NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

ADOPTING IMMUNOLOGICAL METAPHORS IN CYBERSECURITY APPLICATIONS

by

Robert J. Duncan III

September 2022

Co-Advisors: Cristiana Matei
Lauren Wollman (contractor)

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2022	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE ADOPTING IMMUNOLOGICAL METAPHORS IN CYBERSECURITY APPLICATIONS	**5. FUNDING NUMBERS**
6. AUTHOR(S) Robert J. Duncan III	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The evolution of the computer virus remains constant, yet the metaphors used to explain the abstract ideas of computer science remain static. Previous cybersecurity research frames issues of security in physical security metaphors, using tangible ideas or icons, such as castles, to illustrate the need for defense-in-depth models for computer security. Research confirms that security techniques drawn from the castle metaphor serve to prevent infection by a previously identified variant of the virus, but those techniques are weak against novel strain or zero-day exploit. This thesis set out to answer the following question: What role can metaphors from emergent fields play in augmenting the dominant metaphors in cybersecurity applications? This research found metaphors provide limits for defenses and often carry assumptions about system design with them, allowing exploitation in unusual ways. When attacking computer systems designed around physical security models, malicious actors may take advantage of a system's inherent weak points, and infection is inevitable in any networked system. Because complex attacks cannot be prevented by adopting ideas from a single metaphor or discipline of study, this thesis proposes reimagining cybersecurity threats through a wide variety of metaphorical lenses and adopting a plurality of defenses to augment physical security or defense-in-depth metaphors when addressing wicked problems in cybersecurity applications.

14. SUBJECT TERMS computer, consilience, cybersecurity, hacking, internet, immunology, language, metaphor, virus	**15. NUMBER OF PAGES** 63
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

ADOPTING IMMUNOLOGICAL METAPHORS IN CYBERSECURITY APPLICATIONS

Robert J. Duncan III
Attorney-Advisor (Instructor), Federal Law Enforcement Training Centers,
Department of Homeland Security
BSBA, Bacone College, 2006
JD, Oklahoma City University, 2009
LL.M., University of Tulsa, 2014

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF ARTS IN SECURITY STUDIES
(HOMELAND SECURITY AND DEFENSE)**

from the

**NAVAL POSTGRADUATE SCHOOL
September 2022**

Approved by: Cristiana Matei
Co-Advisor

Lauren Wollman
Co-Advisor

Erik J. Dahl
Associate Professor, Department of National Security Affairs

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The evolution of the computer virus remains constant, yet the metaphors used to explain the abstract ideas of computer science remain static. Previous cybersecurity research frames issues of security in physical security metaphors, using tangible ideas or icons, such as castles, to illustrate the need for defense-in-depth models for computer security. Research confirms that security techniques drawn from the castle metaphor serve to prevent infection by a previously identified variant of the virus, but those techniques are weak against novel strain or zero-day exploit. This thesis set out to answer the following question: What role can metaphors from emergent fields play in augmenting the dominant metaphors in cybersecurity applications? This research found metaphors provide limits for defenses and often carry assumptions about system design with them, allowing exploitation in unusual ways. When attacking computer systems designed around physical security models, malicious actors may take advantage of a system's inherent weak points, and infection is inevitable in any networked system. Because complex attacks cannot be prevented by adopting ideas from a single metaphor or discipline of study, this thesis proposes reimagining cybersecurity threats through a wide variety of metaphorical lenses and adopting a plurality of defenses to augment physical security or defense-in-depth metaphors when addressing wicked problems in cybersecurity applications.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Cybersecurity is a highly abstracted idea and relies on extensive use of metaphor to convey those ideas in a tangible way. Despite the pervasive use of metaphor in the field, the corpus of computer science literature holds little discussion of the interaction between metaphor and subsequent security design. When thinking of interaction with a computer, one might use metaphor to view the action of rearranging sequences of information as writing a letter or rearranging pieces of paper on a larger board rather than an abstract data process. Similarly, one might imagine an attack on critical infrastructure security systems as physical attack or virus infection.

The physical attack metaphor is often addressed through system designs that are inspired by military architecture, with castles serving as the most dominant metaphor in the cybersecurity field. This metaphor, while apt, only serves to protect against one dimension of threat: intrusion. When designed around a castle metaphor, a computer system becomes vulnerable in similar ways to historic castles. These castles, strong against a frontal assault, are weak against deception or covert entry. Similarly, computer systems may provide formidable barriers to simple attacks that originate from outside a network but are vulnerable to compromise from within. Once the defenses of a computer system are compromised, the metaphor of castle as security design loses both aptness and effectiveness.

In 2009, devastating cyberattacks against Iranian nuclear centrifuges exploited dimensions of vulnerability in a system designed around the castle metaphor of defense. In a surprising paradigm shift, attackers destroyed physical systems by compromising software systems rather than simply compromising the system or exfiltrating data. The attack was unique in two significant ways: the scale of damage caused by the attack far exceeded what one would expect from a single system, and the vector of attack drew from ideas wholly outside the realm of computer science. Such an attack exploited these new ideas and the underlying assumptions about castles—that they could not be taken from external threat—to develop a strategy that bypassed nearly every defense mechanism and run unchecked through an internal network.

A similar idea allowed the strongest of fortresses to be taken by stealth and guile and demonstrated that ideas from a source domain (in this case, castles) are adapted to a target domain (cybersecurity) will carry parts of a solution set but also parts of a problem set that may go undetected if viewed solely through the lens of the source domain metaphor. The weaknesses of the dominant castle metaphor in cybersecurity applications do not require abandoning existing security measures. Instead, those solutions may be augmented by looking to diverse and divergent source domains, wholly outside the realm of computer science. By viewing problems from different metaphorical lenses, computer scientists may look to any number of ideas from immunology or biology and correct flaws that would otherwise remain undiscovered. As one example, metaphors drawn from a study of vertebrate immune systems may ameliorate weaknesses in the defense-in-depth model used for critical infrastructure.

Future research in areas of epidemiology (drawing experience from contact tracing and social distancing to limit the spread of the computer virus), machine learning (comparing and consolidating blacklist and whitelist data sets), and human interface (resolving exploits related to computer systems that inherently trust human input) may mark the beginning of a new and exciting period in computer science. Other ideas in new and emergent fields, from arachnology to zoology, allow for new and exciting opportunities to reinterpret and reimagine the wicked problems of cybersecurity and provide unbounded solution sets that address each dimension of threat posed to a system. All of this is possible with a thoughtful review of how metaphors from emergent fields play a role in augmenting the dominant metaphors in cybersecurity applications.

ACKNOWLEDGMENTS

I owe a great deal of thanks to a great number of people.

To my grandmother, Dorothy, for encouraging me to learn about castles and computers early in life.

To my daughter, Rian, for demonstrating the creativity to put new ideas together in surprising ways.

To Mike Bunker, David Brunjes, and Joe Haefner for the opportunity to attend the Center for Homeland Defense and Security (CHDS).

To the faculty and staff of CHDS for their valuable guidance, wise counsel, and sympathetic ears.

To Craig Coon for wise advice to back up frequently.

To Marianne Taflinger and Dee Neely for frequent review of my drafts and thoughtful suggestions for improvement.

To Dr. Cris Matei and Dr. Lauren Wollman for patiently guiding me on the way to asking deeper questions and thinking critically about difficult questions in language.

To Nancy Sharrock for teaching me how to sneak past dragons.

And lastly, to Dr. Carolyn Halladay for helping me weather difficulty with constant encouragement: “Don’t give up the ship.”

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

During a hacking event in 2009, devastating cyberattacks against critical infrastructure in Iran marked the first time that digital attacks caused real-world physical damage. The attack was unique in two significant ways: the scale of damage caused by the attack far exceeded what one would expect from a single system, and the vector of attack drew from ideas wholly outside the realm of computer science. If attacks on cybersecurity continue to draw on new dimensions of thought and new domains of knowledge, the ideas used in the homeland defense and security enterprise to conceptualize these threats must incorporate both new theories, as well as ideas that have lain dormant in the computer science lexicon for the past 30 years.

In 1987, Fred Cohen explained the dangers of self-replicating code that could execute commands, override system protections, and spread to networked computers. He described this code as a "computer virus" and predicted future viruses could edit the source code of other programs and "infect" them in such a manner as to replicate endlessly. Through an infection, the virus would spread from a single user account throughout a computer system or network using the authorizations of every user until reaching the root user (the highest level of permissions available to the computer system).[1] A virus capable of evolution beyond the reach and control of its original programming was nothing short of revolutionary and drew from unusual source material: Cohen found inspiration from immunology rather than electrical engineering or the nascent computer science discipline.

Cohen's education came at a time when computer science developed from theory to practice, and when hardware requirements for specific tasks—particularly in government and military service drove the design of systems—gave way to the idea of modular, programmable devices. Older systems handled classified information; security focused on preventing exfiltration and dissemination of data processed by the system, often incompatible with other devices. In recounting the history of computer science, author

[1] Fred Cohen, "Computer Viruses: Theory and Experiments," *Computers & Security* 6, no. 1 (1987): 22–35.

