Mapping the Most Significant Computer Hacking Events to a Temporal Computer Attack Model

Renier van Heerden[1,2], Heloise Pieterse[1] and Barry Irwin[2]

[1]Council for Scientific and Industrial Research, Pretoria, South Africa
rvheerden@csir.co.za, hpieterse@csir.co.za
[2]Rhodes University, Grahamstown, South Africa
b.irwin@ru.ac.za

Abstract. This paper presents 18 of the most significant computer hacking events (also known as computer attacks). These events were selected because of their unique impact, methodology, or other properties. A temporal computer attack model is presented that can be used to model computer based attacks. This model consists of the following stages: Target Identification, Reconnaissance, Attack, and Post-Attack Reconnaissance stages. The Attack stage is separated into: Ramp-up, Damage and Residue. This paper demonstrates how our 18 significant hacking events are mapped to the temporal computer attack model. The temporal computer attack model becomes a valuable asset in the protection of critical infrastructure by being able to detect similar attacks earlier.

Keywords: computer attack model, ontology, network attack prediction

1 Introduction

Computer hacking (also referred as computer cracking) developed in conjunction with the normal usage of computer systems. This paper discusses some of the most significant hacking events and the features that made them unique. The events listed are considered to be significant because of their unique impact, methodology or other properties. The level of significance is an abstract and relative measure. Other attempts to judge the importance of hacking events have been made by Heater [1], Hall [2] and Julian [3].

Research in computer network attack prediction at the Counsel for Scientific and Industrial Research (CSIR) in South Africa has resulted in the development of a Taxonomy and Ontology of computer network attacks. A temporal attack model was developed with the goal of separating the different stages of a computer network attack.

The model consists of the following basic stages: Target Identification; Reconnaissance; Attack; and Post Attack. The Attack stage has the following sub-stages: Ramp-up; Damage; and Residue.

Research was also organized into strategies for identifying the Reconnaissance and Ramp-up stages. The attack model is a valuable asset in the protection of critical in-

adfa, p. 1, 2011.

frastructure as it has the ability to identify attacks at an earlier stage and so improve the responsiveness to incidents.

This paper presents the authors' version of the most important hacking events, and cannot in itself be considered absolute. We chose events based on either the uniqueness of the technique used or their unique impact.

The attack model is presented in more detail in Section 2. In Section 3 we describe the most significant hacking events and their characteristics. In Section 4 we identify trends in the development of hacking. In Section 5 we map the mayor hacking events to our temporal attack model. Section 6 focuses on the protection of critical infrastructure. In Section 7 we contemplate mayor future hacking events.

2 Attack Model

2.1 Computer Attack Taxonomy and Ontology

A detailed Taxonomy that describes computer based attacks has the following classes [4]: Actor; Actor Location; Aggressor; Attack Goal; Attack Mechanism; Automation Level; Effects; Motivation; Phase; Scope; Target; and Vulnerability.

The Taxonomy was then used to describe the following scenarios [4]: Denial Of Service; Industrial Espionage; Web Deface; Spear Phishing; Password Harvesting; Snooping for secrets; Financial theft; Amassing computer resources; Industrial Sabotage; and Cyber Warfare.

2.2 Temporal Attack Model

The Phase class in Section 2.1 was used to build the Temporal Attack Model.

The Target Identification stage represents actions undertaken by an attacker in choosing his/her target. Identification of these actions falls outside the scope of the network attack prediction project, but forms part of the overall attack model.

The Reconnaissance stage represents actions undertaken by an attacker to identify potential weak spots. These actions are the earliest indicators of an impending network attack, and occur before any real damage has occurred. Popular reconnaissance actions include network mapping and scanning with tools such as Nmap and Nessus. Google and other search engines can also be used to identify potential weak spots.

The Attack stage represents modification of the target system by an attacker. The system can be modified in the following aspects: Confidentiality; Integrity; and Availability.

These aspects are also known as the CIA principles. Confidentiality refers to prevention of disclosure of information to unauthorized individuals or systems. Integrity means that data in a system cannot be modified undetectably. Availability refers to the availability of information when required by the system to serve its purpose. In computing, e-Business and information security, it is necessary to ensure that data, transactions, communications and documents are genuine. It is also important that authentication validates the identities of both parties involved.

In figure 1 the Temporal Attack Model is represented. The Attack stage is subdivided into sub-stages. The first sub-stage is the Ramp-up stage. This sub-stage refers to the preparatory actions performed by an attacker before his/her final goal can be attained. The targeted computer network is modified in this stage, but only in preparation for some other goal. This stage typically includes the installation of backdoors and other malware.

