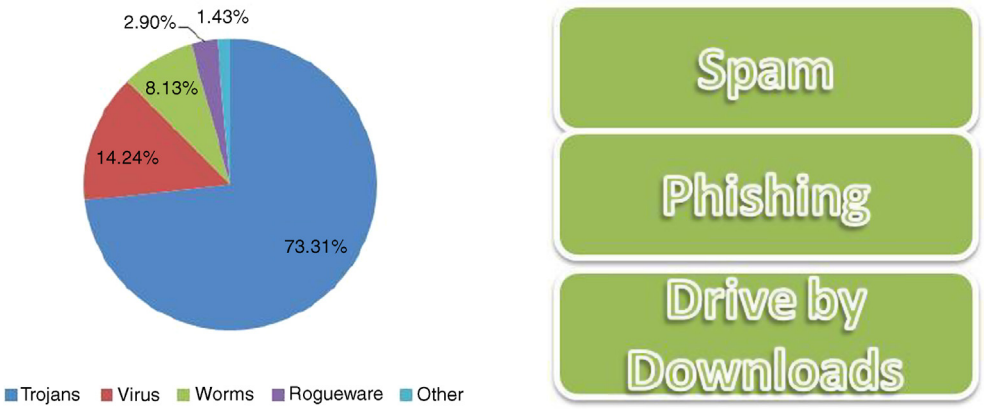

Fig. 2. Types of malware and mediums to spread them [101].

use of computer resources like utilities where the users pay only for the usage without having to set up any upfront expense or requiring any skills in managing complex computing infrastructure. The growing trove of data concentrated in the cloud storage services is now attracting attackers. In June 2012, attackers compromised Distributed Denial of Service (DDoS) mitigation service on CloudFlare by using flaws in AT&T's voicemail service for its mobile users; similarly, Google's account-recovery service for its Gmail users [19]. With the subjected growth by 2 billion smartphone users by 2015, a significant growth in mobile malware has been witnesses in recent times. For example, the number of unique detections of malware for Android increased globally by 17 times in 2012 from the previous year [107]. There is also growing concerns in cyber threats to critical infrastructure such as electricity grids and healthcare systems to use in terrorism, sabotage and information warfare. Apart from investigating exploitations through unique characteristics in the selected emerging technologies, we also discuss general malware attack patterns appear in them to understand the methods and trends of the new attacks.

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

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.

2. Malware as attack tool

In early days, malware was simply written as experiments often to highlight security vulnerabilities or in some cases to show off technical abilities. Today, malware is used primarily to steal sensitive personal, financial, or business information for the benefit of others [129,131]. For example, malware is often used to target government or corporate websites to gather guarded information or to disrupt their operations. In other cases, malware is also used against individuals to gain personal information such as social security numbers or credit card numbers. Since the rise of widespread broadband Internet access that is cheaper and faster, malware has been designed increasingly not only for the stealth of information but strictly for profit purposes [130]. For example, the majority of widespread malware have been designed to take control of user's computers for black market exploitation such as sending email spam or monitoring user's web browsing behaviors and displaying unsolicited advertisements. Based on Anti-Phishing group report [101], there was a total of 26 million new malware reported in 2012. Fig. 2 describes relative proportions of the types of new malware samples identified in the second half of 2012 reported by the Anti-Phishing group.

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

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

- *Spam* refers to sending irrelevant, inappropriate and unsolicited messages to thousands or millions of recipients. Spam has turned out to be a highly profitable market since spam is sent anonymously with no costs involved beyond the management of mailing lists. Due to such low barrier to entry, spammers are numerous, and the volume of unsolicited mail has grown enormously. In the year 2011, the estimated figure for spam messages is around seven trillion [2]. This figure includes the cost involved in lost productivity and fraud, and extra capacity needed to cope with the spam. Today, most widely recognized form of spam is email spam. According to the Message Anti-Abuse Working Group report [1], between 88–92% of email messages sent in the first half of 2010 carried spam.
- *Phishing* is a way of attempting to acquire sensitive information such as username, password or credit card details by masquerading as a trustworthy entity. Most phishing scams rely on deceiving a user into visiting a malicious web site claiming to be from legitimate businesses and agencies. Unsuspecting user enters private information in the malicious web site which is then subsequently used by malicious criminals. Most methods of phishing use some form of technical deception designed to make a link in an email (and spoofed website) appear to belong to a legitimate organization, such as well known bank. Misspelled URLs or the use of sub-domains are common tricks used by phishers. The Anti-Phishing technical report [101] stated that, there was a visible trend of phishers in 2011 to hide their intentions by avoiding the use of obvious IP host to host their fake login pages. Instead the phishers preferred to host on a compromised domain to avoid detection. It is reported that there was 16 percent drop in the number of phishing URLs containing the spoofed company name in the URL. These combined trends show how phishers are adapting as users becoming more informed and knowledgeable about the traits of a typical phish.
- *Drive-by Downloads* concerns the unintended downloads of malware from the Internet and have been increasingly used by the attackers to spread malware fast. Drive-by downloads happen in a variety of situations; for example, when a user visits a website, while viewing an email message by user or when users click on a deceptive pop-up window. However, the most popular drive-by downloads occur by far when visiting websites. An increasing number of web pages have been infected with various types of malware. According to Osterman Research survey [3], 11 million malware variants were discovered by 2008 and 90% of these malware comes from hidden downloads from popular and often trusted websites. Before a download takes place, a user is first required to visit the malicious site. To lure the user into visiting a website with malicious content, attackers would send spam emails that contain links to the site. When unsuspecting user visits the malicious website, malware is downloaded and installed in the victim's machine without the knowledge of the user. For example, the infamous Storm worm makes use of its own network, multiple of infected computers, to send spam emails containing links to such attack pages [102].

3. Exploiting existing vulnerabilities

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

3.1. Hardware

Hardware is the most privileged entity and has the most ability to manipulate a computing system. This is the level where it has the potential to give attackers considerable flexibility and power to launch malicious security attacks if the hardware is compromised [23,24]. Compare to software level attacks where many security patches, intrusion detection tools,

and anti-virus scanners exist to detect malicious attacks periodically, many of the hardware-based attacks have the ability to escape such detection. Taking advantage in lack of tools support in hardware detection, the hardware-based attacks have been reported to be on the rise [23].

Among different types of hardware misuse, hardware Trojan is the most hideous and common hardware exploits [24]. The hardware Trojans are malicious and deliberately stealthy modification made to electronic devices such as Integrity Circuits (IC) in the hardware [25]. The hardware Trojans have a variety of degrees which cause different types of undesirable effects. A hardware Trojan might cause an error detection module to accept inputs that should be rejected. A Trojan might insert more buffers in the chip's interconnections and hence consume more power, which in turn could drain the battery quickly. In more serious case, Denial-of-Service (DoS) Trojans prevent operation of a function or resource. A DoS Trojan can cause the target module to exhaust scarce resources like bandwidth, computation, and battery power. It could also physically destroy, disable, or alter the device's configuration, for example, causing the processor to ignore the interrupt from a specific peripheral.

