Computer Viruses and security threats of the web

Contents

[Overview 1](#_Toc401928492)

[Literature review 5](#_Toc401928493)

[Technical Analysis 6](#_Toc401928494)

[Modern Day Problems 7](#_Toc401928495)

[Conclusion 9](#_Toc401928496)

Overview

In this report I will explain several aspects of computer security namely, computer viruses and explain what they are, what they can do and what has been done to counter them. I will also explain all of this and more in a structure manner. We will first discuss in the literature review, the origins of computer viruses, how they came to be and what we the motivations of these computer virus. I will then explain what counter measures have been done to prevent if not limit the amount of damage, the afflicting virus has caused to the system and their users. In my technical analysis I will break down the components of a computer virus and explain the inner workings of 3 popular types of computer virus I.e. Trojans, worms and malware/spyware. I will also discuss the state of the art approaches employed by popular internet security forms such as McAfee, AVG and Norton to counter these types of computer viruses.

In the modern problems section I will also discuss the ever growing problem of phishing, its origins and purpose and what progress has been done to counter phishing, at the end of the section I will give a detail analysis on the progress made by security firms and whether or not it has been effective or counter phishing or not.

To conclude I will summarise my finding on the progress to counter the phishing problem and provide my perspective on what has been done correctly and what can be improved, finally I will give some advice to readers, on how to build upon this analysis and provide a set of advice to further counter the phishing problem.

# Background Information

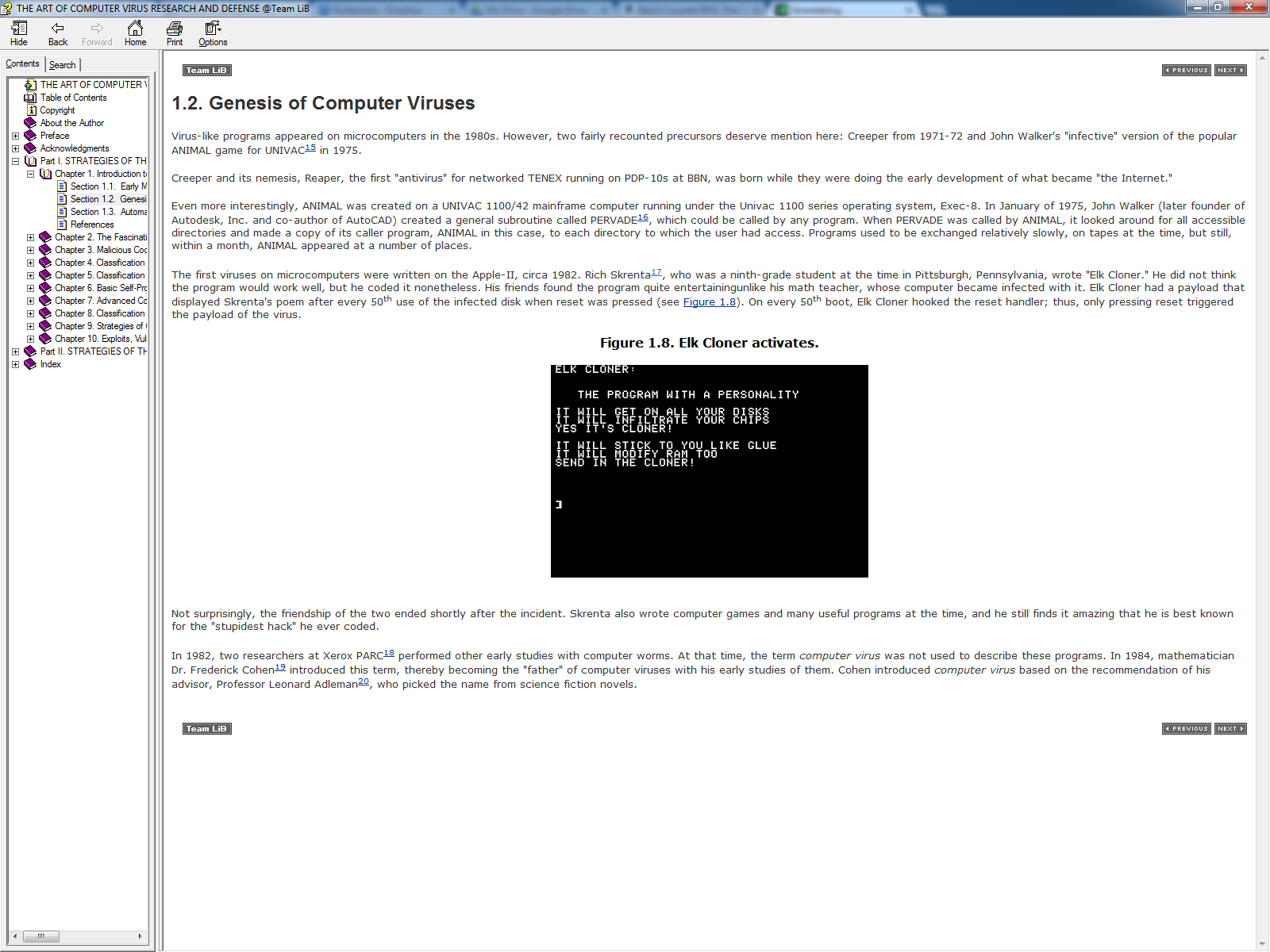
This sections serves to provide an insight on a computer virus, from what it is, however it acts and its characteristics, we will also discuss the approach used to develop anti-virus solutions, again this section will help identify what they are and how they work in theory, finally we will discuss in detail a particular form of internet threat called phishing, which will serve to be the main topic of discussion for further chapters namely the analysis of phishing and anti-phishing solutions. Which talks about a critical analysis on the performance of anti-phishing solutions and anti-phishing techniques, and provides the authors perspective on these techniques and solutions.