Steven Levy found the government entities operating the system more likely to fail from parts breakage or clumsy programming than outside tampering or influence.[2]

The new threat of a computer virus created a sudden need for a new kind of defense. Levy notes that before Cohen's demonstration, computer scientists developing systems often did so at the behest of non-technical project heads drawn from military ranks. In computer science applications, abstractions as cybersecurity and resilience require well-established and understood analogs for outsiders to conceptualize them.[3]

In areas that require abstraction of thought for concepts outside one's frame of reference or personal experience—from spatial reasoning and perception of color to perception of identity and reality—metaphor provides a mechanism of explanation and understanding.[4] Therefore, military officers responsible for early large-scale computing projects used familiar metaphors from physical security and the language of war to conceptualize potential defenses.

Deborah Frincke and Matt Bishop confirm "the original and most commonly used metaphor [of defense] is the computer (or network) as a fortress, the walls of which must be guarded against potential breaches."[5] Fortress as a metaphor for computer security conveyed a sense of digital security in line with physical security; computer systems would be protected in the same fashion as walled cities impervious to physical attacks by invading forces.[6] Eric Byres, an International Society of Automation fellow and prolific cybersecurity author, describes the way early adopters of the fortress or castle metaphors

---

[2] Steven Levy, *Hackers: Heroes of the Computer Revolution* (Sebastopol, CA: O'Reilly Media, 2010).

[3] Ronald L. Jackson and Michael A. Hogg, "Sapir-Whorf Hypothesis," in *Encyclopedia of Identity*, vol. 1, ed. Ronald L. Jackson and Michael A. Hogg (Thousand Oaks, CA: SAGE Publications, Inc., 2010), 652–654, https://doi.org/10.4135/9781412979306.n207.

[4] Bevil R. Conway and Ted Gibson, "Languages Don't All Have the Same Number of Terms for Colors—Scientists Have a New Theory Why," *The Conversation*, accessed June 2, 2018, http://theconversation.com/languages-dont-all-have-the-same-number-of-terms-for-colors-scientists-have-a-new-theory-why-84117.

[5] Deborah A. Frincke and Matt Bishop, "Guarding the Castle Keep: Teaching with the Fortress Metaphor," *IEEE Security & Privacy Magazine* 2, no. 3 (May 2004): 69–72, https://doi.org/10.1109/MSP.2004.13.

[6] Levy, *Hackers*.

combined ideas from physical security literature with computer science ideas together in a security approach known as "defense-in-depth." [7]

The technique, described by Byres, is "also known as deep or elastic defense, [and is derived from] a military strategy; it seeks to delay rather than prevent the advance of an attacker, buying time and causing additional casualties by yielding space."[8] At first, this defense was completely successful. Cohen's virus required a human user to inject code into a system deliberately, which could be prevented if the system was insulated against attack. Keeping unwanted users out by making it too difficult to get inside and execute code mirrors the hostile architecture of castles, which Bernard Bachrach explains in his text on castle defenses:

> Military topography includes not only the great walled cities…but also numerous castra, castella, and even less elaborate fortifications along with a magisterial road system…innumerable stone bridges, and an exceptionally elaborate network of ports.[9]

Simon Woodside, writing for *Medium*, explains that a properly designed computer security system should be "designed like a medieval castle, to provide an oasis of security in an uncertain world."[10] Security researchers like Woodside see networks and connections between computers much like the roads and bridges leading to a castle: military fortifications extending well beyond the physical boundaries of the castle and providing a commanding advantage against adversaries.[11] Ideally, "the presence of many independent layers of defences will geometrically increase the difficulty of an attacker to breach the walls, and slow them down to the point where an attack isn't worth the expense it would take to initiate it."[12]

---

[7] Eric J. Byres, "Defense in Depth," *InTech; Durham* 59, no. 6 (December 2012): 38–40, ProQuest.

[8] Byres, 38–40.

[9] Bernard S. Bachrach, "Medieval Siege Warfare: A Reconnaissance," *Journal of Military History* 58, no. 1 (January 1994): 119, ProQuest.

[10] Simon Woodside, "Defence in Depth: The Medieval Castle Approach to Internet Security," Medium, June 20, 2016, https://medium.com/@sbwoodside/defence-in-depth-the-medieval-castle-approach-to-internet-security-6c8225dec294.

[11] As late as 2019, the flagship security product Microsoft Security Essentials uses a stylized blue castle as its application icon.

[12] Woodside, "Defence in Depth."

As described by Byers and Woodside, defense-in-depth represents risk analysis well suited to early computer science in the 1980s: controlling multiple computers to launch a coordinated attack required expertise outside the reach of all but the most dedicated attackers.[13] William Gibson, the progenitor of the cyberpunk genre of fiction, noted that until the 1990s "virus-writers seemed, at least at first, to be in it for anything but money. The outcome was simply vandalism, as dull as someone smashing out the light fixtures in a bus shelter."[14]

In Gibson's far-future work, external penetration was one of many threats to future computer systems. One of the foremost authorities on information security in the 1990s, Winn Schwartau, believed the future was already here. Schwartau saw the potential for sophisticated virus-writers to exfiltrate sensitive trade secrets or classified information vital to national security with simple insider attacks. To this end, Schwartau testified before the United States House Committee on Science, Space, and Technology that computers constantly look for connections in networked systems, and "so-called privacy afforded by walls and doors with locks is actually useless since the computer is indiscriminately transmitting its contents to the world" when waiting for a reply.[15]

A team led by Jeffrey O. Kephart, a Fellow of the Institute of Electrical and Electronics Engineers and IBM researcher, also warned that physical security models would be ineffective in stopping the spread of a computer virus when a computer system listened for commands or inputs.[16] In one of the best-known examples, the so-called Trojan Horse virus, trusted systems are infected by way of an insider and subsequently connected to a second system. The trust afforded to the original system would allow the connection to the second system to carry an infection into a larger network and infect computers at an

[13] Byres, "Defense in Depth," 38–40.

[14] William Gibson, "25 Years of Digital Vandalism," *New York Times*, sec. Opinion, January 27, 2011, https://www.nytimes.com/2011/01/27/opinion/27Gibson.html.

[15] *Hearing before the Subcommittee on Technology and Competitiveness of the Committee on Science, Space, and Technology*, House of Representatives, 102nd Cong., 1st. sess., June 27, 1991, 14, https://winnschwartau.com/wp-content/uploads/2019/06/Testimoney-1991-Computer-security_hearing.pdf.

[16] Jeffrey O. Kephart, Steve R. White, and Dave M. Chess, "Computers and Epidemiology," *IEEE Spectrum* 30, no. 5 (1993): 20–26.

exponential rate. The mechanism of infection, and even the terms used to describe it come from immunology. Kephart thus proposed a model inspired by epidemiology:

> The best approach a company can take today is to encourage users to inform a central agency about their machines' infection, and to have the central agency respond by helping those users clean up their machines and then check neighboring machines for infection.[17]

Kephart's suggestion was never implemented. No clear theory has emerged to explain why, but researcher Andy Greenberg theorizes that "zero-day exploits"—malicious code targeting existing unpatched vulnerabilities in software—are more profitable when security researchers weaponize the exploit into a virus rather than to immunize systems against it:

> [F]ind a previously unknown method for dismantling the defenses of a device…present it at a security conference to win fame and lucrative consulting gigs. Share it with HP's Zero Day Initiative instead and earn as much as $10,000 for helping the firm shore up its security gear[…] [or] arrange a deal through [a] pseudonymous exploit broker to hand the exploit information over to a government agency, don't ask too many questions, and get paid a quarter of a million dollars.[18]

Schwartau's and Kephart's fears became reality in the early 2000s. Deliberate virus attacks against "cryptographic systems that protect strategically sensitive and often classified information" shifted from theory to practice, with such attacks becoming increasingly common.[19]

In 2017, an expose by WIRED magazine contributor Lily Hay Newman revealed the United States government kept knowledge of exploits secret, preventing software makers from developing a fix. One exploit was stolen by hackers and later formed the basis of the destructive WannaCry virus:

---

[17] Kephart, White, and Chess, 20–26.

[18] Andy Greenberg, "Shopping for Zero-Days: A Price List for Hackers' Secret Software Exploits," Forbes, March 23, 2012, https://www.forbes.com/sites/andygreenberg/2012/03/23/shopping-for-zero-days-an-price-list-for-hackers-secret-software-exploits/.

[19] Dan Patterson, "U.S. Grapples with Controlling 'Cyber-Munitions' While Recruiting 6,000 New Cyber-Warriors," TechRepublic, accessed June 3, 2018, https://www.techrepublic.com/article/u-s-grapples-with-controlling-cyber-munitions-while-recruiting-6000-new-cyber-warriors/.

> WannaCry's evolution is the latest example. The attack spread by exploiting a Windows server vulnerability known as EternalBlue. The NSA discovered the bug and was holding on to it, but information about it and how to exploit it was stolen in a breach and then leaked to the public by a hacking group known as the Shadow Brokers.[20]

The damage caused by the leaked virus was so substantial, Microsoft called "for a new 'Digital Geneva Convention' to govern [the] issues [of so-called cyberweapons], including a new requirement for governments to report vulnerabilities to vendors, rather than stockpile, sell, or exploit them."[21] The damage is also persistent, causing long-lasting effects on systems. Scott Granneman, reporting for *The Register* on the proliferation of computer virus exemplars, noted that the infection rate was "growing faster than the average time it took to download an update package. When the increase in new infection is past the point that updates can keep up, infection is inevitable."[22]

As the computer virus remains a wicked problem that cannot be addressed by a single domain or dominant metaphor, this thesis proposes reimagining cybersecurity threats through a wide variety of metaphorical lenses and adopting a plurality of defenses to augment defense-in-depth.