Illegal clones of hardware become source of hardware-based exploitation since the chances of illegally counterfeited hardware to contain malicious backdoor or hardware Trojans increase. The chance to produce unauthentic hardware has increased with a new trend in IT companies trying to reduce their IT expense via outsourcing and buying off untrusted hardware from online sites. Karri et al. [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 examines a number of way cryptographic algorithm's secret key leaked as a result of analyzing radio frequency.

A number of techniques have been proposed to thwart attacks on hardware level. Tamper-resistant hardware devices have become an important consideration due to its criticality as an entry point to the overall system security. Trusted Platform Module (TPM) provides cryptographic primitives and protected storage along with the functionality to exchange tamper resistant evidence with remote servers [29–31,28]. The term Trusted Computing Base (TCB) has been defined to refer to parts of a system, the set of all hardware and software components, to be critical to the overall security of the system. The TCB must not contain any bugs or vulnerabilities occurring inside because this might jeopardize the security of the entire system. An exhaustive and rigorous examination of its code base is conducted through computer-assisted software audit or program verification to ensure the security of TCB. In a hardware watermarking, the ownership information is embedded and concealed in the description of a circuit preventing the host object from illegal counterfeit. Hardware Obfuscation is a technique to modify the description or the structure of electronic hardware to intentionally conceal its functionality [22]. These techniques are used to prevent adversaries from obtaining the original design or counterfeiting /cloning important parts of the hardware such as IC units. Some of the countermeasures to count against side channel attacks includes introducing noises so that the physical information cannot be directly displayed, filtering some parts of physical information, and making/blinding which seeks to remove any correlation between the input data and side channel emission [23,24].

3.2. Software defects

A software bug is the common term used to describe an error, flaw, mistake, or fault in a computer program such as internal OS, external I/O interface drivers, and applications [103]. Cyber attacks utilize the software bugs in their benefits to cause the systems to behave unintended ways that are different from their original intent. The majority of cyber attacks today still occur as a result of exploiting software vulnerabilities caused by software bug and design flaws [104].

Software-based exploitation occurs when certain features of software stack and interface is exploited. Most common software vulnerabilities happen as a result of exploiting software bugs in the memory, user input validation, race conditions and user access privileges [40,39,42]. Memory safety violations are performed by attackers to modify the contents of a memory location. Most exemplary technique is buffer overflow. The buffer overflow occurs when a program tries to store more data in a buffer than it was intended to hold. Since buffers are created to contain a finite amount of data, the extra information can overflow into adjacent buffers, corrupting or overwriting the valid data held in them. It allows attackers to interfere into existing process code. Input validation is the process of ensuring that the input data follows certain rules. Incorrect data validation can lead to data corruption such as seen in SQL injection. SQL injection is one of the most well known techniques that exploit a program bug in a website's software. An attacker injects SQL commands from the web form either to change the database content or dump the database information like credit cards or passwords to the attacker. Adversary exploits a flaw in a process where the output of the process is unexpectedly and critically dependent on the timing of other events. The time of check to time of use is a bug caused by changes in a system between the checking of a condition and the use of the results of that check. It is also called exploiting race condition error. Privilege confusion is an

act of exploiting a bug by gaining elevated access to resources that are normally protected from an application or user. The result is that adversaries with more privileges perform unauthorized actions such as accessing protected secret keys.

In the programming community, a number of projects have been initiated that are devoted to increasing the security as a major goal [36–38]. Not only attending to fix inherent common set of security flaws, the primary concern of these projects is to provide new ideas in an attempt to create a secure computing environment. In a code review-based secure coding practice, software engineers identify common programming errors that lead to software vulnerabilities, establish standard secure coding standards, educate software developers, and advance the state of the practice in secure coding. In a language-based secure coding practice, techniques are developed to ensure that programs can be relied on not to violate important security policies. The most widely used techniques include analysis and transformation. A well-known form of analysis is "type checking" where the program detects any unsafe type of objects before the program is run. Another well-known form of program transformation is the addition of runtime checks where the program is instrumented in a way that prevents the program from making any policy-violating transformation [42]. Code obfuscation is a process of producing source or machine code that has been made difficult to understand for humans [43,44]. Programmers often deliberately obfuscate code to conceal its purpose or its logic to prevent any possibility with reverse engineering. Secure design and development cycle has also been proposed in [41,27] which provides a set of design techniques enabling efficient verification that a piece of system component is free of any potential defects from its original design. Though they are not straightforward approaches, formal methods provide the ability to comprehensively explore the design and identify intricate security vulnerabilities. Tools [34,35] and techniques [32,33] have been developed to facilitate the verification of mission critical security properties. These tools and techniques help to translate higher-level security objectives into a collection of atomic properties to be verified.

3.3. Network infrastructure and protocol vulnerabilities

The early network protocol was developed to support entirely different environment we have today in a much smaller scale and often does not work properly in many situations it is used today. Weaknesses in network protocols are complicated when both system administrators and users have limited knowledge of the networking infrastructure [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.

One of the most common network attacks occurs by exploiting the limitations of the commonly used network protocols Internet Protocol (IP), Transmission Control Protocol (TCP) or Domain Name System (DNS) [14]. The IP is the main protocol of the network layer. It provides the information needed for routing packets among routers and computers of the network. The original IP protocol did not have any mechanism to check the authenticity and privacy of data being transmitted. This allowed the data being intercepted or changed while they are transmitted over unknown network between two devices. To prevent the problem, IPSec was developed to provide encryption of IP traffic. In many years, IPSec has been used as one of the main technology for the creation of a virtual private network (VPN) which creates a secure channel across the Internet between a remote computer and a trusted network (i.e., company intranet). TCP sits on top of the IP to transmit the packets in reliable (i.e., retransmitting lost packets) and ordered delivery of the packets. SSL was originally developed to provide end-to-end security, as oppose to only layer-based protocol, between two computers which sits over the transmission control protocol (TCP). SSL/TLS is commonly used with http to form https for secure Web pages. The domain name server (DNS) is the protocol that translates the human-readable host names into 32-bit Internet protocol (IP) addresses. It is essentially works as a directory book for the Internet telling routers to which IP address to direct packets when the user gives a url. Because DNS replies are not authenticated, an attacker may be able to send malicious DNS messages to impersonate an Internet server. Another major concern about DNS is its availability. Because a successful attack against the DNS service would create a significant communication disruption in the Internet, DNS has been the target of several Denial-of-Service (DoS) attacks.