## Computer Virus

Virus-like programs first appeared in computers in the 1980’s, however there was two famous examples before the term computer virus was coined, which were Creeper from 1971-72[1] and John walker’s “Infective” version of UNIVAC[cite here[], a popular ANIMAL game in 1975. In fact, the Creeper virus and its rival Reaper[1], which is the first “antivirus” solution for systems within a network namely, TENEX running on PDP-10s. Were both born during the early development what became “the Internet”[1].

The first virus on microcomputers were written on the Apple-ii circa 1982, by Rich Skrenta[1], where he wrote a virus called “Elk Cloner” which in time he commented that he is best known for what he called the “stupidest hack” he ever coded[1]. The Elk Cloner virus was initially thought to not be able to work by its author who continued with the hack despite this opinion. When the virus was deployed, its intention was for its comedy value however, the teachers whom were the initial victims of the Elk Cloner virus proved to be unamused by this. The Elk Cloner virus works by activating its payload (subroutine ) that displayed the author’s poem after every 50th use of the infected disk, when the system was resettled, specifically on the 50th boot Elk Cloner hooked the reset handler meaning the payload would only activate after pressing the reset button. The result is shown in the figure below (Fig 1).

(Fig 1 – Elk Cloner activates [1]).

 Interestingly enough the term “Computer Virus” and thus the classification of these programs, was not introduced until in 1984, a mathematician called Dr. Frederik Cohen[2] introduced term which consequently, named him the “father” of computer viruses due to his early studies of them. Funnily enough the term “Computer Virus” came from science fiction novels, that was picked up and recommended to Dr Cohen’s by his advisor Professor Leonard Adleman [1].

In Cohen’s work a formal mathematical model for computer viruses was created in 1984[1] .

In this study Cohen provided an informal definition of a computer virus: "A virus is a program that is able to infect other programs by modifying them to include a possibly evolved copy of itself." [1] – this definition provides the core interesting properties of a computer virus, such as the possibility of evolution i.e. the ability to make a modified copy of a piece of code with some variations.