A. RESEARCH QUESTION

What role can metaphors from emergent fields play in augmenting the dominant metaphors in cybersecurity applications?

B. RESEARCH DESIGN

My research confirms the computer virus has evolved into a weaponized polymorphic virus, or a biological weapon capable of escaping confinement and mutating. Despite the ability of the computer virus to escape confinement and mutate, current security

---

[20] Lily Hay Newman, "Why Governments Won't Let Go of Secret Software Bugs," WIRED, May 16, 2017, https://www.wired.com/2017/05/governments-wont-let-go-secret-software-bugs/.

[21] Brad Smith, "The Need for Urgent Collective Action to Keep People Safe Online: Lessons from Last Week's Cyberattack," *Microsoft on the Issues* (blog), May 14, 2017, https://blogs.microsoft.com/on-the-issues/2017/05/14/need-urgent-collective-action-keep-people-safe-online-lessons-last-weeks-cyberattack/.

[22] Scott Granneman, "Infected in 20 Minutes," August 19, 2004, https://www.theregister.co.uk/2004/08/19/infected_in20_minutes/.

techniques drawn from the castle metaphor serve to prevent infection by a previously identified variant of the virus. If the most sophisticated defense-in-depth solutions are targeted by a novel strain or a strain engineered to take advantage of a system's weak points, infection is inevitable in any networked system. The rapid evolution of the computer virus thus presents a wicked problem that cannot be solved with ideas from a single branch of study.

To address this wicked problem, I first explored public source data on targeted attacks against nuclear infrastructure in Iran and the failed attempts to prevent such an attack. The efforts to prevent an attack are rooted in physical security domains, and I argue that the differences in metaphor used to explain physical security and cybersecurity domains conceal attack vectors. In the following chapter, I then describe metaphor as the mechanism by which complex ideas may be expressed, discuss the connection between historical use of metaphor and modern thinking, the constraints applied to metaphor as a borrower of an ideas, and the dangers of such constraints in Cohen's virus. With the need for plurality of defenses against the computer virus, I conclude by posing the following question: What role can the augmentation of dominant metaphors with those from emergent fields play in existing cybersecurity problems?

THIS PAGE INTENTIONALLY LEFT BLANK

II. DEFENSE-IN-DEPTH

Defense-in-depth is modeled after the defensive systems common to historical castles. Castles served as a deterrent against peasant revolts or border raids; defense-in-depth deters unskilled attackers by presenting a strong defense against intrusion.[23] In a time when virus-writers attacked hobbyist computers and could cause little harm, one could more easily understand the abstract concept of defense by looking at the threat posed from outside penetration as more significant than a threat from within. As complex networks developed—especially those systems installed in sensitive areas, such as nuclear enrichment sites or those systems handling sensitive data, such as servers at an Office of Personnel Management data center—the dimensions of risk increased to encompass both outside penetration and insider threats. The simultaneous attack of both outer defenses and internal network presents complex problems not addressed by the dominant metaphor of castle or defense-in-depth.

In a modern attack, determined adversaries will seek to defeat the defense-in-depth model of computer systems by first breaching the network and then introducing a malicious program or virus that can spread through a network.[24] To better understand how a Trojan Horse virus works, consider the following hypothetical sequence of events for a typical virus. Through routine use of the computer, an infected file (a tracking cookie from a malicious website or a file with hidden code) is downloaded to a target computer before sending out a beacon to an external command and control server. The beacon allows the command-and-control server to detect or "sniff" open ports on a machine. These ports expect some reply or command (such as incoming email or website requests) and allow the reply or command to be executed on the target computer. Once the computer executes the command, the malicious code elevates lower user privilege functions until administrator or root access is gained. Once the command-and-control server gains administrator or root access, the infected computer becomes a launching point for attacks on both internal and

[23] Woodside, "Defence in Depth."

[24] Frincke and Bishop, "Guarding the Castle Keep."

external networks.[25] This model of attack exploits vulnerabilities in the defense-in-depth model that have been carried over from castles; castles are robust defensive structures when assaulted from outside but extremely weak against insider attack and deception. Even when systems are designed to restrict all incoming traffic and close all ports, sophisticated attacks rely on overlooked vectors to inject code. This chapter describes the use of overlooked vectors to bring down significant infrastructure, the use of the castle metaphor to develop defense-in-depth systems, the weaknesses of historical castles, and the weaknesses carried over to systems that adopt the metaphor to explore the role augmentation of dominant metaphors with those from emergent fields can play in existing cybersecurity problems.

A. DEFENSE-IN-DEPTH FAILURES AT NATANZ

In 2009, Iranian officials discovered widespread physical damage at the Natanz facility when programming changes in nuclear control systems caused large centrifuges to spin out of control. More worrisome than the damages, the design of Natanz made it seemingly impossible to change the programming control logic. Following the design principles of defense-in-depth, all systems at Natanz were isolated from the outside world in a configuration known as air-gapping: no connection to the Internet, no remote access, and no open ports to the outside of any kind. Further, Iran relied only on trusted and vetted outside contractors to provide software updates and technical support for programmable logical controllers running centrifuge equipment. To ensure that the system itself could not be compromised, contractors developed software updates off-site using mirror images of the programmable logical controllers inside the Natanz facility. Any updates developed would need to be physically carried into the facility on a portable solid-state memory drive (commonly known as a flash drive or universal serial bus (USB) drive before they took effect; contractors would be escorted into the facility under heavy guard, physically carry a USB drive into the facility, run updates from the USB drive while being monitored, and then be escorted out of the facility.

[25] Cohen, "Computer Viruses."

In theory, the use of defense-in-depth to design the facility would have completely prevented the introduction of a virus. In practice, the defense-in-depth model failed to protect against a sophisticated virus known as Stuxnet because the defense-in-depth model failed to address insider threats.[26] Two key assumptions allowed for failure and were exploited: the facility could not be penetrated from the outside and input inside the system should be trusted. Natanz, as a nuclear facility, had strict access controls to address a very real safety concern: outside commands should be ignored as an attempt to disable the system, while commands issued by a human operator on a keyboard inside the system should be immediately obeyed without question to prevent a nuclear disaster.

The first assumption—outside penetration is impossible—was true only to the extent that Natanz could not be penetrated. Natanz, however, relied on outside contractors to bring in updates on a USB drive. Therefore, the contractor computers were part of the Natanz network even though the systems were not physically linked, and thus provided a vector for infection. A similar attack formed a significant plot point for the French novel *The Count of Monte Cristo*, in which the protagonist works to send misleading information among semaphore lines rather than the intended target's location.[27] Semaphore, a means of communicating by waving flags, relies on trained operators to decode and pass on messages within their line of sight.[28] There is no possible way to disrupt the system physically, so a semaphore operator is bribed in the novel instead: he does not pass on the correct message and instead sends the message written by the protagonist. Other semaphore operators down the line pass the information without question.[29] Both attacks are deceptively simple—attack the message handler rather than the mechanism used to pass on the message—and take elegant design cues from both the biological virus and Cohen's 1987 virus with the use of a carrier.

---

[26] David Kushner, "The Real Story of Stuxnet," IEEE Spectrum: Technology, Engineering, and Science News, February 26, 2013, https://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet.

[27] Alexandre Dumas, *The Count of Monte Cristo* (New York: Penguin Books, 2001).

[28] Rebecca Robbins Raines, *Getting the Message Through: A Branch History of the U.S. Army Signal Corps* (Washington, DC: Center for Military History, U.S. Army, 1996).

[29] Dumas, *The Count of Monte Cristo*.

Exploiting the second assumption—commands issued by a human operator on a keyboard inside the system should be immediately obeyed—takes advantage of an implied condition: the keyboard is a human interface device, thus the input from a keyboard is from a human. Vlad Savov, writing about the phenomenon of user error accepted by computers, notes that system designers in critical infrastructure systems program computers to accept "human commands with an uncritical, unquestioning diligence."[30] At Natanz, computer systems were designed to trust the human users entering code, but trust is rarely considered as a dimension of security systems using a defense-in-depth model.[31] Human interface devices (especially keyboards) have no check or safeguard to ensure that a keyboard is a keyboard and that the command entered by keyboard is entered by a human.

Hak5 Gear, a company that specializes in network penetration equipment explains, "nearly every computing devices [sic] accepts human input from keyboards… Keyboards announce themselves to computers as [Human Interface Device] devices and are in turn recognized and accepted."[32] To exploit this implicit trust of human users, malicious actors create a device that would be recognized as a keyboard. Once recognized and accepted, the device will run a script to inject keystrokes and run commands. In the case of Stuxnet, Kim Zetter explains that the virus "spread via USB flash drives using the Windows Autorun feature…using [a] print-spooler zero-day exploit" without detection or resistance.[33] From there, the code was able to "propagate to other machines within that network and gain privilege once it has infected those machines," eventually reaching privileges needed to cause centrifuges to spin themselves apart.[34]

---

[30] Vlad Savov, "The Death of Garbage in, Garbage Out," The Verge, August 16, 2016, https://www.theverge.com/2016/8/16/12499854/first-click-the-death-of-garbage-in-garbage-out.

[31] Sonia Sousa, Paulo Dias, and David Lamas, "A Model for Human-Computer Trust," in *9th Iberian Conference on Information Systems and Technologies* (Barcelona, Spain, IEEE, 2014), 435.