Cryptography is an essential tool to protect the data that transmits between users by encrypting the data so that only intended users with appropriate keys can decrypt the data. Cryptography is the most commonly used mechanism in protecting data. A survey conducted by Computer Security Institute in 2007 [132] revealed that 71% of companies utilized encryption for their data in transit. Further to protect today's sophisticated attackers exploiting the limitations of existing cryptography algorithms, a number of movements are on the rise. The US National Institute of Standards and Technology (NIST) recently announced discontinuation of SHA-1 and to use the Advanced Hash Standard (ASH) from 2012 [15]. The potential to use identity-based encryption is an active research agenda for applications that require high-speed encryption to avoid the use of slow 2048 bit RSA key length along with impractical involvement of the trusted certifying authority [15]. Quantum cryptography is an emerging technology in which two parties simultaneously generate shared, secret cryptographic key material using the transmission of quantum states of light [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.

3.4. Discussion

Though many separate techniques and proposals exist to remedy vulnerabilities in hardware, software and network layers, rather than focusing on each layer, bundled security protection techniques that protect everything inside from outside attacks have been adopted in the traditional approach. The overwhelming majority of companies employ a perimeter defense security model to guard the company's network from any potential intrusion from outside [47]. This approach focuses on "layered defense" or "defense in depth" strategies in which important internal IT assets, such as servers or mission critical data, are protected by walls and fortifications.

Typical perimeter defenses include technologies such as firewalls and intrusion detection systems (IDS). The firewall has been the most widely used technology to protect the internal assets. Its primary objective is to control the incoming and outgoing network traffic by analyzing the data packets and determining whether it should be allowed through or not, based on a predetermined rule set. A firewall can be placed in different layer in the network infrastructure. Network layer firewalls, also called packet filters, operate at a relatively low level of the network layer and prevent packets to pass through the firewall unless they match the established rule set (i.e., configurations) defined by network administrators. Though many modern firewalls are more sophisticated, the network layer firewalls cannot filter undesired traffic, such as malware payload, that utilizes legitimate IP addresses and ports. Application layer firewall operates by monitoring and potentially blocking the input, output, or system service calls which do not meet the configured policy of the network layer firewall. A proxy server may act as a firewall by responding to the input packets (for example, connection requests) in the manner of an application while blocking other packets. Both application layer firewall and proxies make tampering with an internal system more difficult. But with the increased capability and sophistication, attackers today have devised more advanced attack methods to pass malicious packets to a target network. For example, intruders may hijack a publicly-reachable system and use it as a proxy for their own purposes. Using the intercepted proxy, the intruder creates the packets with a forged IP address with the purpose of concealing the identity of the sender or impersonating another computing system.

The intrusion detection systems filter any suspicious or anomalous activity over the network [19,47]. These detect systems are valuable in a way that they seek to detect the early stages of an attack (e.g., an attacker's probing of a machine or network for specific vulnerabilities) and can then aid in protecting a machine from the subsequent stages of the attack. Also, these systems seek to detect telltale signs of suspicious activities or patterns of behavior whether by a user, an application, or a piece of malicious code that firewalls or other protection tools might miss or ignore. Many detect system variants exist to identify malicious network payloads. Such detections are either signature-based or anomaly-based. In signature-based, the detection system recognizes attack packets due to their well-known fingerprints or signatures as those packets cross the network's gateway threshold. In anomaly-based, the detection system has no prior knowledge of what bad packets are. The detection system determines what normal traffic is by examining the pattern, often in real-time, and reports abnormal traffic behaviors based on the analysis on the pattern. The signature-based detection system has been considered ineffective as the proliferation and sophistication of malware writer have improved in recent years [16,17]. It is considered that it is almost impossible to catch up ever evolving malware signature with pattern recognition methods popularly used in the signature-based approach. Proposing advanced anomaly-based detections have been an active research area [133]. In this method, the system learns by example (self-learning) what constitutes normal by observing traffic for an extended period of time and building some model of the underlying process. The process is evolved (self-adaptive) as the signature of malware evolves.

Rather than focusing on fixing specific aspects of firewalls and IDS, more general approaches to understand the network attack patterns are needed in order to devise better mechanisms to thwart undesired traffic coming from external sources. The area of network forensic involves the study of monitoring and analysis of network traffic by eavesdropping to Ethernet, TCP/IP, or the Internet including web browser, email, newsgroup, synchronous chat, and peer-to-peer traffic [49–51]. The evidences are used for legal action or to understand the network traffic attack patterns. eMailTrackerPro [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 SmartWhoIs [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 are used to research the threats organizations face and to learn how to better protect against those threats. Virtualization techniques are often employed to host multiple honeypots on a single physical machine. Therefore, even if the honeypot is compromised, there is a chance for quicker recovery with less expense. A large scale honeypots, such as honeynet which connects two or more honeypots on a network, is used for monitoring a larger and more diverse network. These honeynets are often implemented as parts of larger network intrusion detection systems. A honeyfarm is a centralized collection of honeypots and analysis tools [54].

As it is not possible to give uniform access to resources, access control mechanisms have been used to enable an authority to control access to only certain resources. In the access control, the entities that can perform actions in the system

are called "subjects" and the entities representing resources to which access may need to be controlled are called "objects". In the capability based access control, a subject is granted to access an object if the subject holds a reference or capability. For example, if a user provides a correct userID and password, the user is granted to view his/her bank statement. Access is conveyed to another party by transmitting such a capability over a secure channel. For example, a certificate is created for the user to present it for the verification purpose. In an access control list based approach, a subject's access to an object depends on whether its identity is on a list associated with the object. For example, if Alice is on the list of doctors, she is granted to view patient's records. Access is conveyed by editing the list. For example, when Alice leaves the hospital, she is no longer on the list of the doctors and won't be able to view the patient's records. The three most widely recognized models are Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role Based Access Control (RBAC). In DAC, the owner decides who is allowed to access certain objects and what privileges they have. In MAC approach, it is the operating system constrains the ability of subjects (e.g. process or thread) to access or perform operation on objects (e.g. files, directories, TCP/UDP ports, or shared memory segments), either by a rule that defines specific conditions or by a mathematical structure that defines greatest lower-bound and least upper-bound values. RBAC is a newer alternative approach to MAC and DAC which restricts the system access only to authorized users. It is used by the majority of enterprises and most IT vendors offer RBAC in one or more products.