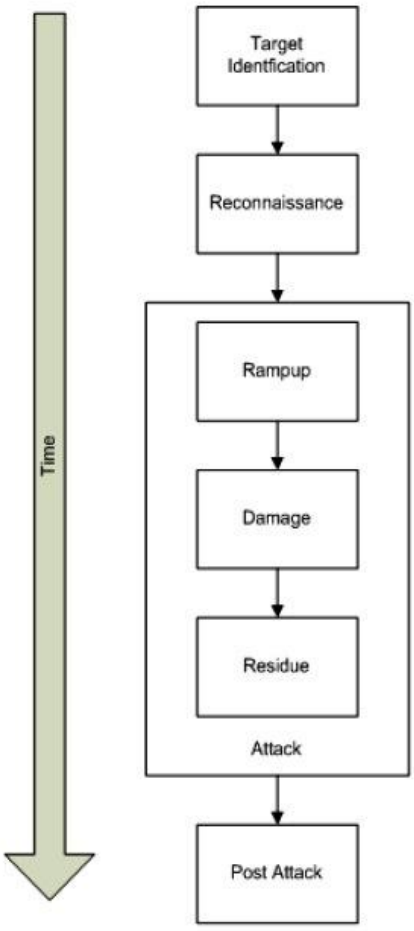

Fig. 1. Temporal Network Attack Model.

The Damage sub-stage refers to actions undertaken by an attacker during the achievement of his/her final goal. In this sub-stage the network is damaged according to the Information Security CIA principles. For example when an attacker launches a Distributed Denial of Service (DDoS) attack on a network, the Damage sub-stage is

entered as soon as the attack is launched. The action of installing DDoS attack software falls under the Ramp-up stage.

The Residue sub-stage refers to unintended communications and actions by malware after an attack has been completed. For example, computers that have incorrect time settings may attack their target at a later date and/or time than when the original coordinated attack was planned. This is also noticed in DDoS attacks.

The Post-Attack Reconnaissance stage refers to scouting and other similar reconnaissance actions performed by an attacker after completion of the Attack stage. The attacker's goal in this stage is to verify the effects of his/her attack and to assess whether the same methodology can be used again in the future.

3 Significant Hacking Events

We consider the following to be the most significant hacking events.

3.1 The Original Logic Bomb 1982

In 1982 the Central Intelligence Agency (CIA) discovered that Soviet spies are planning to secretly acquire a gas pipeline controller built in Canada. The CIA proceeded to plant a Trojan horse, which consisted of a logic bomb, in the software of the controllers [5]. The controller software controlled the testing of the pipeline pressure gauges, and the logic bomb caused the resetting of the gauges to read two-fold higher than the actual gas pressure in the pipelines. This resulted in one of the most monumental non-nuclear explosions ever seen from space [6]. This event is one of the first known physical attacks perpetrated by means of hacking.

3.2 Brain Virus

The world's first computer virus was created by two brothers, Basit and Amjad-FarooqAlvi, in Lahore, Pakistan [7]. This was a boot sector virus since it only affected boot records [8]. The Brain virus marked the area where the virus code was hidden as having bad sectors [9]. It occupied a part of the computer memory and infected any floppy disk that was accessed. It hid itself from detection by hooking into the INT13. When an attempt was made to read the infected sector, the virus simply showed the original sector. This resulted in a change to the volume label.

3.3 Morris Worm

On 2 November 1988 a Cornell graduate student, Robert Tappan Morris, unleashed the first computer worm into the wild [10]. It started as a benign experiment with a simple bug in a program, but the worm replicated much faster than anticipated [11]. By the following morning it had infected over 6000 hosts, nearly 10% of the Internet at the time [12]. Ultimately the worm became a victim of its own success as it could not determine whether a host had already been infected or not. As a result the worm

distributed multiple copies of itself on a single host. The exponential increase in data load eventually tipped off the system administrators and the worm was discovered.

3.4 Chameleon Virus

The first polymorphic viruses appeared in the early 1990s [13]. The Chameleon virus (also known as the 1260 virus), created by Mark Washburn, was the first polymorphic virus to appear in 1990. It consisted of a combination of the Vienna virus and the Cascade virus. Washburn extended the original Cascade virus code and developed a decryptor with a mutable body. The creation of the new polymorphic virus shocked the antiviral community since detection techniques used at the time relied on fixed signatures [14].

3.5 Laroux Virus

The first Excel virus, called Laroux, was discovered in July 1996 [15]. It was a macro virus that consisted of two other macros called Auto_Open and Check_Files, which were contained in a hidden datasheet with the name: Laroux [9]. The Check_Files macro copied the infected worksheet into a file named: Personal.xls. This particular file was located in the Excel start-up directory and enabled all other spreadsheets to be infected. The purpose of the Laroux virus was simply to replicate [9].

3.6 CIH Virus

The CIH virus, also referred to as the Chernobyl or E95.CIH virus, first appeared in June 1998 [9]. It was created by a Taiwanese college student called Chen Ing-Hau [15]. It possessed a destructive payload with the purpose to destroy data. On release, the virus attempted to override a portion of the hard disk as well as the flash ROM of the PC. It infected over a million computers in Korea at the time [9].

3.7 Melissa Virus

The Melissa virus arrived in the early hours of March 26, 1999 in the form of a Word document [15]. David L. Smith was the alleged creator [16]. The Word document, called list.doc, supposedly contained a list of passwords to adult-content web sites. Upon opening the document, the virus turned off the security protocol and e-mailed copies of the infected document to other users of Microsoft Outlook [9]. Melissa was responsible for serious disruptions in big organizations such as Intel, Lockheed-Martin and Microsoft. It was at its time one of the most damaging computer viruses ever created.