[32] "USB Rubber Ducky," Hak5 Gear, 5, accessed June 2, 2018, https://hakshop.com/products/usb-rubber-ducky-deluxe.

[33] Kim Zetter, "An Unprecedented Look at Stuxnet, the World's First Digital Weapon," WIRED, November 3, 2014, https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/.

[34] Bruce Schneier, "The Story behind the Stuxnet Virus," Forbes, October 2010, https://www.forbes.com/2010/10/06/iran-nuclear-computer-technology-security-stuxnet-worm.html.

This attack demonstrates the significant weaknesses of the defense-in-depth model. Daniel Cohen, after studying the Natanz attack, found the keystroke injection method at the heart of the Stuxnet virus. The Stuxnet virus was written in such a way that it would replicate human interaction and exploit the assumption human users should—and therefore must—be trusted. Cohen noted "hackers compromised the fully air-gapped plant on multiple occasions by targeting companies working with the plant, using USB drives to infiltrate the plant, and finally reaching uranium-enriching centrifuges controlled by programmable logical controllers (PLCs)" where Stuxnet could do the most damage.[35] From there, according to author and technology commentator David Kushner, Stuxnet was "designed to gain system-level privileges even when computers have been thoroughly locked down."[36] Because the device does not contain malware—only a script of commands—the attack cannot be prevented by antivirus software. Worse, keystroke injectors inject commands in milliseconds, so human users may never realize a USB device inserted into a computer is masquerading as a keyboard and entering commands. The introduction of even a single infected USB drive or keyboard allows compromise of a system, often without detection, and is almost impossible to prevent without disabling every USB port on every computer system connected to or networked with the target system.[37] Given the interconnected nature of computer systems, nearly every system is a potential vector for an insider attack; nearly every defense-in-depth security model ignores that potential vector and is thus vulnerable to such an attack.

B. DEFENSE-IN-DEPTH FAILURES AT CHATEAU GAILLARD

Reading the accounts of Stuxnet reminded me of historical castles in England and France taken not by force but by deception, guile, or the introduction of disease. Therefore, I argue that modern security systems built on the castle metaphor only protect against

[35] David Cohen, "Ditching the Air-Gapping Myth. Power-Grid," July 23, 2017, https://www.power-grid.com/td/ditching-the-air-gapping-myth.

[36] Kushner, "The Real Story of Stuxnet."

[37] Bruce Sterling, "The Dropped Drive Hack," WIRED, June 29, 2011, https://www.wired.com/2011/06/the-dropped-drive-hack/.

physical intrusion; the threat imagined when castles were built. Likewise, other attack vectors are overlooked in these designs, and these oversights may thus be exploited.

All through European history, the most effective means of taking a castle was to simply bypass the castle walls: waylaying and impersonating trusted visitors, bribing someone to open the castle gates from inside, or even infiltrating the castle by wriggling up an unsecured toilet shaft. If deception or guile can defeat a castle, similar methods can also defeat a security system designed around the castle metaphor. A computer system may be targeted by a zero-day exploit just as easily as a castle with an unbarred chapel window; the introduction of a custom virus into Natanz is very much the modern equivalent of sending a soldier up the garderobe of a castle. Consider Chateau Gaillard (or "Strong Castle"), a 12th-century French castle long considered impenetrable after the English occupied and rebuilt it. As a critical English stronghold, the design of the inside spaces represented the height of defensive thought at the time: thick walls, counterclockwise staircases meant to hinder attackers, higher ground for defenders, and doors that opened from the inside so defenders could overwhelm assaulters.

Assuming the French would have to first breach the walls before reaching the interior of the castle, the English designers of the castle focused attention on an outside attack. As a result of this assumption, several areas of the castle were built with minimal precautions as a result of this assumption: the chapel featured large, unbarred windows and the exit for the castle's garderobe had no protection.[38] When Philip II gave orders to attack Chateau Gaillard, soldiers found the garderobe and the unsecured chapel provided a perfect entry point. Phillip Warner, an historian with a particular interest in siege warfare of the Middle Ages, recounts the events that led to the castle's fall:

> [O]ne of the French soldiers, who probably knew the castle well, observed that a garderobe (latrine) emptied on the west side. Just above this was a chapel window that was not barred as might have been expected. He crawled up this unattractive path, entered the chapel, and pulled in a few companions through the window.[39]

[38] A garderobe is an early toilet shaft.

[39] Philip Warner, *Sieges of the Middle Ages* (Havertown, PA: Pen and Sword, 2015).

Once inside, French assaulters took English defenders by surprise. The English set fire to their own chapel to smoke out the assaulters, but the French moved through from the inner walls of the castle to the outer ring of defenses at a rapid pace. Unbarring doors as they went, the French soon defeated the castle's defenses so their companions outside could take the castle. The rapid assault demonstrates the weakness of the defense-in-depth model: strong against outside attack and powerless to stop insider attacks. Cohen's virus propagates inside a network, thus insider attacks pose grave risk to computer systems if introduced. Chateau Galliard fell to a simple design flaw: an overlooked and unsecured chapel window. One exploitable line of code among millions is enough to allow a system to be taken in a similar way to a castle with an open window. Just as Chateau Gaillard was vulnerable to entry through an unexpected point and could not be defended once an invader made it past the castle walls, Natanz was vulnerable to infection from an unexpected vector and could not be defended once malicious code infected the servers inside the facility.

Recall how Stuxnet, the computer virus used to attack Natanz, owes its evolution to Fred Cohen's 1987 work.[40] The virus continually escalates user privileges, replicating at each level of computer access, until it eventually controls the entire system.[41] The Stuxnet attack was effective because the popularity of the castle metaphor concealed its significant weakness: defense-in-depth cannot stop the spread of a virus once it takes hold. A well-designed virus will seek out and bypass weaknesses in the targeted system (much like the French soldiers who studied Chateau Gaillard), avoid obvious defenses (much like the French avoided assaulting the castle walls directly), use an unprotected vector to enter (much like the French climbing up to an unsecured window), and exploit trust to defeat the system (much like the French attacking from the inside and using architectural features meant to benefit defenders against the English), just as Cohen warned.

[40] Cohen, "Computer Viruses," 22–35.

[41] Cohen, 22–35.

C. THE METAPHORS ARE NOT THE SAME

Cohen's threat metaphor of virus is firmly entrenched in computer science, but the literature gives no explanation of why his immune system solution was overlooked in favor of the physical defense metaphor of castle despite repeated failures of defense-in-depth (with Natanz as one among many) to contain the virus. Conceptualizing highly abstract concepts like rearranging sequences of information in a computer program into more tangible and relatable ideas, like writing a letter or rearranging pieces of paper on a larger board requires the use of metaphor. The attacks on Natanz demonstrate that the castle metaphor driving defense-in-depth security does not adequately protect against insider attack; I theorize that the weaknesses of castles carried over to Natanz by way of metaphor.

Metaphor is a significant part of the execution of ideas in computer science, even if discussion of metaphor is largely absent in the computer science literature. The importance of metaphor in computer science can be seen without looking any farther than one's keyboard: we "cut" or "paste" when referring to copying text in a word processing document; we have "files" and "folders" instead of describing the way bits and bytes are saved to disk sectors and discuss saving in the "cloud" rather than considering the complex relationship of networked computers balancing loads of data in multiple physical locations. These terms are part of our everyday vocabulary and are used without ever considering the distance between the real act and the metaphorical act or the way we think of one as the other, because of the power of metaphor.

Metaphor allows one idea to stand in for another, even when the two ideas are not strongly related to each other, to provide context and understanding. Although the literature does not directly address the connection between metaphor and cybersecurity, the distance between the way one thinks of source and threat can create an exploitable gap. Understanding metaphor may help to reduce that gap.

Adriane LaPointe, an influential figure in American cybersecurity policy, further notes that "metaphors and analogies emphasize relevant similarities, offer insight into complex issues, and give us ways to talk about new things or situations which are hard to

grasp more literally" like cybersecurity or the concept of cyberspace.[42] To explain online shopping, retailers rely on analogs or metaphors (web, page, storefront, or shopping cart) instead of literal explanations of how an online retail system works. Timothy R. Colburn and Gary M. Shute explain metaphors for applications we use in everyday life often have nothing to do with how the applications work, but can be used to understand abstract concepts:

> In web applications, for example, it is common to refer to certain complicated data structures as *shopping carts*, even though within the application, a complex compendium of web pages, programs, and databases, there is nothing to which the concept of a shopping cart could conventionally apply.[43]

The metaphor, however, connects the ideas in only a tangential way and may lead to misunderstanding if the metaphor is inapt or imprecise. Aristotle's *Poetics* distills metaphor into this explanation: "Metaphor consists in giving the thing a name that belongs to something else; the transference being either from genus to species, or from species to genus, or from species to species, or on grounds of analogy."[44] In their seminal work on metaphor, George Lakoff and Mark Johnson explain that metaphorical thinking forms the warp and woof of modern society's collective thoughts and actions:

> Metaphor is for most people a device of the poetic imagination and the rhetorical flourish—a matter of extraordinary rather than ordinary language...most people think they can get along perfectly well without metaphor. We have found, on the contrary, that metaphor is pervasive in everyday life, not just in language but in thought and action. Our ordinary conceptual system, in terms of which we both think and act, is fundamentally metaphorical in nature.[45]

---

[42] Adriane Lapointe, *When Good Metaphors Go Bad: The Metaphoric 'Branding' of Cyberspace* (Washington, DC: Center for Strategic and International Studies, 2011), https://csis-prod.s3.amazonaws.com/s3fs-public/legacy_files/files/publication/110923_Cyber_Metaphor.pdf.