Traditional Access control systems provide the essential services such as authentication, authorization, and accountability. Authentication and Authorization is the process of verifying that a subject is bound to an object. Traditional authentication and authorization mechanisms use three different factors to identity a subject to verify if the subject has a right capability to access the object. First factor is something you know, for example, password or a personal identification number (PIN). This assumes that only the owner of the account who knows the password or PIN needed to access the account. Second factor is something you have which includes smart card or security token. This assumes that only the owner of the account has the necessary smart card or token needed to unlock the account. Third factor is something you are, such as fingerprint, voice, or iris characteristics. The current trend in authentication is layered approach, often called as strong authentication, replying on the presentation of two or more authentication factors [55]. A number of different types of pocket-sized authentication tokens have been proposed. These tokens contain a cryptographic key in tamper-resistant storage. Taking advantage of ubiquitous nature of today's computer with USB ports, USB-based tokens have also proposed either simply as a storage of a X.509 certificate or often running challenge/response protocol [56,13,86]. To take advantage of ever fast growing population of mobile users, a number of authentication mechanisms targeting mobile users have been proposed [56–58]. Biometric technology has been used in limited applications. Some PC and workstations have become more sophisticated with audio-visual interfaces, there is renewed interest in employing biometric authentication technology in the network environment [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].

4. Emerging threats

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

4.1. 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 sites have been very popular and become the preferred method of communication for most young generations. Each of these social media websites typically provide tools where users share their personal information (i.e. name, address, gender, date of birth, preference in music and movie), photos, stories and disseminate links.

Attackers are taking advantage of the social media craze as a new medium for launching insidious attacks. By the end of 2008, the Kaspersky Lab collection contained more than 43,000 malicious files relating to social media sites [109]. A report published by IT security and data protection firm Sophos has revealed an alarming rise in attacks on users of social media websites. According to their report [111], around 60% of the users in the social networks have received spam. Due to the unlimited access to the profile of users, attackers can further gain the information of corporation and commercial secrets. In the survey conducted by Sophos [111], around 60% companies concern that their employees provide too much information in social networks while around 66% companies think that using social networks pose a great threat to the companies.

Koobface worm [110] that spreads through social media sites in 2009 is notably the best known malware case that utilizes the proliferation of social media sites. Leveraging its zombie arsenal, the Koobface botnet automates the creation of new social media accounts used to befriend unsuspecting users, in turn spamming enticing links that redirect to malware. Victims that fall prey to the social engineering attacks witness their own social networking accounts turn into vehicles for sending spam to the victim's friends, while the victim's machine is repurposed into a zombie. Thomas and Nicol [110] constructed a zombie emulator which was able to infiltrate the Koobface botnet and identified fraudulent and compromised social network accounts used to distribute malicious links to over 213,000 social network users generating over 157,000 clicks. They discovered the ineffectiveness of current blacklisting services offered by social network operators to filter malicious virus through most prominent blacklisting services. They argued that those blacklisting services only recognize 27% of threats and take on average 4 days to respond while they found that 81% of visitors to Koobface's spam occur within the first 2 days of a link being posted, leaving the majority of social networking users vulnerable. Another popular malware attack is done by the use a significant number of Twitter or Facebook accounts that are not legitimate or not in use. Cyber criminals are becoming a lot more sophisticated in their efforts to appear as trustworthy users. Then the criminals trick users in the social network site into "friendling" or following them and clicking on their status updates which often lead to malicious web sites. In another study [109], it is illustrated that a large number of malware were spread after clicking for content on "trending" topics via Twitter. Understanding the social network platforms and simulations to spread malware using mock up services over Facebook has also been studied in [91].

Social networking sites also have raised the stakes for privacy protection because of the centralization of massive amounts of user data, the intimacy of personal information collected, and the availability of up-to-date data which is consistently tagged and formatted [109]. This makes social networking sites an attractive target for a variety of organizations seeking to aggregate large amounts of user data, some for legitimate purposes and some for malicious ones. In most cases, extracting data violates users' expectation of privacy. Protecting user's private data kept in the social networking service providers has been explored. Lucas et al [112] proposed a Facebook application for encrypting and decrypting sensitive data using client-side JavaScript. This architecture ensures that data never arrives at the social network service providers in an unencrypted form preventing them from observing and accumulating the information that users transmit through the network. Privacy awareness related issues and tools which can help users to set their privacy setting more intuitively have been proposed as well. For example, Fang and LeFevre [113] proposed privacy wizard. The wizard iteratively asks the user to assign privacy "labels" to selected friends, and it uses this input to construct a classifier, using a machine learning model, which can in turn be used to automatically assign privileges to the rest of the user's friends. The intuition for the design comes from the observation that real users conceive their privacy preferences of which friends should be able to see which information, based on implicit set of rules they set and repeatedly use in most friends setting.

4.2. Cloud computing

The efficiencies of moving data and applications to the cloud continue to attract consumers who store their data in DropBox and iCloud, use Gmail and Live mail to handle email, and track their lives using services such as Evernote and Mint.com. Cloud computing is arguably one of the most significant technological shifts in recent times [16]. The mere idea of being able to use computing in a similar manner to using a utility is revolutionizing the IT services world and holds great potential. Customers, whether large enterprises or small businesses, are drawn towards the cloud's promises of agility,

reduced capital costs, and enhanced IT resources. IT companies are shifting from providing their own IT infrastructure to utilizing the computation services provided by the cloud for their information technology needs [66].

Cloud computing provides unique characteristics that are different from the traditional approaches. The five key characteristics of cloud computing include on-demand self-service, ubiquitous network access, location-independent resource pooling, rapid elasticity, and measured service, all of which are geared towards using clouds seamlessly and transparently [67]. Resource polling refers to the ability where no resources are dedicated to one user but instead are pooled together to serve multiple consumers. Resources, whether at the application, host or network levels, are assigned and reassigned as needed to these consumers. On demand self service refers where the users can assign themselves additional resources such as storage or processing power automatically without human intervention. This is comparable with autonomic computing where the computer system is capable of self management. Along with self provisioning of resources, cloud computing is characterized with the ability to locate and release resources as rapidly as needed, the term often called as "elasticity". This allows consumers to scale up the resources they need at any time to address heavy loads and usage spikes, and then scale down by returning the resources to the pool when finished [66]. Measured service, also often called as pay as you go, enables the cloud to be offered as a utility where users pay on a consumption basis, much the same way it is done to pay utilities like electricity, gas and water.

Cloud computing is also a model of integration that delivers various resources to clients at different layers of the system and utilizes different resources. Generally speaking, the architecture of a cloud computing environment can be divided into 4 layers: the hardware layer (including data centers), the infrastructure layer, the platform layer and the application layer [68].