3.8 I-LOVE-YOU Worm

The I-LOVE-YOU worm first appeared on May 4, 2000 in the form of an e-mail with the subject: I-LOVE-YOU [9]. It was created by a student named Onel de Guzman, and originated from Manila, Philippines. The worm code was written using Visual Basic and processed by the WScript engine [17]. It targeted computers using Internet Explorer and Microsoft's Outlook application. Within a few hours it had spread worldwide via e-mail by making use of addresses in the Outlook address books of infected users. This worm exploited human curiosity in order to entice people into opening an untrusted email.

3.9 Code Red Worm

The Code Red worm appeared on July 12, 2001. It exploited a buffer-overflow vulnerability in Microsoft's IIS web servers [18]. Upon infection of a machine, it checked whether the date was between the first and the nineteenth of the month. If so, a random list of IP addresses was generated and each machine on the list was probed to infect as many other machines as possible. Proper propagation of the worm failed due to a code error in the random number generator [19]. On 19 July a second version of the Code Red worm appeared that infected computers at a rate of 200 hosts per minute [11].

3.10 Titan Rain

A series of 'cyber raids' carried out by alleged government-supported cells in China was discovered in 2004 by Shawn Carpenter [2]. Several sensitive computer networks were infiltrated, including Lockheed-Martin and Sandia. The FBI later named it 'Titan Rain'. The motives were mostly political and economical [20]. It is considered one of the most sophisticated state-sponsored computer attacks ever detected. The scale and ambition of this attack made it unique.

3.11 Cabir Worm

The Cabir worm was discovered by Symantec on 14 June 2004. It was the first worm to infect mobile devices [21]. It targeted mobile devices using the Symbian OS. Its creator lived in France and used the name Vallez [22]. Infection occurred via Bluetooth. The infection rate was significantly restricted due to the short transmission distances of Bluetooth [23]. The worm did not succeed in creating major havoc on mobile devices, and caused little damage [21]. However, with the rising popularity of smart-phones it is expected that mobile devices will increasingly become the targets of malware.

3.12 MyDoom Worm

January 27, 2004, saw the arrival of the mass-mailing worm called MyDoom [24]. The worm spread via executable e-mail attachments, and also set up a backdoor Trojan on infected computers. It used its own SMTP engine to forward infected e-mails. During its lifetime it caused an increase in e-mail traffic from 14% to 30% [24]. Public awareness, antivirus software and firewalls using SMTP filtering prevented it from growing rapidly.

3.13 Sony XCP

Sony BMG included digital rights management technologies in the Compact Discs (CDs) released during 2005 [25]. One such technology was XCP, a CD-based protection measure developed by First4Internet. The initial purpose of XCP was to place certain restrictions on the use of purchased CDs. In addition to the restrictions, XCP also created a number of security vulnerabilities for Windows users. Mark Russinovich was the first person to release information about these risks to the public on October 31, 2005. Sony's initial response was slow. By the end of 2005, millions of infected CDs were still available in retail stores before their eventual recall. This vulnerability was an example of where a large international corporation's (SONY) desire to protect its content led to damaging of users' computers and thereby the corporation's reputation.

3.14 Estonia Hack Attack

Early in 2007, a series of politically motivated cyber-attacks struck Estonia [26]. The attacks included web defacements and DDoS attacks on well-known Estonia government agencies, banks and Internet Service Providers. The attacks followed the removal of a 6-foot-tall bronze statue in Tallinn, which commemorated the dead from the Second World War [27]. At the time of the attacks, Estonia was one of the leading nations in Europe with regards to information and communication technologies [26]. This can be considered an example of cyber warfare and its potential effects.

3.15 Conficker Worm

The Conficker worm was the first worm to penetrate cloud technology [21], [28]. It first appeared in November 2008 and quickly became one of the most infamous worms to date. The Conficker worm controlled over 6.4 million computer systems and also owned the world's largest cloud network at the time. As a result of the infrastructure of a cloud, the worm could propagate much faster, infect a broader range of hosts and cause greater damage. Conficker has not been used as an attack weapon since, and it is speculated that it might have been a precursor to Stuxnet.

3.16 Ikee Worm

The first worm to infect Apple's iPhones emerged in 2009. Ikee targeted jailbroken iPhones. It did not cause serious damage to the infected iPhone, but simply changed the wallpaper to an image of the singer Rick Astley. After changing the wallpaper, it sought out other jailbroken iPhones to infect. The creator, a 21-year-old TAFE student called Ashley Towns, only developed the worm in order to raise concerns about certain security issues [29]. It did not contain any malicious content.

3.17 Stuxnet Worm

Stuxnet was one of the most complex threats ever analysed [30]. The primary purpose of Stuxnet was to target industrial control systems such as gas pipelines and power plants with the goal of reprogramming the programmable logic controls (PLCs) of the systems to enable an attacker to control them. Stuxnet was also the first to exploit four zero-day vulnerabilities as well as compromise two digital certificates. As of September 29, 2010, Iran had the greatest number of infected computer systems. Stuxnet has shown that direct-attack attempts on critical infrastructures are no longer a myth but a definite possibility. Stuxnet actions can be considered an act of war, but no one has officially claimed responsibility for it.