[43] Timothy R. Colburn and Gary M. Shute, "Metaphor in Computer Science," *Journal of Applied Logic* 6, no. 4 (December 2008): 526–33, https://doi.org/10.1016/j.jal.2008.09.005.

[44] Aristotle, *Poetics* 21, 1457b, 6–7.

[45] George Lakoff and Mark Johnson, *Metaphors We Live By* (Chicago: University of Chicago Press, 2003).

Geary seconds Johnson and Lakoff in arguing that metaphor exists in all human enterprise requiring original thought, which means that choice of metaphor influences the perception of reality. Geary contends that "there is no aspect of our experience not molded in some way by metaphor's almost imperceptible touch. Once you twig to metaphor's modus operandi, you'll find its fingerprints on absolutely everything."[46] Thus, so what?

Geary describes the mechanism used to help grasp concepts works because "metaphor juxtaposes two different things and then skews our point of view so unexpected similarities emerge. Metaphorical thinking half discovers and half invents the likenesses it describes."[47] In an article for *American Psychologist*, Keith Holyoak and Paul Thagard explain that the type of framing provided by metaphor allows the thinker to adopt past experience and understand a new concept or solve a similar problem.[48] Kovecses describes the way in which objects or experiences serve to explain abstractions:

> If we want to fully understand an abstract concept, we are better off using another concept that is more concrete, physical, or tangible than the abstract target concept for this purpose. Our experiences with the physical world serve as a natural and logical foundation for the comprehension of more abstract domains.[49]

Metaphors connecting ideas between source and target domains in logical ways will convey information useful to understanding and solving novel problems. According to Kovecses, a metaphor in which the source and target domains align will convey useful information as in Figure 1.[50] Here, the use of a source domain (hot fluid in a container) better explains the volatile emotional state of an angry person.

---

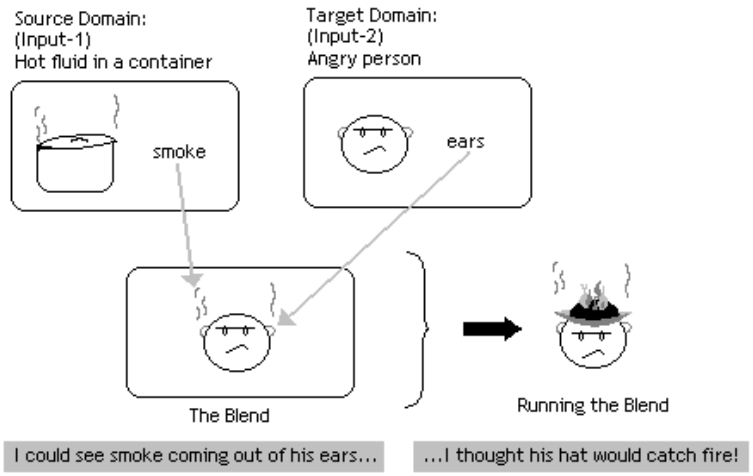
[46] James Geary, *I Is an Other: The Secret Life of Metaphor and How It Shapes the Way We See the World* (New York: Harper, 2011).

[47] Geary.

[48] Keith J. Holyoak and Paul Thagard, "The Analogical Mind," *American Psychologist*, no. 52 (1997): 35–44.

[49] Zoltan Kovecses, "Cognitive Linguistics," School of English and American Studies, Eötvös Loránd University Budapest, 2013, http://seas3.elte.hu/VLlxx/kovecses.html.

[50] Kovecses.



In this blend, hot fluid in a container represents the source domain (HEAT) and an angry person represents the target domain (EMOTION). By equating an abstract emotion (ANGER) to a container that is boiling over, the abstract idea becomes relatable to everyday experience.

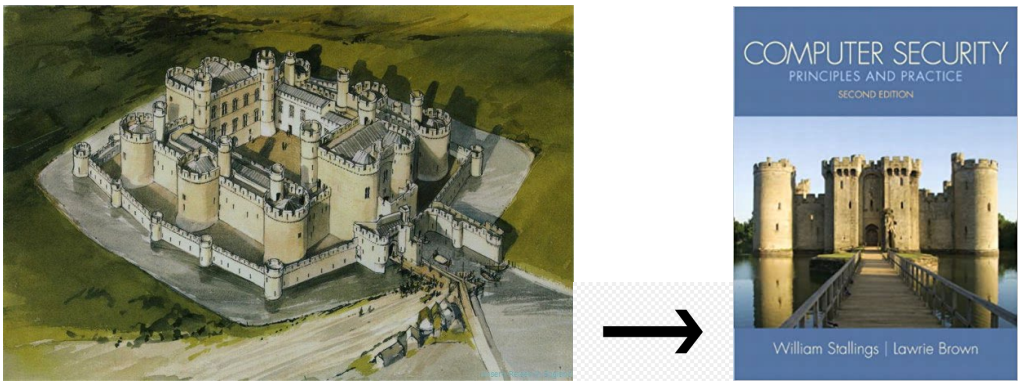
Figure 1. Source and Target Explained.[51]

The processes of operating systems also provide examples for Kovecses' domains. Readers link the forced end of life by another through the concept of kill; UNIX systems likewise have a command to force the end of all processes immediately and without any warning to users. This terminal command, kill -9 PID, draws the metaphor of a program ending its life cycle from a source domain rooted in biology to support the abstract idea of ending a process into the target domain of computer science; linking kill to the forced ending of a process and death with termination of a system's function allows the command to be easily understood (and implemented with great care by savvy users!).

LaPointe cautions that a "good metaphor's strengths, however, are also its weaknesses: a metaphor which grabs us…can also restrict our thinking by framing the discussion so effectively that we fail to question our vantage point."[52] Kovecses' idea thus has a negative implication: if the source and target do not align, the metaphor will fail to convey useful information. Consider Figure 2, demonstrating the persistent appeal of castle as computer security in the literature:

[51] Adapted from Kovecses, "Cognitive Linguistics."

[52] Lapointe, *When Good Metaphors Go Bad*.



Castles capture the imagination as strongholds against attack and serve as a source domain (SECURITY) for an abstract target (CYBERSECURITY). The metaphor is pervasive in computer science and is used in textbooks, apps, and icons to refer to security concepts.

Figure 2. Castle as Metaphor.[53]

Metaphor provides both useful context about unknown ideas, but an absence of useful information (past failures, constraints, or parameters) limits thought. Wendy Holliday, Dean of Library at Weber State University, notes that "[m]etaphors can also 'break.' In some cases, they do not explain 'enough,' or with enough clarity, to be useful."[54] Without a shared culture or a clear understanding of a metaphor's origins, Holliday warns that a metaphor that does not capture the relationship between source and target will carry weaknesses along with it.[55] The failures of defense-in-depth at Natanz provide a significant insight for security researchers: imprecision in an apt metaphor may appear to provide coverage for a problem without actually doing so or conceal significant security flaws that may be exploited. Consider a security team focused only on defense-in-depth; they may secure the network system from intrusion and isolate the system from any outside interference and still fail to protect the system from insider threat.

[53] Adapted from Woodside, "Defence in Depth" (left image); William Stallings and Lawrie Brown, *Computer Security: Principles and Practice*, 2nd ed. (London: Pearson, 2011), cover page (right image).

[54] Wendy Holliday and Northern Arizona University, "Frame Works: Using Metaphor in Theory and Practice in Information Literacy," *Comminfolit* 11, no. 1 (2017): 4, https://doi.org/10.15760/comminfolit.2017.11.1.44.

[55] Holliday and Northern Arizona University, 4.

Dan Coats, the Director of National Intelligence for the United States has "compared the cyber threat today with how U.S. officials said before 9/11 that intelligence channels were 'blinking red' with warning signs that a terror attack was imminent."[56] There is a real threat to United States infrastructure, known to be vulnerable to exploit and attack, from both intrusion and cyberattack by non-state actors, near-peer competitors, or adversary nations.[57] Even so, "today's computer systems pose individual and communal dangers that we'd never accept in more concrete structures like bridges, skyscrapers, power plants, and missile-defense systems," according to Ian Bogost, writing for *The Atlantic.*[58] To shore up defense-in-depth, we must look to other domains and find novel metaphors to conceptualize threats and develop solution sets unbounded by the dominant metaphor.

[56] Deb Riechmann, "Intel Official: Cyber Threat Warnings 'Blinking Red,'" *Military*, July 14, 2018, https://www.military.com/daily-news/2018/07/14/intel-official-cyber-threat-warnings-blinking-red.html.

[57] Riechmann.

[58] Ian Bogost, "Programmers: Stop Calling Yourselves Engineers," *The Atlantic*, November 5, 2015, https://www.theatlantic.com/technology/archive/2015/11/programmers-should-not-call-themselves-engineers/414271/.

THIS PAGE INTENTIONALLY LEFT BLANK

III. NOVEL METAPHORS IN DIVERSE SOURCE DOMAINS PROVIDE ADDITIONAL SOLUTION SETS

Novel metaphors that juxtapose machine thinking with biological evolution break through the implied limitations of difference engines or adding machines and allow a more nuanced understanding of a programmable, learning computer system. Known for finding a literal bug in her programming, Rear Admiral Grace Hopper began to demand that her staff and students think like a pirate and question assumptions; a rare thing for career naval officers![59] Hopper pioneered the view of computers that could compile code, assemble instructions, and execute those instructions without being physically reset by human operators: in her mind, those computers were less like machines and more like organisms that could learn and evolve.