- *The hardware layer*: This layer is responsible for managing the physical resources of the cloud, including physical servers, routers, switches, power and cooling systems. In practice, the hardware layer is typically implemented in data centers. A data center usually contains thousands of servers that are organized in racks and interconnected through switches, routers or other fabrics. Typical issues at hardware layer include hardware configuration, fault tolerance, traffic management, power and cooling resource management.
- *The infrastructure layer*: This layer is also known as the virtualization layer. The infrastructure layer creates a pool of storage and computing resources by partitioning the physical resources using virtualization technologies such as Xen, Kernel based Virtual Machine and VMware. The infrastructure layer is an essential component of cloud computing, since many key features, such as dynamic resource assignment, are only made available through virtualization technologies.
- *The platform layer*: Built on top of the infrastructure layer, the platform layer consists of operating systems and application frameworks. The purpose of the platform layer is to minimize the burden of deploying applications directly into VM containers. For example, Google App Engine operates at the platform layer to provide API support for implementing storage, database and business logic of typical web applications.
- *The application layer*: At the highest level of the hierarchy, the application layer consists of the actual cloud applications. Different from traditional applications, cloud applications can leverage the automatic-scaling feature to achieve better performance, availability and lower operating cost.

However, in practice, clouds offer services that can be grouped into three categories: software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS) [68]. Applications running on or being developed for cloud computing platforms pose various security and privacy challenges depending on the underlying delivery and deployment models. In IaaS, the cloud provider supplies a set of virtualized infrastructural components such as virtual machines (VMs) and storage on which customers can build and run applications. The application will eventually reside on the VM and the virtual operating system. PaaS enables programming environments to access and utilize additional application building blocks. Such programming environments have a visible impact on the application architecture, such as constraints on which the application can request services from an OS. Finally, in SaaS, the cloud providers enable and provide application software as on-demand services.

Multi-tenancy is a feature unique to clouds which allows cloud providers to manage resource utilization more efficiently by partitioning a virtualized, shared infrastructure among various customers. For example, to isolate multiple tenants' data, Salesforce.com employs a query rewriter at the database level, whereas Amazon uses hypervisors at the hardware level. Virtualization is an important enabling technology in this area that helps abstract infrastructure and resources to be made available to clients as isolated VMs. Providing strong isolation, mediated sharing, and secure communications between VMs are active research areas. Using a flexible access control mechanism that governs the control and sharing capabilities of VMs within a cloud host has been suggested as a potential solution [67]. Because clients acquire and use software components from different providers, crucial issues include securely composing them and ensuring that information handled by these composed services is well protected [67]. For example, a PaaS environment might limit access to well-defined parts of the file system, thus requiring a fine-grained authorization service.

Trust management and policy integration is an active area of research in cloud computing as the outsourcing model of the cloud, where the cloud providers control and manage user's data and services, forces the clients to have significant trust in their provider's technical competence [67]. In cloud computing environments, the interactions between different service domains driven by service requirements are also dynamic, transient, and intensive. Thus, a development of trust framework has been proposed to allow efficient capturing of a generic set of parameters required for establishing trust and to manage evolving trust and interaction/sharing requirements [115,116]. The cloud's policy integration is another active

area of research to address challenges such as semantic heterogeneity, secure interoperability, and policy-evolution management [117]. Furthermore, customers' behaviors can evolve rapidly, thereby affecting established trust values. This suggests a need for an integrated, trust-based, secure interoperation framework that helps to establish, negotiate, and maintain trust to adaptively support policy integration [114,117].

4.3. Smartphones

Smartphones, coupled with improvement in wireless technologies, have become an increasingly sophisticated computer and communication device that is readily carried by individuals throughout the day. The convergence of increasing computing power, personalization and mobility makes them an attractive means of planning and organizing work and private life of individuals. According to [16], the sheer volume of mobile phone users around the world indicates a current need for proactive mobile security measures. It is assumed that over 4.5 billion use a cell phone every day and an estimated 2 billion smartphone will be deployed by 2013.

Going beyond the simple SMS messaging, increasing level of sensitive information is stored in the smartphones. Within companies, these technologies are causing profound changes in the organization of information systems and therefore have become the source of new risks. As smartphones collect and compile increasing amount of sensitive information, the access must be controlled to protect the privacy of the user and the intellectual property of the company.

These staggering growths in mobile technology have created an attractive target for cyber criminal. Security concerns in mobile are different from the traditional security problems in PC and enterprise computing due to their embedded nature and different operational environment. Mulliner [119] listed the following features 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 tempered 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 a number of different styles of attacks targeted to take advantage of the proliferation of mobile computing. Communication related attacks are derived from flaws in the design and management of mobile communication infrastructure. The attacker may try to break the encryption of the mobile network. The GSM (Global System for Mobile Communication) network today uses two variants of algorithms known as A5/1 and A5/2, latter being known to be weaker. Since the encryption algorithm was made public, it was proved that it is possible to break the encryption in about 6 hours [118]. An attacker can try to eavesdrop on Wi-Fi communications to derive information (e.g. username, password). These types of attacks are not unique to smartphones, but they are very vulnerable to these attacks because very often the Wi-Fi is the only means of communication they have to access the Internet. Security issues related to Bluetooth on mobile devices have been studied and have shown a number of problems. For example, Cabir is a worm that spreads via Bluetooth connection [120]. The worm searches for nearby phones with Bluetooth in discoverable mode and sends itself to the target device. The user must accept the incoming file and install the program. After installing, the worm infects the machine. To prevent communication related attacks, network traffic exchanged by phones can be monitored such as surveillance on network routing points or monitoring the use of network mobile protocols.

Another type of attacks is derived from the vulnerabilities in mobile software applications especially exploiting mobile web browser. Just as common Web browsers, mobile web browsers are extended from pure web navigation with widgets and plug-ins which many attackers use as means to spread malware through. Jailbreaking the iPhone was based entirely on vulnerabilities on the web browser [121] based on a stack-based buffer overflow in a library used by the web browser. Vulnerability in the web browser for Android was discovered in October 2008 exploiting obsolete and vulnerable library [121].

Malicious attackers target mobiles phones as a medium to spread malware [106]. Both Georgia Tech emerging cybersecurity threats reports [16] and Symantec threats reports [17] in last couple of years warn the growing number of malware that are specifically created for mobile phones such as targeting Google Android based phones and Apple iPhones. To control the malware propagation, mobile companies offer a centralized public market place complimented with an approval process before hosting the application. The centralized marketplace helps to remove any application if found suspicious before they are downloaded by the users. For example, Apple adopts a vetting process to ensure all applications conform to Apple's rules before they can be offered via the App Store. Apple approves an application by code signing with encryption keys. Accessing the applications via App store is the only way for iPhone devices to install applications. Similar to Apple, Android too has a public marketplace to host applications. However, unlike Apple, the Android application can be self-signed. Android uses crowd sourcing to rate the applications by users. Based on user complaints, applications can be removed from marketplace and remove them from the device as well. Another approach taken by the mobile companies

to protect their mobile platforms found in the idea of a 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.