3.18 HBGary Hack

In February 2011, a computer attack was launched on one of the leading computer security firms, HBGary Federal [31]. The CEO of HBGary Federal, Aaron Barr, announced that he was going to unmask the well-known hacking group, Anonymous. Anonymous responded swiftly and caused severe damage to the security firm. The attacks resulted in defacement of their website and deletion of vast amounts of data. In addition, a website owned by the owner of HBGary, Greg Hoglund, also went offline and the user registration database was published on the Internet. Anonymous ultimately removed the links to the published e-mails after negotiations with Barr and Hoglund [32]. This attack is considered significant in that it demonstrates the potentially negative impact of skilled hacker groups, and the inherent vulnerability of individuals.

4 Trends

Although our selection of significant hacking events is subjective and does not represent a comprehensive list, some interesting trends can be identified:

4.1 Monetary Impact

The monetary impact of each event is shown in figure 2. The vertical scale represents an estimation of the effect. Effects are classified as follows:

5 – Severe Financial impact
4 – Significant Financial impact
3 – Major Financial impact
2 – Minor Financial impact
1 – Negligible Financial impact

On the horizontal scale, the attacks are listed in chronological order.

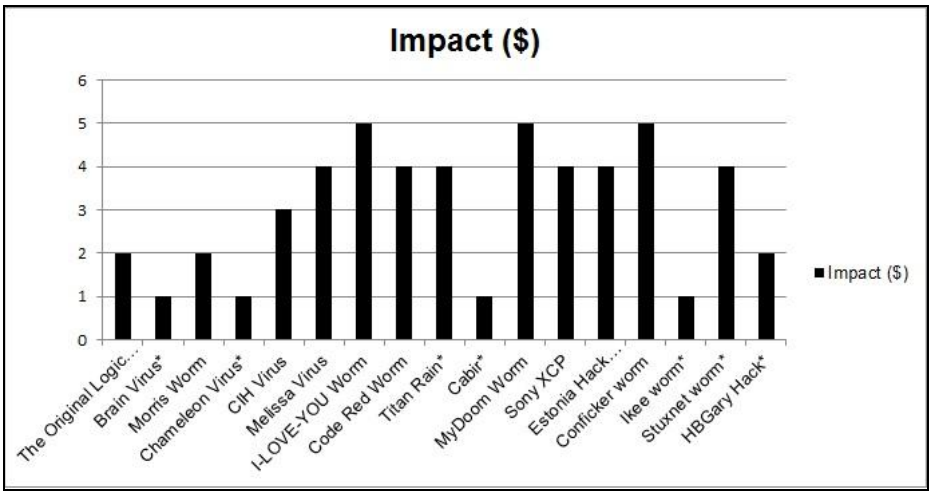

Fig. 2. Monetary impact of hacking events.

4.2 Origin of Hacking Events

Figure 3 lists the most common countries of origin of hacking events. Most events originate from the USA with the Philippines surprisingly in second place.

Figure 4 illustrates the number of events per continent, with the Americas and East Asia at the top of the list.

4.3 Infection Size

In most hacking events, small malware of between 1,000 and 100,000 bytes were utilized. The significant exception is Stuxnet, with a size of over 1.5 megabytes. The progressive increase in bandwidth and computer memory size will likely lend itself to the use of bigger malware (figure 5).

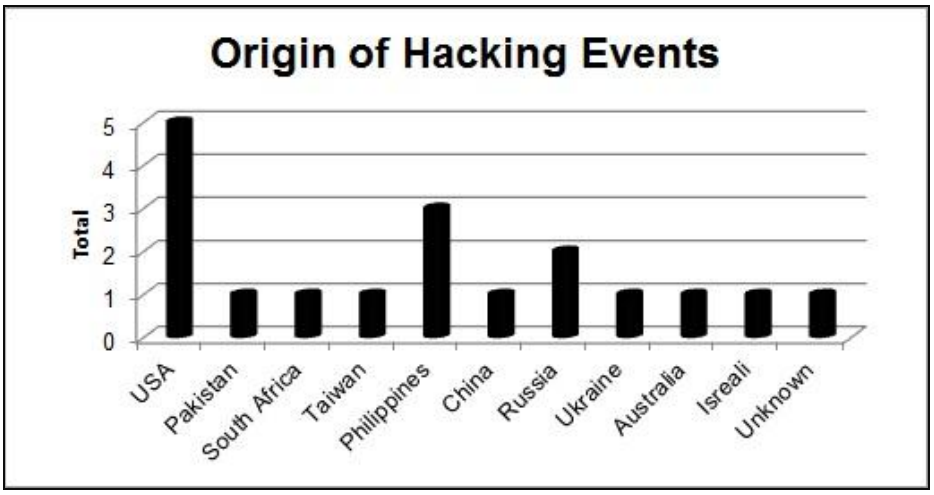

Fig. 3. Countries of origin of hacking events.