Like Hopper, Cohen found inspiration for his code's function in the study of disease, which can grow, evolve, or mutate to survive and spread through a host. By looking to new source domains, he suggested a solution set unbounded by the constraints of computer science in a machine capable of interpreting instructions and executing programs. Cohen and Hopper drove much of computer science's evolution: dampened by a period of strict orthodoxy, they were among the first of many pioneers in the field to challenge the accepted orthodoxy and find solution sets unbounded by previous assumption or bias. Despite the successes yielded by novel metaphors, an exhaustive search in the literature failed to reveal any discussion as to why they were rarely employed after computer science matured in the late 1990s and early 2000s. After studying examples and analogues in the biological sciences, I propose, in this chapter, adopting ideas from a wider range of disciplines—from applied linguistics to zoology—to reinvent the lenses, tools, and language used to address the wicked problem of protecting computer systems.

To researchers like Stephanie Forrest and Catherine Beauchemin, computer science professors who champion the need for proactive network defense mechanisms, ideas from immunology are a natural complement to existing defense-in-depth models. Cohen's virus

[59] Walter Isaacson, "Grace Hopper, Computing Pioneer," *Harvard Gazette* (blog), December 3, 2014, https://news.harvard.edu/gazette/story/2014/12/grace-hopper-computing-pioneer/.

infects a system in the same way that microbes serve as a pathway to infection (frequently harmful to the host organism). An immunological metaphor recognizes that a system may have microbial attackers lurking inside the system and proactively searches for them. If a computer system were viewed as an organism and the metaphor of infectious disease were used in addition to defense-in-depth, the threat of infection from a pathogen or microbe from inside the network would pose a significant and obvious threat missed under defense-in-depth alone.

The previous chapter illustrates physical security metaphors cannot stand alone to detect and counter all known threats; this chapter proposes adopting additional source domains to reframe problems in cybersecurity. In an opinion piece describing the need to adopt new metaphorical lenses in computer science, Forrest writes "many of us don't recognise just how much we can learn by thinking more deeply about the biology."[60] Following Forrest's line of reasoning, biological sciences, from immunology to malacology, provide excellent teaching points for cybersecurity. In exploring these fields, I found several examples in which a novel metaphor from a source domain unrelated to computer science provided an unconstrained solution set for a cybersecurity problem.

A. IMMUNOLOGY

In a network, intersection between devices creates the opportunity for malicious code to spread in the same manner as an infectious disease, meaning a single infection on one computer can rapidly turn into an epidemic.[61] Cohen's view of the virus as disease was shared by Kephart, who suggested in 1993 that adopting ideas from immunology would allow a computer disease to be treated like any other disease.[62] With this in mind, immunology may serve as a source domain rich in ideas useful in cybersecurity applications.

---

[60] Stephanie Forrest, "Biology and Computers: Drawing Parallels between Immunology and Cyber-Security," *SC Media UK*, February 23, 2017, https://www.scmagazineuk.com/opinion/biology-and-computers-drawing-parallels-between-immunology-and-cyber-security/article/637267/.

[61] Kim Zetter, "Nov. 10, 1983: Computer 'Virus' Is Born," *WIRED*, November 10, 2009, https://www.wired.com/2009/11/1110fred-cohen-first-computer-virus/.

[62] Kephart, White, and Chess, "Computers and Epidemiology."

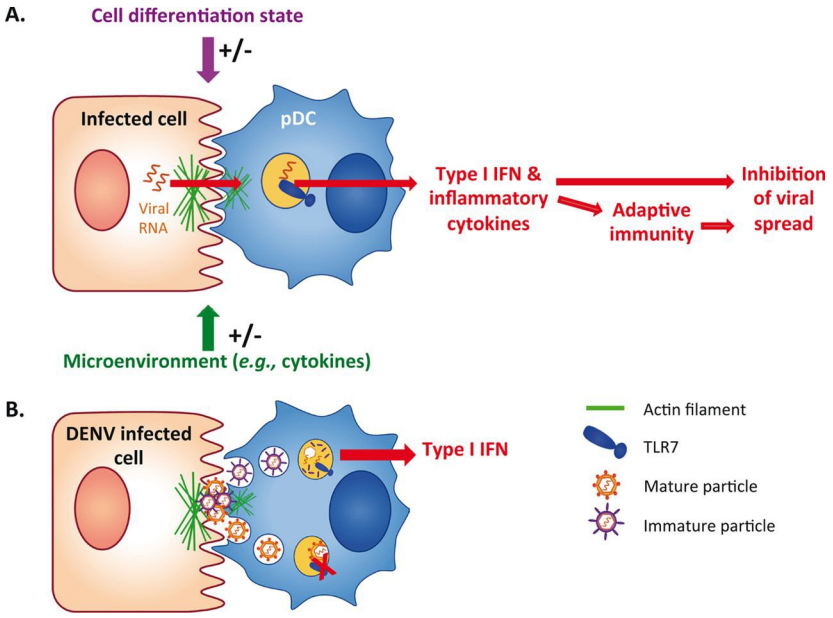
Madeline Drexler, in her text on infectious diseases, explains how microbes survive in some of the most hostile conditions imaginable by rapidly spreading and replicating throughout a host. "The vast majority of microbes establish themselves as persistent 'colonists,' thriving in complex communities within and on our bodies," she writes, making it possible to survive external pressures (such as solar radiation in soil samples or immune response in biological samples).[63] Drexler further explains that any changes that create new intersections between microbes and people pave the way for disease-causing agents to enter our species.[64] Just as with computer infections, infections in organisms become inevitable once an organism interacts with its environment or other organisms that inhabit the environment. Forrest and Beauchemin, writing for *Immunological Reviews*, explain healthy organisms have a robust immune system that attempts to block infection and destroy infection once it takes hold. In the article, they describe "[a] key capability of the immune system is its ability to recognise dangerous novel foreign pathogens and control their damage. At the same time, it must also avoid attacking the body, known as 'self,' in what is known as autoimmunity."[65]

Likewise, computer systems must eradicate harmful code while avoiding any change to code that provides a computer's basic functions. Computer systems manage to recognize harmful code by using signature-based systems, just as immune systems recognize novel pathogens in Figure 3.

[63] Madeline Drexler, *What You Need to Know about Infectious Disease* (Washington, DC: National Academies Press, 2010), https://www.ncbi.nlm.nih.gov/books/NBK209710/.

[64] Drexler.

[65] Stephanie Forrest and Catherine Beauchemin, "Computer Immunology," *Immunological Reviews* 216, no. 1 (April 2007): 176–197, https://doi.org/10.1111/j.1600-065X.2007.00499.x.



Cells differentiate between "safe" genetic code and unknown pathogens by comparing types of proteins. The act of distinguishing between "known safe," "known harmful," and "unknown" provides inspiration to cybersecurity applications that compare hashes or lines of code to allow known "safe" code to run and prevent unknown or harmful code from executing.

Figure 3. Pathogen Identification.[66]

Signature-based systems work by scanning code for identifiable strings that correspond with previously seen threats, erasing code that appears on a list of known threats (or blacklist) and retaining code known to be required for a computer's function (whitelist). The theory behind signature-based systems is sound, but the implementation is flawed for two reasons.

The first flaw is an assumption that one system is aware of all known threats. This is not possible, as individual blacklist providers cannot compare lists. If one were to compare multiple blacklists to determine whether the list contains all known variants of the virus, one would immediately reveal the threats the blacklist does not protect against:

> [M]ost blacklist providers are engaged in essentially a battle of wits with adversaries, and the providers cannot reasonably disclose the precise procedure of generating the lists without the risk of compromising the

[66] Adapted from Brian Webster, Sonia Assil, and Marlène Dreux, "Cell-Cell Sensing of Viral Infection by Plasmacytoid Dendritic Cells," *ASM Journals, Journal of Virology* 90, no. 22 (October 28, 2016): 10051, https://doi.org/10.1128/JVI.01692-16.

> quality of the list […] There is no known comparison of existing blacklists in the open literature.[67]

Worse, the files needed to operate—the so-called "whitelist"—is known to every user using the scanning software. With this knowledge, malicious code may masquerade as a necessary file or conceal itself piecemeal in multiple files.

The second flaw is the use of a singular detection strategy. The most sophisticated malicious code will spread from file to file as scans are run and remain undetected in a system, defeating any effort to eradicate it using a signature-based file. This virus behavior is novel in computer science but well known in immunology; the behavior is very common in vertebrate diseases and some types of cancer. Forrest and Beauchamin elaborate how the vertebrate immune system provides an elegant solution to the limits of signature-based detection: use multiple detection strategies. Unlike computer systems, "the vertebrate immune system uses [two] strategies, relying on anomaly detection to identify novel pathogens (zero-day attacks), and on signature detection to respond quickly and aggressively to previously seen threats," they write.[68]

Pier Luigi Gentili, a professor at the University of Perugia studying immune networks, finds that a complex system like the Internet "acts in a self-organizing manner and generates memory effects. Immune network algorithms have been used in clustering, data visualization, control, and optimization domains" with great success.[69] Without definitional boundaries to restrict thinking, computer scientists are free to match up any number of ideas from immunology or biology and correct the flaws that would otherwise remain undiscovered. Leveraging algorithms designed to study immune systems is one of many instances in which adopting metaphors from immunology unbounded by the constraints, assumptions, and biases of computer science yields surprising and effective solution sets. I propose that metaphors drawn from a study of vertebrate immune systems

---

[67] Leigh Metcalf, Jonathan M. Spring, and CERT Network Situational Awareness, *Everything You Wanted to Know about Blacklists but Were Afraid to Ask*, Publication CERTCC-2013-39 (Pittsburgh, PA: CERT, 2013), 309.