4.4. Critical infrastructure

The critical infrastructure systems that form the lifeline of a modern society and their reliable and secure operation are of paramount importance to national security and economic vitality. In most sense, the cyber system forms the backbone of a nation's critical infrastructures, which means that a major security incident on cyber systems could have significant impacts on the reliable and safe operations of the physical systems that rely on it. The recent findings, as documented in government reports [15], indicate the growing threat of physical and cyber-based attacks in numbers and sophistication on electric grids and other critical infrastructure systems. Cybersecurity related to critical infrastructure seeks to limit vulnerabilities of these structures and systems to [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.

Critical infrastructure protection is harder to address than information and communication technology (ICT) protection because of these infrastructures' interconnection complexity, which can lead to different kinds of problems [78]. Consider the power grid, in which geographically dispersed production sites distribute power through different voltage level stations (from higher to lower voltage) until energy eventually flows into our houses. Both the production and distribution sites are typically controlled by supervisory control and data-acquisition (SCADA) systems, which are remotely connected to supervision centers and to the corporate networks (intranets) of the companies managing the infrastructures. The intranets are linked to the Internet to facilitate, for example, communication with power regulators and end clients. These links create a path for external attackers. Operators' access SCADA systems remotely for maintenance operations, and sometimes equipment suppliers keep links to the systems through modems. The prevalence of proprietary solutions and use of older versions plagued with vulnerabilities are sought to add another dimension to propose solutions to protect nation's crucial infrastructure.

As the research into the critical system is quite new, researchers are still trying to understand the nature of critical infrastructure systems. This includes understanding criticality in system, understanding interdependencies among systems and infrastructures, and identifying and quantifying consequences of attacks on the critical systems. Because of the tight dependency of these systems and millions of users in their daily life, it is important that the critical infrastructure operates on $24 * 7$ bases without any downturn. Self-diagnostic techniques using heartbeats, challenge-response, built-in monitoring of critical functions and detection of process anomalies which can capture any signs of non operative functions have been proposed [79,77]. Another relevant topic of interest is the development of self-healing systems to pursue automated and coordinated attack response and recovery [77].

4.5. Other emerging areas of concern

Cybersecurity in embedded systems and sensors are the topics that have received an increasing amount of attention from industry and academia in recent years due to their increased use in every facet in our lives. For example, embedded small devices inserted in cars, home appliances, mobile phone, and audio/video equipments, increasingly become a part of our lives. Similarly, sensors are seeing broader research and commercial deployments in military, scientific, and commercial applications including monitoring of biological habitats, agriculture, and industrial processes. Security concerns in these areas are different from the traditional security problems in PC and enterprise computing due to their different embedded nature and operational environment [72,73]. Embedded systems and sensors are often highly cost sensitive requiring them to use smaller processors which have limited room for security overhead for example storing a big cryptography key. Therefore, the most enterprise security solutions do not work in the embedded system world. Embedded systems and sensors are resource constrained in energy, memory, computational speed and communications bandwidth due to the nature of small size. They have a very weak physical trust boundary. For example, they are installed in residents and commercial properties, outside fields, or carried by human in their hands or pockets which enables many different physical-oriented attacks. They use an intimate connection between hardware and software often without the shielding of an operating system. The different

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 wire tapping, have become another source of growing concern in the field.

Cyber warfare refers to politically motivated hacking to conduct sabotage and espionage. In the book Cyber Ware [92], cyber warfare was defined as "actions by a nation-state to penetrate another nation's computers or networks for the purposes of causing damage or disruption". Most cyber warfare concerns are focused on national security breaches and sabotage of nation's critical infrastructure [76]. The former case concerns with the international espionage where classified information that may breach the national security is illegally accessed or altered by unauthorized people. The latter case concerns with any potential disruption of nation's critical infrastructure such as power grid system and transportation system. In 2008, a simulated exercise codenamed "cyber storm" was conducted by the Department of Homeland Security. The purpose of the exercise was to test the nation's defense against digital espionage. The Cyber Storm exercise highlighted the gaps and shortcomings of the nation's cyber defenses. Since then, researchers have proposed a number of new priorities in regard to nation's cyber defense strategy [77,78]. Identification of Nation's Critical Systems has been proposed [80] to recognize the Internet-enabled systems that are critical to nation's cyber defense and any interdependencies among the systems. A number of strategies to protect nation's critical infrastructures by vulnerability identification and remediation, and threats mitigation and response have been proposed [79,77].

4.6. 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]. Smishing, 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 [81,89,90] 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 cybersecurity 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 botmaster 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 testbeds 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 honeynet techniques has been suggested for the testbeds [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.

5. Future research direction

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

Traditionally the practice in the cybersecurity community has been based on incremental patches which rectify the current security and privacy issues and then moves onto next step. Some believe that this incremental approach has not worked well and will not be able to accommodate future needs since the original Internet was invented for a very different environment than how it is used today. An approach to think “outside box” without relying on the current computing system and the Internet but starting something afresh has been suggested to make a better use of the fast growing demands of the Internet [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 traceback techniques have become an active area of research as a strategic plan to thwart increasing number of cyber attackers in the future, especially when the critical infrastructure is involved. We delve into more detailed of these speculated future research directions in the following sections. The summary of the research questions and future research directions is illustrated in Fig. 5.

5.1. Focus on privacy

In recent years, privacy has become a critical issue in the development of IT systems with the widespread of networked systems and the Internet. Now, the Internet is used in all wakes of our lives demanding increasing volume of personal information to be entered in the cyberspace. According to JP Morgan's annual report [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.

5.2. Next generation secure internet

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

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

A new paradigm of architectural design described as "clean-slate design" has been suggested [123,124]. The theme of "clean-slate design" is to design the system from scratch without being restrained by the existing system, providing a chance to have an unbiased look at the problem space [142]. However, the scale of the current Internet forbids any changes, and it is extremely difficult to convince the stakeholders to believe in a clean-slate design and adopt it. There is simply too much risk involved in the process. The only way to mitigate such risks and to appeal to stakeholders is through actual Internet-scale validation of such designs that show their superiority over the existing systems [123]. Despite the risk, research funding agencies all over the world have realized this pressing need and a world-wide effort to develop the next generation Internet is being carried out [123,124]. The National Science Foundation (NSF) was among the first to announce a GENI (Global Environment for Networking Innovations) program for developing an infrastructure for developing and testing futuristic networking ideas developed as part of its FIND (Future Internet Design) program. The NSF effort was followed by the FIRE (Future Internet Research and Experimentation) program which support numerous next generation networking projects under the 7th Framework Program of the European Union, the AKARI program in Japan, and several other similarly specialized programs in China, Australia, Korea, and other parts of the world.