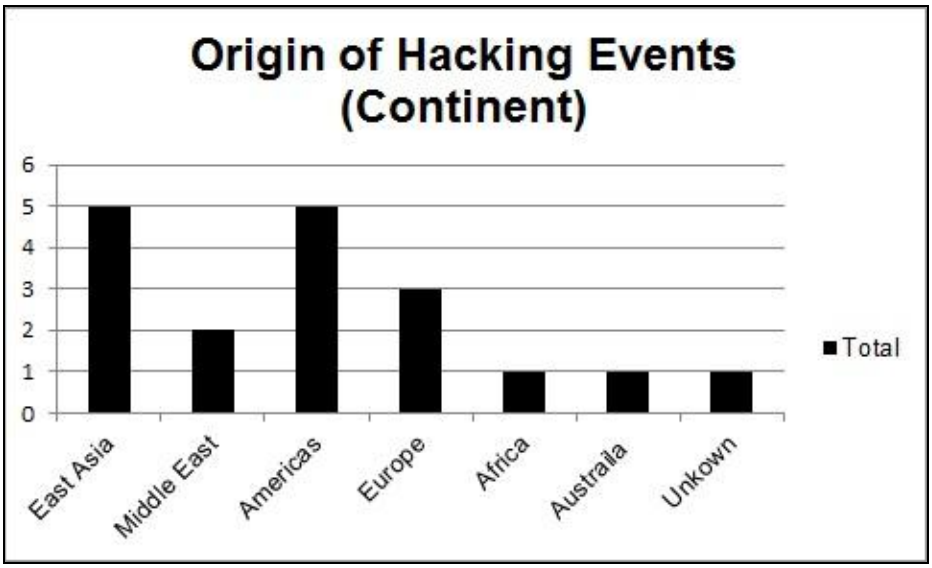

Fig. 4. Hacking events per continent.

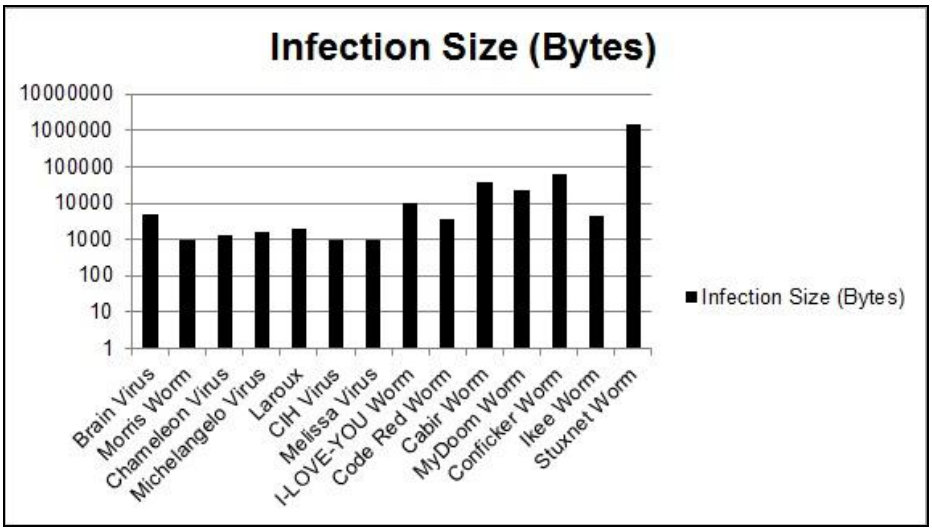

Fig. 5. Infection size in bytes.

5 Attack Model Map

The following sections describe how the most significant hacking events map to the Attack Model in Section 3.

5.1 The Original Logic Bomb

- Target Identification: The CIA was tipped off in the Farewell Dossier document about the Soviet plans to steal control software from a Canadian firm.
- Reconnaissance: Exploring the pipeline control software.
- Ramp-up: Flaws were created in the pipeline control software by means of including a Logic Bomb.
- Damage: The execution of the Logic Bomb that led to the eventual explosion of the gas pipeline.
- Residue: Communications among the CIA after the explosion occurred.
- Post attack: Confirmation received that the explosion occurred.

5.2 Brain Virus

- Target Identification: Experimentation with 5.25 inch floppy disks.
- Reconnaissance: Exploring and experimenting with the DOS File Allocation Table (FAT) file system on the floppy disks.
- Ramp-up: Writing and inclusion of malicious code on a 5.25 inch floppy disk.
- Damage: Changing the volume label to read either Brain or ashar.
- Residue: The changed disk label and the message left in infected boot sectors.

- Post attack: Experimenting with the possibility of using the same technique to infect hard disks.

5.3 Morris Worm

- Target Identification: Experimenting with the ARPANET.
- Reconnaissance: Scanning the ARPANET network for flaws and vulnerabilities.
- Ramp-up: Writing the experimental program which includes the source code for the worm.
- Damage: The release of the experimental program on the ARPANET. An estimated 10% of the current Internet was infected.
- Residue: A machine being infected multiple times rather than only once.
- Post attack: Experimenting with the possibility of a worm in different environments.

5.4 Chameleon Virus

- Target Identification: Evaluating antivirus software.
- Reconnaissance: Exploring and experimenting with the Vienna virus.
- Ramp-up: Deriving the Chameleon virus from the Vienna virus.
- Damage: Circumventing anti-virus software by mutating the virus code.
- Residue: The general logic of the decryption algorithm was preserved.
- Post attack: Experimenting with new polymorphic viruses.

5.5 Laroux Virus

- Target Identification: Exploring Microsoft Excel software.
- Reconnaissance: Experimenting with Microsoft Excel software in search of potential weaknesses.
- Ramp-up: Planting malicious code in a Microsoft Excel file.
- Damage: Infecting Microsoft Excel files and so allow for the spreading of the virus.
- Residue: An error occurs when Excel tries to open the infected file, and the virus cannot replicate.
- Post attack: Experimenting with viruses in other Microsoft Office products.