[68] Forrest and Beauchemin, "Computer Immunology," 176–197.

[69] Pier Luigi Gentili, *Untangling Complex Systems: A Grand Challenge for Science*, 1st ed. (Boca Raton, FL: CRC Press, 2018).

may ameliorate weaknesses in the defense-in-depth model, especially when shared observations of a new viral strain by a large pool of researchers results in rapid documentation and proactive guidance.

B. MALACOLOGY

Using solution sets from other fields, even those wholly unrelated to cybersecurity—like malacology, the study of invertebrate mollusks like snails—may yield promising results. Consider the mystery of the so-called "love dart." For millennia, the foremost minds in malacology have been baffled by the function of a calcium projectile that snails shoot into one another during mating. A team of scientists led by Michael Stewart notes that "love dart activity has been documented in the literature as far back as the mid-17th century, and love dart-possessing snails were known to the ancient Greeks, probably influencing the creation of the cupid myth."[70] As late as the 20th century, scientists still did not know the true purpose of the love dart. Two camps had emerged, both working to explain the biology through the mythological lens of Cupid's arrow.

According to Janet Leonard, a researcher at the University of California, Santa Cruz, the models of these two camps, "energy investment and 'gamete-trading,' make opposite predictions as to [the love dart's] function. Eggtrading predicts that it represents a gift of calcium to induce a partner to act as a female. The gamete-trading model predicts that it should serve to induce the partner to act as male."[71] These ideas, long considered orthodoxy in malacology, were flawed. In this case, applying ideas from a mythology source domain (Cupid's arrow) to a biological target domain (love dart) misdirected and hindered naturalists and biologists for years. Rather than examine the function of the love dart through other lenses, like serology and genetics, generations of malacologists attempted to fit their understanding into the mythological model of their Greek forebears. In Greek mythology, Cupid offers Psyche a multitude of gifts to induce her to remain his

[70] Michael J. Stewart et al., "A 'Love' Dart Allohormone Identified in the Mucous Glands of Hermaphroditic Land Snails," *Journal of Biological Chemistry* 291, no. 15 (April 8, 2016): 7938–50, https://doi.org/10.1074/jbc.M115.704395.

[71] Janet Leonard, "The 'Love-Dart' in Helicid Snails: A Gift of Calcium or a Firm Commitment?" *Journal of Theoretical Biology* 159, no. 4 (December 21, 1992): 513–21, https://doi.org/10.1016/S0022-5193(05)80695-2.

wife; the eggtrading camp drew on this mythological story to suggest the love dart is an inducement to act as a female, providing needed calcium to reproduce. For the gamete-trading camp, the love dart served as Cupid's arrow and induced the snail to mate more vigorously. This metaphor can also be used in computer science applications to explain the effect of malicious code; the malicious code reprograms a computer's function to allow the propagation or reproduction of additional harmful code.

In 1995, however, a research team led by Kazuki Kimura made a breakthrough discovery. Kimura's team discovered the calcium projectile is a delivery mechanism for a mucous coating, which "increases sperm storage…[S]nail pairs injected with mucus subsequently mated less often than control pairs."[72] As snails are hermaphroditic, there is biological advantage in reprogramming another snail's genes to accept a female role in reproduction; the snail that is not impregnated may continue to reproduce. Later researchers, including Monica Lodi and Joris Koene, found the practice caused significant injury and decreased the life span of snails, but "despite these injuries, hitting and being hit by the dart seems to be a standard component of mating…The partners are hit continuously, which is inevitable if they are both motivated to continue mating."[73] Figure 4 demonstrates the process.

[72] Kazuki Kimura, Kaito Shibuya, and Satoshi Chiba, "The Mucus of a Land Snail Love-Dart Suppresses Subsequent Matings in Darted Individuals," *Animal Behavior* 85, no. 3 (March 2013): 631–55, https://doi.org/10.1016/j.anbehav.2012.12.026.

[73] Monica Lodi and Joris M. Koene, "The Love-Darts of Land Snails: Integrating Physiology, Morphology and Behaviour," *Journal of Molluscan Studies* 82, no. 1 (February 1, 2016): 1–10, https://doi.org/10.1093/mollus/eyv046.

Snails are hermaphroditic, but female reproduction function is optimized when a snail is stabbed with a partner's so-called "love dart." The snail on the left attempts to stab the snail on the right with a love dart. The love dart allows for hormones to enter the injured snail's bloodstream and prevent the snail from producing spermicide. If successful, the injured snail's genetic code will be altered and the snail's eggs will be fertilized.

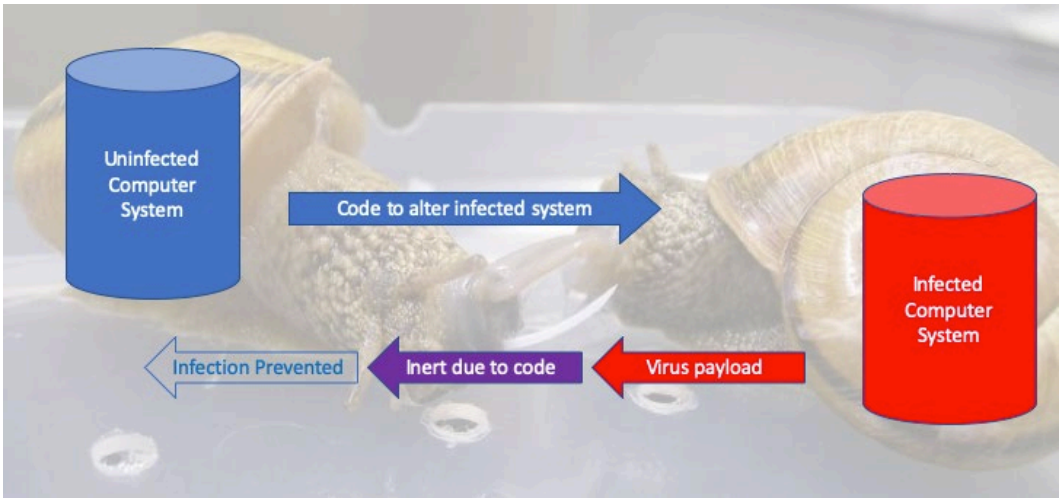
Figure 4. Snail with Love Dart.[74]

Understanding the purpose of the love dart may reveal a potential defense to a little-known vector for network intrusion. Consider a common practice in business, where trusted users have access to a trusted network. The trusted user must connect to a network outside the trusted network and afterward return to the trusted network. Using the castle metaphor, our user has returned safely to the castle and no threat is perceived or detected; no cyber risk is visible in the physical security lens. Introducing unsecure or infected devices under a bring-your-own-device policy is one of the main vectors for circumvention in security systems but does not always raise suspicion in a physical security model.[75] If one were to imagine these devices to be less like the hostile invaders envisioned in defense-

[74] Adapted from Ralph Martins, "Love Hurts: What Happens When Snails Stab Their Mates," March 10, 2015, https://www.nationalgeographic.com/animals/article/150310-snails-reproduction-sex-animals-science-evolution.

[75] Tao Xie et al., "Science of Human Circumvention of Security," Information Trust Institute, 2019, http://publish.illinois.edu/science-of-security-lablet/science-of-human-circumvention-of-security/.

in-depth and more like a snail in the garden—simultaneously receptive to data and potentially harmful—a new idea for security may emerge. Snail biology assumes that mating as inevitable; Greenberg's statistics on computer infection likewise assumes infection is inevitable. Snails have adopted a mechanism to force attempted mates to alter biology; computer defenses could use a similar idea to encrypt data on a connecting machine, rendering any attempt to inject a virus useless. Figure 5 provides an example of how such a system may protect against infection.



The snail on the left injects new genetic code into the snail on the right; the overlay shows a similar process with computer systems.

Figure 5. Snail/Computer System Genetic Code Updates

The imprecise metaphorical lens used to conceptualize the love dart hindered earlier naturalists, but Kimura approached the problem by examining the composition of a snail's mucus rather than the purpose of its delivery method. When his team realized what the mucus did, it reframed their ideas of delivery mechanism. By reframing his problem set with new ideas from a broader array of source domains, Kimura could work outside the constraints of the metaphor that applied to the love dart and seek ideas in new fields—fields not influenced by a mythological lens or labels drawn from mythology—to consider alternate explanations for the love dart's purpose.

Likewise, defenses built around a castle metaphor alone do not anticipate polymorphic threats capable of rewriting code. In a computer virus, an infection begins with a small change in the code, which in turn creates a cascading effect of greater and more virulent infection, until the target system is controlled entirely by the virus.[76] A computer virus rewriting the code running an operating system can be analogized to a snail injecting its partner with gene-editing mucus. Snails and computers both share hermaphroditic properties: snails both receive and pass on genetic information, and computers both receive and pass on genetic information. Both snails and computers have an interest in passing on their own information over others: snails wish to mate frequently and share genetic materials with other snails but cannot do so if impregnated, while computers wish to share and exchange information but cannot do so if compromised by a computer virus or controlled by an outside server. Taking ideas from malacology and applying them to computer science may suggest new defenses unimaginable if viewed solely through the metaphorical lens of a castle or with only the constraints of defense-in-depth in mind.