The "clean state design" idea can be approached in a number of areas. In the area of Internet security aspect, security mechanisms are placed as an additional overlay on top of the original architecture rather than as part of the Internet architecture. This includes proposals and projects related to security policies, trust relationships, names and identities, cryp-

tography, anti-spam, anti-attacks, and privacy. Concerning on new mechanisms for content delivery over the Internet as the next generation Internet is set to see a huge growth in the amount of content delivered over the Internet, newer paradigms for networking with content delivery at the center of the architecture is proposed [123] rather than connectivity between hosts, as in the current architecture. Challenged network research focuses specifically on heterogeneous networking environments where continuous end-to-end connectivity cannot be assumed such as seen in the wireless ad hoc networks. The discussions in this area relate to two important perspectives of the future Internet design requirements: Energy efficient protocol design and implementation and federation of heterogeneous networking environments. Another area is the management and control framework. The current Internet works on a retro-fitted management and control framework that does not provide efficient management and troubleshooting. The proposals for the future Internet in this area vary from completely centralized ideas of management to more scalable and distributed ideas.

5.3. Towards trustworthy systems

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

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

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

5.4. Global-scale identity management and traceback techniques

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

There are many current approaches to identity management. For example, many websites employ logging in process with username and password combination to screen only eligible users to enter into the service. However, many of these are not yet fully interoperable with other services across different organizations and scalable. They are only for single-use or limited in other ways. It has been pointed out [15] that due to the lack of adequate identity management it is often extremely difficult to trace identity theft.

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

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 traceback technique. A mark, typically an IP address or the edges of the path that the packet traversed to reach the router, is inserted into a packet and then used to trace the source of the attack. However, it is criticized that most current traceback methods only work well for a single cooperative defense and skilled attackers easily evade most currently deployed traceback systems by tweaking the header IP addresses [96,15]. The development of

global scale traceback system with a defense mechanism which can trace and block evolving packet signatures are listed as solutions required for the future computing environment.

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

5.5. Usable security

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

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

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

6. 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 traceback 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: J.R. Goodall, G.J. Conti, K.-L. Ma (Eds.), VizSEC, in: Lect. Notes Comput. Sci., vol. 5210, Springer, 2008, pp. 111–118.
[54] M. Vrable, 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.
[55] H.K. Lu, A. Ali, Communication security between a computer and hardware token, in: ICONS 2008, pp. 220–225.
[56] 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.
[57] 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.
[58] D. de Borde, Two-factor authentication, Siemens Enterprise Communications UK-Security Solutions, available at http://www.insight.co.uk/files/whitepapers/Twofactor authentication (White paper).pdf, 2008.
[59] J. Bringer, H. Chabanne, An authentication protocol with encrypted biometric data, in: AFRICACRYPT, in: Lect. Notes Comput. Sci., 2008, pp. 109–124.
[60] 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.
[61] G. Coker, J. Guttman, P. Loscocco, J. Sheehy, B. Sniffen, Attestation: Evidence and trust, in: ICICS '08, 2008, pp. 1–18.
[62] J. Jang, H. Hwang, S. Nepal, Biometric enabled portable trusted computing platform, in: TurstCom 2011, pp. 436–442.
[63] 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.
[64] 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.
[65] P. Guarda, N. Zannone, Towards the Development of Privacy-Aware Systems, Information and Software Technology, 2008.
[66] K. Dahbur, B. Mohammad, A.B. Tarakji, A survey of risks, threats and vulnerabilities in cloud computing, 2011, pp. 12–18.
[67] H. Takabi, J. Joshi, G. Ahn, Security and privacy challenges in cloud computing environments, IEEE Secur. Priv. (2010) 24–31.
[68] Q. Zhang, L. Cheng, R. Boutaba, Cloud computing: state-of-the-art and research challenges, J. Internet Serv. Appl. 1 (2010) 7–18.
[69] 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.
[70] C. Soghoian, A remote vulnerability in Firefox extensions, http://paranoia.dubfire.net/2007/05/remote-vulnerability-in-firefox.html, last accessed: June 2013.
[71] C. Reis, A. Barth, C. Pizano, Browser security: lessons from google chrome, Commun. ACM 52 (2009) 45–49.
[72] P. Koopman, Embedded system security, IEEE Comput. 37 (7) (2004) 95–97.
[73] 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.
[74] 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.
[75] Y. Zhou, Y. Fang, Y. Zhang, Securing wireless sensor networks: a survey, IEEE Commun. Surv. Tutor. 10 (3) (2008) 6–28.
[76] 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.
[77] C.W. Ten, Cybersecurity for critical infrastructures: Attack and defense modeling, IEEE Trans. Syst. Man Cybern. 40 (4) (2010) 853–865.
[78] 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.
[79] 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.
[80] 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.
[81] S. Abraham, I.S. Chengalur-Smith, An overview of social engineering malware: Trends, tactics, and implications, Technol. Soc. 32 (2010) 183–196.
[82] M. Feily, A. Shaherestani, S. Ramadass, A survey of botnet and botnet detection, in: SECURWARE 2009, pp. 268–273.
[83] M. Bailey, E. Cooke, F. Jahanian, et al., A survey of botnet technology and defense, in: CATCH 2009, pp. 299–304.
[84] Z. Zhu, G. Lu, Y. Chen, et al., Botnet research survey, in: COMPSAC 2008, pp. 967–972.
[85] C. LI, W. Jiang, X. Zou, Botnet: survey and case study, in: ICICIC 2009, pp. 1184–1187.
[86] S. Nepal, J. Zic, D. Liu, J. Jang, Trusted computing platform in your pocket, in: ECU 2010, pp. 812–817,.
[87] R. Sailer, X. Zhang, T. Jaeger, L. van Doorn, Design and implementation of a TCG-based integrity measurement architecture, SSYM 2004,.
[88] M. Johnson, S. Egelman, S. Bellovin, Facebook and privacy: it's complicated, in: Proc. SOUPS 2012, ACM, 2012.
[89] 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.
[90] 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.
[91] G. Hogben, Security issues and recommendations for online social networks, in: Position Paper. ENISA, European Network and Information Security Agency, 2007.
[92] R.C. Clark, Cyber War, Ecco, 2010.
[93] D. Weitzner, et al., Information accountability, Commun. ACM 51 (6) (2008) 82–87.
[94] J. Yao, S. Chen, C. Wang, Accountability as a service for the cloud, in: SCC 2010, pp. 81–88.
[95] 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.
[96] J. Wang, J.N. Whitley, R.C.W. Phan, Unified parametrizable attack tree, J. Inform. Security Res. (IJISR) 1 (1) (2011) 20–26.
[97] A. John, T. Sivakumar, Ddos: Survey of traceback methods, in: IJRTE, 2009.
[98] 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.
[99] L.F. Cranor, S. Garfinkel (Eds.), Security and Usability: Designing Secure Systems that People Can Use, O'Reilly Media, 2005.
[100] L.F. Cranor, S. Garfinkel, Secure or usable?, IEEE Secur. Priv. 2 (2004) 16–18.