5.6 CIH Virus

- Target Identification: Exploring the small gaps left in PE (Portable Executable) files.
- Reconnaissance: Experimenting with PE file formats under Windows 95, 98 and ME for potential vulnerabilities.
- Ramp-up: Writing the CIH virus code.

- Damage: Spreading of the CIH virus on computers, and destroying certain PC's BIOS, thus disabling PC use.
- Residue: No unintended attacks caused by the virus in this case.
- Post attack: Verifying the effects of the virus by means of scouting.

5.7 Melissa Virus

- Target Identification: Exploring e-mail systems.
- Reconnaissance: Experimenting with Microsoft Word and Outlook for potential vulnerabilities.
- Ramp-up: Writing the Melissa virus code.
- Damage: Spreading of the Melissa virus via e-mail which causes servers to overload and then performs a Denial of Service Attack.
- Residue: Switching off of security protocols.
- Post attack: Experimenting with the possibility of spreading viruses via e-mail systems.

5.8 I-LOVE-YOU Worm

- Target Identification: Exploring and experimenting with the Windows operating system and Microsoft Outlook.
- Reconnaissance: Using well-known search engines to search for potential weaknesses in the Windows operating system and Microsoft Outlook.
- Ramp-up: Writing the I-LOVE-YOU worm code.
- Damage: The spreading the I-LOVE-YOU worm via e-mail. It led to an effective Denial of Service attack for e-mail.
- Residue: Only hidden files with .mp2 and .mp3 extensions.
- Post attack: Searching for other additional weaknesses in the Windows operating system and Microsoft Outlook.

5.9 Code Red Worm

- Target Identification: Exploring the Microsoft IIS server configurations.
- Reconnaissance: Using well-known search engines to search for potential vulnerabilities in the IIS server software.
- Ramp-up: Writing the Code Red worm code and identifying a buffer overflow vulnerability in the software.
- Damage: Launching a Denial of Service attack against randomly selected server IP addresses.
- Residue: The worm used a static seed as its random number generator and so generated identical lists of IP addresses that caused computers to be infected multiple times.
- Post attack: Searching for additional weaknesses in Microsoft’s IIS servers.

5.10 Titan Rain

- Target Identification: Chinese attackers that explore sensitive computer networks in the United States.
- Reconnaissance: Using network mapping and scanning techniques to find potential weaknesses in the network.
- Ramp-up: Installing Trojan horses (to create zombie computers) and spyware.
- Damage: Launching of the attack by executing Trojan horses and spyware on the infected computers and stealing secret and classified information.
- Residue: No unintended communication occurred in this case.
- Post attack: Searching for other additional weaknesses in other network systems by using the same techniques.

5.11 Cabir Worm

- Target Identification: Experimenting with and exploring mobile operating systems.
- Reconnaissance: Using well-known search engines to find potential weaknesses in the Symbian operating system.
- Ramp-up: Writing the code for the Cabir worm.
- Damage: Launching the worm on mobile phones using the Symbian operating system.
- Residue: Decreased battery life of the infected mobile device.
- Post attack: Searching for additional weaknesses in other mobile operating systems.

5.12 MyDoom Worm

- Target Identification: Experimenting with and exploring the Windows operating system.
- Reconnaissance: Using well-known search engines to find potential weaknesses in the Windows operating system.
- Ramp-up: Writing the code for the MyDoom worm.
- Damage: Spreading the worm in computer networks via e-mail. It was also used to attack SCO with a Denial of Service attack.
- Residue: No unintended effects resulted from the worm in this case.
- Post attack: Releasing a second version of the worm to verify the effects of the first version.

5.13 Sony XCP

- Target Identification: The inclusion of XCP (Extended Copy Protection) software by Sony on Compact Discs.
- Reconnaissance: Exploring XCP software.
- Ramp-up: Installation of XCP software on Windows operating systems, which subsequently created a rootkit.

- Damage: Damaging Personal Computers to allow other malware to infect and control them.
- Residue: Rootkit remaining unknowingly on the computer.
- Post attack: Other malware exploiting the vulnerabilities created by the XCP rootkit.

5.14 Estonia Hack Attack

- Target Identification: The relocation of the Bronze Soldier in Tallinn.
- Reconnaissance: Using well-known search engines to identify possible weaknesses in the websites of well-known Estonian organizations.
- Ramp-up: Installation of Trojan horses and other malware on targeted computer systems.
- Damage: Government and commercial services (such as banks) became unavailable during the attack.
- Residue: Russian-language bulletin boards and one defaced website with the phrase: "Hacked from Russian hackers".
- Post attack: Scanning of the infected computer networks to determine the effects of the attacks.

5.15 Conficker Worm

- Target Identification: Exploring cloud computing.
- Reconnaissance: Using well-known search engines to identify possible vulnerabilities in a cloud computing system.
- Ramp-up: Writing the code for the Conficker worm.
- Damage: Launching the Conficker worm in the cloud, thus making its target resources available to the attacker.
- Residue: No unintended attacks caused by the worm in this case.
- Post attack: Releasing additional versions of the worm to verify the effects.