However, there is no precise definition of a computer virus due to the abundance of various viruses, which there are infinite variations of themselves. An example of such viruses is called companion viruses[1]that do not necessarily modify the code of other programs. Companion viruses do not strictly follow Cohen's definition because they do not need to include a copy of themselves within other programs. Instead, they make devious use of the program's environment properties of the operating system by placing themselves with the same name ahead of their victim programs on the execution path[1]. This causes problems for other programs that block malicious actions of other viruses. – this assume the principles of these blocker programme developed by their authors strictly follow Cohen’s definition of what virus, thus it only looks for viruses that make unwanted changes to the code of another program, as a result will entirely miss out companion viruses[1].

Integrity checker programs also rely on the fact that a program’s source code remains unchanged over time. Such programs rely on a database (created at some initial point in time) which assumes to represent a "clean" state of the programs on a machine [1]. However, as you can see, companion viruses could easily take advantage of this dependency unless the integrity checker also alerted the user about any new application on the system. This approach to counter companion viruses was tested by Cohen’s own system which properly performed this successfully [1]. Unfortunately, such persistent method would be unpopular with the public. As members do not like to be bothered each time, a new program is introduced on their system. Overall Cohen's approach is definitely the safest technique to use to counter viruses with characteristics of a companion virus[1].

In the real world, behaviour-blocking defence systems often alarm in such a situation. For instance, Norton Commander[3], the popular command shell, might be used to copy the commander's own code to another hard drive or network resource. This action might be confused with self-replicating code, especially if the folder in which the copy is made has a previous which the program would essentially overwrite itself in order to upgrade itself[1].

Taking this example into consideration, a more accurate definition of a computer virus would be the following: "A computer virus is a program that recursively and explicitly copies a possibly evolved version of it."[1]

Computer viruses are self-automated programs that, against the user's wishes, make copies of themselves to spread themselves to new targets[1]. Although particular computer viruses ask the user with prompts before they infect a machine, such as, "Do you want to infect another program? (Y/N?)," this obviously does not make the program not viruses [1].

In the end when it comes to trying to classify a particular program as a virus, we need to ask the important question of whether a program is able to replicate itself recursively and explicitly. A program cannot be classed as a computer virus if it dependant of aids to make further copies of itself, such aid can be in the form of modifying the environment of such a program (for example, manually changing bytes in memory or on a disk)[1].

## Antivirus

In the early stages of virus detection and removal, computer viruses were easily managed because there were very few viruses at the time (there were fewer than 100 known strains in 1990)[cite here].

Initial antivirus solution development were not difficult, this is shown in the late 1980’s and early 1990’s where many users were able to create some sort of antivirus program against a particular strain of computer virus. [cite here].

Fredrick Cohen [cite here] showed that antivirus programs couldn’t solve the computer virus problem as no single program can detect all future computer virus in finite time [cite here]. Despite this, antivirus programs have been quite successful in dealing the computer virus problem for a while. At the same time, other solutions have been researched and developed, however computer antivirus programs are still the most used form of defence against computer viruses at present regardless of their many drawbacks, including the inability to defend and solve the aforementioned problems [cite here].

Often users do not completely understand how to protect themselves against viruses, nor do they know how virus infection prevention can be applied using proper protocols. Unfortunately, negligence is one of the biggest reasons to the spread of computer viruses [cite here]. It was said that the sociological aspects of computer security appear to be more relevant than technology. As careless neglect of the most minimal level of computer maintenance, network security configuration, even failure to clean an infected computer allows further problems to occur in other systems [cite here].

# 

# Literature review

-Background info

--what a virus is

--forms of virus

--examples of virus

# Technical Analysis

# Modern Day Problems

# Conclusion

# Bibliography

[1] P. Szor, *The art of computer virus research and defense*. 2005.

[2] F. B. Cohen, *A Short Course on Computer Viruses*. Wiley, 1994, p. 288.

[3] N. Bezroukov, “The History of Development of Norton Commander,” *Web Page*. [Online]. Available: http://www.softpanorama.org/OFM/Paradigm/Ch03/norton\_commander.shtml. [Accessed: 05-Dec-2014].