[101] Anti-phishing group tech reports: http://www.antiphishing.org/phishReportsArchive.html, last accessed: June 2013.
[102] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G.M. Voelker, V. Paxson, S. Savage, Spamalytics: an empirical analysis of spam marketing conversion, in: CCS, 2008, pp. 3–14.
[103] H. Shahriar, M. Zulkernine, Mitigating program security vulnerabilities: Approaches and challenges, ACM Comput. Surv. 44 (3) (2012), Article No. 11.
[104] S. Liu, B. Cheng, Cyberattacks: "Why, what, who and how", in: IT Pro., IEEE Computer Society, May/June 2009.
[105] http://www.pcmag.com/article2/0,2817,2392570,00.asp, last accessed: June 2013.
[106] http://www.cert.org/cybersecurity-engineering/, last accessed: June 2013.
[107] http://www.welivesecurity.com/2012/12/11/trends-for-2013-astounding-growth-of-mobile-malware/, last accessed: June 2013.
[108] http://www.huffingtonpost.com/brian-honigman/100-fascinating-social-me_b_2185281.html, last accessed: June 2013.
[109] W. Luo, J. Liu, J. Liu, C. Fan, An analysis of security in social networks, in: Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2009, pp. 648–651.
[110] K. Thomas, D.M. Nicol, The Koobface botnet and the rise of social malware, in: Proceedings of the 5th International Conference on Malicious and Unwanted Software (Malware 2010), 2010, pp. 63–70.
[111] Sophos security threat reports: http://www.sophos.com/en-us/security-news-trends/reports/security-threat-report.aspx.
[112] M. Lucas, N. Borisov, Flybynight: Mitigating the privacy risks of social networking, in: WPES, 2008.
[113] L. Fang, K. LeFevre, Privacy wizards for social networking sites, in: Proc. WWW '10, pp. 351–360.
[114] Nicolas Carr, The Big Switch: Our New Digital Destiny, W.W. Norton & Co., 2008.
[115] Y. Zhang, J. Joshi, Access control and trust management for emerging multidomain environments, in: S. Upadhyaya, R.O. Rao (Eds.), Annals of Emerging Research in Information Assurance, Security and Privacy Services, Emerald Group Publishing, 2009, pp. 421–452.
[116] D. Shin, G.-J. Ahn, Role-based privilege and trust management, Comput. Syst. Sci. Eng. 20 (6) (2005) 401–410.
[117] Q. Zhang, L. Cheng, R. Boutaba, Cloud computing: state-of-the-art and research challenges, J. Internet Serv. Appl. 1 (2010) 7–18.
[118] Timo Gendrullis, A real-world attack breaking A5/1 within hours, in: Proceedings of CHES '08, Springer, November 2008, pp. 266–282.
[119] C.R. Mulliner, Security of smart phones, Master's thesis submitted to University of California, Santa Barbara, June 2006.
[120] Sampo Töyssy, Marko Helenius, About malicious software in smartphones, J. Comput. Virology (Springer Paris) 2 (2) (2006) 109–119, http://dx.doi.org/10.1007/s11416-006-0022-0, retrieved 2010-11-30.
[121] Ken Dunham, Saeed Abu Nimeh, Michael Becher, Mobile Malware Attack and Defense, Syngress Media, ISBN 978-1-59749-298-0, 2008.
[122] http://resources.pbecomm.com/global-ecommerce-blog/understanding-the-global-ecommerce-market-blog/goldman-sachs-forecasts-growth-rate-of-global-e-commerce-sales-asia-factors-big/, last accessed: June 2013.
[123] 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.
[124] A. Feldmann, Internet clean-slate design: What and why?, Comput. Commun. Rev. 37 (3) (2007) 59–64.
[125] 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.
[126] 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.
[127] S. Furnell, Making security usable: Are things improving?, Comput. Secur. 26 (6) (2007) 434–443.
[128] B.D. Payne, W.K. Edwards, A brief introduction to usable security, IEEE Internet Comput. 12 (2008) 13–21.
[129] E.E. Schultz, Where have the worms and viruses gone? New trends in malware, Comput. Fraud Secur. 2006 (7) (2006) 4–8.
[130] 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.
[131] 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.
[132] http://gocsi.com/sites/default/files/uploads/2007_CSI_Survey_full-color_no marks.indd_.pdf, last accessed: June 2013.
[133] P. García-Teodoroa, J. Díaz-Verdejoa, G. Maciá-Fernándeza, E. Vázquez, Anomaly-based network intrusion detection: Techniques, systems and challenges, Comput. Secur. 28 (1–2) (2009) 18–28.
[134] W. Hasselbring, R. Reussner, Toward trustworthy software systems, Computer 39 (4) (2006) 91–92, http://dx.doi.org/10.1109/MC.2006.142.
[135] L.M.R. Gadelha Jr., B. Clifford, M. Mattoso, M. Wilde, Provenance management in Swift, Future Gener. Comput. Syst. 27 (6) (2011) 775–780.
[136] L. Moreau, J. Freire, J. Futrelle, R.E. McGrath, J. Myers, P. Poulson, The open provenance model, http://openprovenance.org/, last accessed: June 2013.
[137] 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.
[138] N.R. Mathiasen, S. Bodker, Experiencing security in interaction design, in: Proc. CHI 2011, 2011, pp. 2325–2334.
[139] L. Barkhuus, The mismeasurement of privacy: using contextual integrity to reconsider privacy in HCI, in: Proc. CHI 2012, 2012, pp. 367–376.
[140] 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.
[141] M. Madden, Privacy management on social media sites, http://pewinternet.org/Reports/2012/Privacy-management-on-social-media.aspx, February 2012.
[142] J. Pan, S. Paul, R. Jain, A survey of the research on future Internet architectures, IEEE Commun. Mag. 49 (7) (2011) 26–36.
[143] H.P. Bui, J. Cox, S. Theobald, S. Wiegand, Trustworthy IT systems, trustworthy.googlecode.com/svn-history/r27/trunk/tex/doc.pdf, 2012.
[144] 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.
[145] T. Eze, R. Anthony, C. Walshaw, A. Soper, A new architecture for trustworthy autonomic systems, in: EMERGING 2012, 2012, pp. 62–68.
[146] 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.
[147] R. Maas, E. Maehle, Applying the organic robot control architecture ORCA to cyber-physical systems, in: Proc. SEAA, 2012, pp. 250–257.
[148] 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.
[149] 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.
[150] J. Jensen, Federated identity management challenges, in: Proc. ARES, 2012, pp. 230–235.
[151] 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.
[152] 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.