5.16 Ikee Worm

- Target Identification: Exploring Apple's new iPhones.
- Reconnaissance: Using well-known search engines to identify possible vulnerabilities in iPhones, and experimenting with iPhones.
- Ramp-up: Writing the code for the Ikee worm.
- Damage: It disabled SSH service and tries to infect more phones.
- Residue: An error caused the background of an infected iPhone to be sent to other iPhones and thereby caused new infections.
- Post attack: Releasing additional versions of the worm to verify the effects.

5.17 Stuxnet Worm

- Target Identification: Uranium enrichment infrastructure in Iran.
- Reconnaissance: Using well-known search engines to identify possible vulnerabilities in industrial software and equipment developed by Siemens.
- Ramp-up: Writing the code for the Stuxnet worm and installing additional malware on targeted computer networks.
- Damage: It physically damaged the Iranian Nuclear enrichment systems.
- Residue: Infiltration of computer systems other than those in Iran.
- Post attack: Verifying the effects on Iran's industrial software systems.

5.18 HBGary Hack

- Target Identification: Unmasking of the hacker group called Anonymous by the CEO of HBGary Federal.
- Reconnaissance: Exploring and experimenting with HBGary's servers using network mapping and scanning tools.
- Ramp-up: Installation of malware in preparation of the attack.
- Damage: Launching the attack on HBGary's computer networks and servers. Damaging HBGary's systems and reputation.
- Residue: Negotiation occurring between the CEO of HBGary and Anonymous.
- Post attack: Using scanning tools to verify the effects of the attack.

6 Protection of Critical Infrastructure

The protection of critical infrastructure involves the readiness to act against serious incidents threatening the critical infrastructure of a nation. Recently there is an increasing need to protect critical infrastructure from terrorist or other physical attacks, including cyber-attacks [33]. The previous sections emphasized this need by reviewing 18 of the most significant computer network attacks. Apart from Stuxnet, there have been other instances of infrastructure attacks through computer networks [34]:

- Maroochy Shire Council's sewage control system in Queensland, Australia was attacked.
- A teenager in Worcester, Massachusetts broke into the Bell Atlantic computer system and disabled part of the public switched telephone network using a dial-up modem connected to the system. This attack disabled phone services at the control tower, airport security, the airport fire department, the weather service, and carriers that use the airport.
- In 2000, the Interior Ministry of Russia reported that hackers seized temporary control of the system regulating gas flows in natural gas pipelines.
- In August 2005, Zotob worm crashed thirteen of DaimlerChrysler's U.S. automobile manufacturing plants forcing them to remain offline for almost an hour. Plants in Illinois, Indiana, Wisconsin, Ohio, Delaware, and Michigan were also forced down.

- The Sobig virus was blamed for shutting down train signaling systems throughout the east coast of the U.S. The virus infected the computer system at CSX Corp.'s Jacksonville, Florida headquarters, shutting down signaling, dispatching, and other systems.
- The Nuclear Regulatory Commission confirmed that in January 2003, the Microsoft SQL Server worm known as the Slammer worm infected a private computer network at the idled Davis-Besse nuclear power plant in Oak Harbor, Ohio, disabling a safety monitoring system for nearly five hours.

The Attack Model of Section 2.2 is able to map these Infrastructure computer based attacks. The ultimate goal of this research is to prevent such attacks by identifying the initial stages early enough for preventative actions. The model is able to present any type of computer network based attack, since computer based attacks on Infrastructure uses the same techniques and methodologies as traditional computer network attacks. The Reconnaissance and Ramp-up stages for attacking Infrastructure are similar for attacking computer networks.

7 Conclusion and Future Work

The goal of the network attack model was to represent the majority of network based attacks. This temporal model was verified by mapping 18 significant computer network attacks. The attacks were chosen to represent the most significant computer attacks (hacks) in the authors view. The mapping of these attacks shows the usability of the temporal model in aiding critical infrastructure protection.

To prevent or protect against future computer attacks, the CSIR are investigating methods to detect the Reconnaissance and Ramp-up stages of an attack. If these stages can be detected, mitigating action can be taken against any computer attack. The attack model is still under development and will evolve as the research progress. Future work includes adding additional dimensions to the classification of the attacks, namely origin and motivation of the attack. Reviewing the reasons of why a network was easily penetrated as well as focusing on the commonalities from which lessons can be learned, will also be explored.

References

1. Heater, B.: Male: A Brief Timeline (2011), http://www.pcmag.com/slideshow/story/261678/malware-a-brief-timeline/
2. Hall, K.: The 7 worstcyberattacks in history (that we know about) (2012), http://dvice.com/archives/2010/09/7-of-the-most-d.php
3. Julian: 10 Most Costly Cyber Attacks in History (2011), http://www.businesspundit.com/10-most-costly-cyber-attacks-in-history/
4. van Heerden, R.P., Irwin B., Burke, I.D.: Classifying Network Attack Scenarios using an Ontology. In: Proceedings of the 7th International Conference on Information Warfare and Security, pp. 331-324 (2012)

5. Goertzel, K.M.: Software Survivability: Where Safety and Security Converge, CrossTalk 34(6), pp. 15-19 (2009)
6. Safire, W.: The farewell dossier. New York Times (2004), http://www.nytimes.com/2004/02/02/opinion/the-farewell-dossier.html
7. Desai, P.: Towards an undetectable computer virus, Master's thesis, San Jose State University (2008), http://www.cs.sjsu.edu/faculty/stamp/students/Desai_Priti.pdf
8. Subramanya, S.R., Lakshminarasimhan, N.: Computer viruses. Potential IEEE, 20(4), pp. 16-19 (2001)
9. Blümler, P.: I-LOVE-YOU: Viruses, Trojan Horses and Worms, www.econmr.org/datapool/page/30/virus.pdf
10. Orman, H.: The Morris worm: a fifteen-year perspective. Security & Privacy, IEEE, 1(5), pp. 35-43 (2003)
11. Chen, T.M., Robert J.M.: Worm epidemics in high-speed networks. Computer, 37(6), pp. 48-53 (2004)
12. Cass, S.: Anatomy of malice [computer viruses]. Spectrum, IEEE, 38(11), pp. 56-60 (2004)
13. Bania, P.: Evading network-level emulation. Arxiv preprint arXiv:0906.1963 (2009)
14. Beaucamps, P.: Advanced Metamorphic Techniques in Computer Viruses, International Conference on Computer, Electrical, and Systems Science, and Engineering CESSE'07 (2007)
15. Bosworth, S., Kabay, M.E.: Computer security handbook. John Wiley & Sons Inc., New York (2002)
16. Garber, L.: Melissa virus creates a new type of threat. IEEE Computer, 32(6), pp. 16-19 (1999)
17. Bishop, M.: Analysis of the ILOVEYOU Worm (2000), http://nob.cs.ucdavis.edu/classes/ecs155-2005-04/handouts/iloveyou.pdf
18. Moore, D., Shannon, C.: Code-Red: a case study on the spread and victims of an Internet worm. In: Proceedings of the 2nd ACMSIGCOMM Workshop on Internet measurement, ACM, pp. 273-284 (2002)
19. Zou, C.C., Gong, W., Towsley, D.: Code red worm propagation modeling and analysis. In: Proceedings of the 9th ACM conference on Computer and Communications security, ACM, pp. 138-147 (2002)
20. Gandhi, R., Sharma, A., Mahoney, W., Sousan, W., Zhu, Q., Laplante, P.: Dimensions of Cyber-Attacks: Cultural, Social, Economic, and Political. Technology and Society Magazine, IEEE, 30(1), pp. 28-38 (2011)
21. Sarwar, U., Ramadass, S., Budiarto, R.: Dawn Of The Mobile Malware: Reviewing Mobile Worms. In: Proceedings of the 4th International Conference on Sciences of Electronic, Technologies of Information and Telecommunications (SETIT2007), pp. 35- 39 (2007)
22. Gostev, A.: Mobile Malware Evolution: An Overview, Part 1 (2012), http://www.securelist.com/en/analysis?pubid=200119916
23. Attewell, J.: Mobile technologies and learning, In London: Learning and Skills Development Agency, vol. 2, pp. 4 (2005)
24. Dübendorfer, T., Plattner, B.: Host behavior based early detection of worm outbreaks in internet backbones. In: Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WET ICE), pp. 166-171 (2005)
25. Mulligan, D.K., Perzanowski, A.K.: The Magnificence of the Disaster: Reconstructing the Sony BMG Rootkit Incident, Berkeley Technology Law Journal, 22(3), pp. 1157 - 1232 (2007)

26. Czosseck, C., Ottis, R., Taliharm, A.M.: Estonia after the 2007 Cyber Attacks: Legal, Strategic and Organisational Changes in Cyber Security. International Journal of Cyber Warfare and Terrorism (IJCWT), 1(1), pp. 24-34 (2011)
27. Davis, J.: Hackers Take Down the Most Wired Country in Europe, Wired Magazine, 9(15) (2007)
28. Sharma, V.: An Analytical Survey of Recent Worm Attacks, In IJCSNS, 11(11), pp. 99 - 103 (2011)
29. Andersen, B.: Australian admits creating first iPhone virus (2012), http://www.abc.net.au/news/2009-11-09/australian-admits-creating-first-iphone-virus/1135474
30. Falliere, N., Murchu, L.O., Chien, E.: W32.stuxnet dossier: version 1.4, White paper, Symantec Corp., Security Response (2011), http://www.wired.com/images_blogs/threatlevel/2011/02/Symantec-Stuxnet-Update-Feb-2011.pdf
31. Bright, P.: Anonymous speaks: the inside story of the HBGary hack (2012), http://arstechnica.com/tech-policy/news/2011/02/anonymous-speaks-the-inside-story-of-the-hbgary-hack.ars
32. Zetter, K.: Anonymous Hacks Security Firm (2011), http://www.wired.com/threatlevel/2011/02/anonymous-hacks-hbgary/
33. Bradley, F.: Critical infrastructure protection. Electric Energy T and D, 7(2), pp. 4-6 (2003)
34. Tsang, S.: Cyberthreats, Vulnerabilities and Attacks on SCADA Networks (2009), http://gspp.berkeley.edu/iths/Tsang SCADA%20Attacks.pdf