C. ARACHNOLOGY

Defense-in-depth is designed to protect computer systems from intrusion, but very little attention is paid to the emissions of heat or radiation that computer systems generate through normal operation. The lack of attention paid to emissions in complex systems may prevent system designers from understanding and protecting against the collection of data and interception of signals; Charles Darwin faced a similar problem when he was stymied by spider flight during his voyages of discovery. When Darwin sailed along the South American coast during his 1832 voyage aboard the HMS Beagle, he encountered thousands of tiny spiders on the deck. Darwin believed the spiders somehow ballooned in from Argentina but had no idea how the spiders came to be stowaways aboard the ship. Spiders are not biologically capable of flight, the wind was relatively light and should not have carried them aloft, spiders have no reason for flight, and the speed and distance of flight

[76] Cohen, "Computer Viruses."

should have been impossible.[77] In 2018, scientists Erica Morley and Daniel Robert performed tests on spiders at the University of Bristol and discovered that currents of static electricity—rather than currents of air—allow spiders to balloon themselves over incredible distances.[78] Morley and Robert conducted their research to solve a centuries-old debate about the flight mechanics of spiders and satisfy their own curiosity; they noted that Darwin failed to study the phenomenon in any detail and wondered why. Morley and Robert comment that because Darwin never explored the idea further, "two competing hypotheses were proposed [by the scientific community of the day] to explain how ballooning animals become airborne, invoking (1) the aerodynamic drag from wind acting on the silk or (2) atmospheric electrostatic forces."[79] Darwin's choice of metaphor—a spider "ballooning" through the air—led generations to assume that spiders somehow used wind or thermal energy, much like early balloonists in the 19th century.

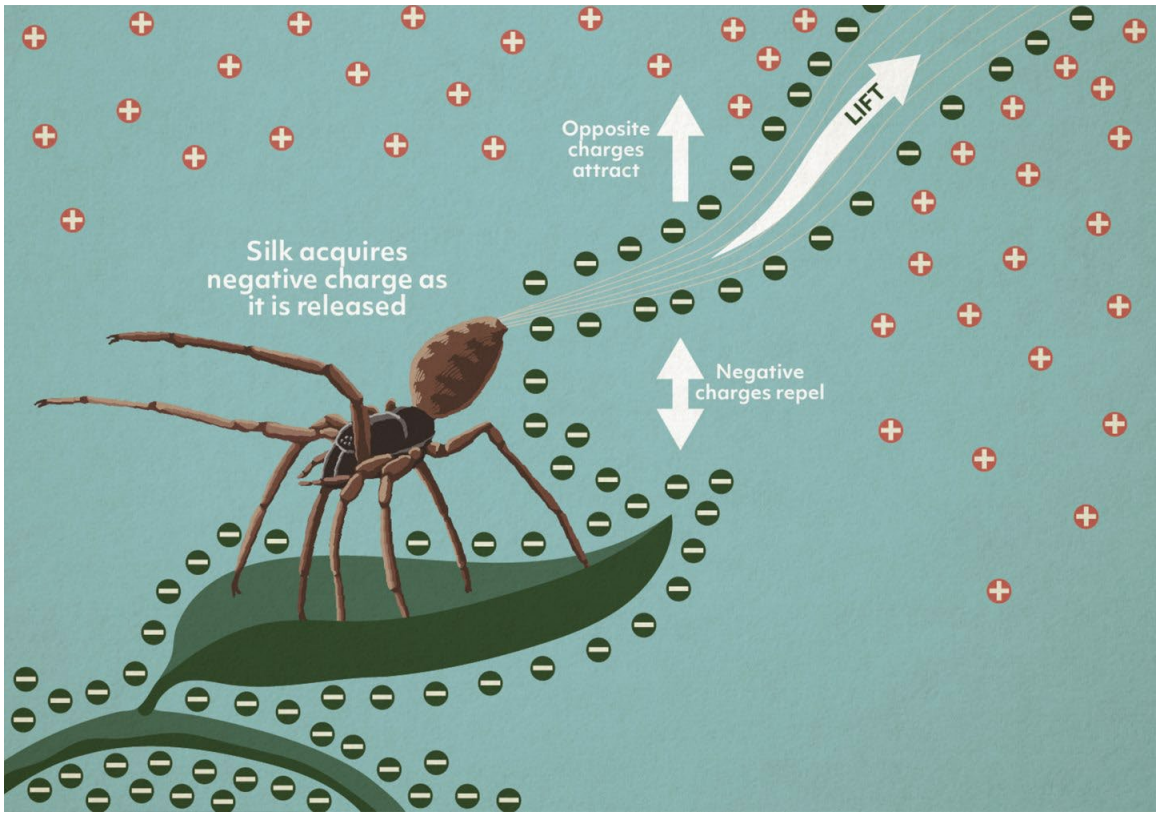
Morley and Robert, however, concluded that the observations of other scientists focused too heavily on the spider when studying ballooning: spiders were able balloon in periods of still wind, on overcast days, and in other conditions that should not be possible given the aerodynamic drag explanation.[80] By focusing on the environment and the conditions—rather than fixating on the dominant metaphor of a balloon and the mechanism of flight produced by the spider—Morley and Robert found conclusive proof that atmospheric electrostatic forces allow spiders to take flight, as shown in Figure 6.

[77] Ed Yong, "Spiders Can Fly Hundreds of Miles Using Electricity," The Atlantic, July 5, 2018, https://www.theatlantic.com/science/archive/2018/07/the-electric-flight-of-spiders/564437/.

[78] Yong.

[79] Erica L. Morley and Daniel Robert, "Electric Fields Elicit Ballooning in Spiders," *Current Biology* 28, no. 14 (July 23, 2018): 2324–2330.e2, https://doi.org/10.1016/j.cub.2018.05.057.

[80] Morley and Robert.



Contrary to early ideas in arachnology, spiders do not "balloon" on air. Instead, spiders use negatively charged silk attracted to charged particles in the air to propel themselves long distances. The act shares more in common with magnetic levitation than hot air balloons; the metaphor of ballooning limited study of spider flight for centuries.

Figure 6. Spider Flight Explained.[81]

Their methodology is instructive; scientists were baffled by hacking attempts at fully secured and air-gapped computer systems until a 2018 article in WIRED magazine demonstrated how "continuous stream[s] of data over the multi-channel memory buses on a computer" and unshielded radio wave emissions could be interpreted and deciphered.[82] A so-called "unhackable" computer isolated from all other systems still generates heat and noise; it is a trivial matter to program code that spins up cooling fans or increases processing speed to generate varying levels of heat and/or noise in a pattern that may be

[81] Adapted from Kathryn Krupin, "Spiders Fly Riding Electric Currents," last updated February 9, 2021, https://asknature.org/strategy/spiders-surf-on-electric-fields/.

[82] Kim Zetter, "Researchers Hack Air-gapped Computer with Simple Cell Phone," WIRED, July 27, 2015, https://www.wired.com/2015/07/researchers-hack-air-gapped-computer-simple-cell-phone/.

deciphered. A simple binary pattern would allow reliable (if slow) exfiltration of data undetected by researchers focused on the computer system rather than the environment around the computer system. As demonstrated by these examples, Cohen, Forrest, and Beauchemin are not alone in adopting new ideas from other domains to explore wicked problems of cybersecurity. Each of the domains described in this chapter provide examples of novel metaphors that—if adopted—allow for augmentation of the dominant defense-in-depth model.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. SYNTHESIS

LaPointe warns that a metaphor that is adopted without careful observations and analysis of its vulnerabilities hinders imagination.[83] When parts that fit well diminish perception of the parts that do not, assumptions fill in areas that conceal a dangerous gap. As Woodside argues, the castle metaphor works in many ways to represent a strong defense against the perceived threat to computer systems.[84] Drawn from the castle metaphor, defense-in-depth denies most attackers the route into a computer system to start an infection and thus appears to provide excellent protection against network penetration. The threat of infection, however, is not entirely prevented by defense-in-depth because the metaphor of castle constrains system designers to look outwards—rather than inwards—at threats, which in turn leaves systems vulnerable to zero-day exploits and novel Stuxnet-like attacks.

Recall a castle metaphor seeks to keep out one attacker, with no action taken until an outside threat appears; a castle metaphor forces the solution set to take exactly the form of the problem set imagined by the metaphor. The fatal flaws in the Natanz system and similar networks, carried over from the castle metaphor used in the development of a defense-in-depth security architecture, may be mitigated by the inclusion of novel metaphors from biological and immunological source domains in a solution set. Understanding of one domain, such as the electromagnetic spectrum, informs understanding in another domain; application of ideas from a broad sets of source domains allows the development of more precise metaphors in abstract fields. In this chapter, I propose using multiple metaphors to conceive risks at different levels and augmenting dominant metaphors with those from emergent fields to resolve complex cybersecurity problems.

[83] Lapointe, *When Good Metaphors Go Bad*.

[84] Woodside, "Defence in Depth."

A. USING IT ALL TOGETHER

Edward Wilson, a myrmecologist and avid reader of 19th century theologian and scientist William Whewell, adopted a method of consilience—using "facts and fact-based theory across disciplines to create a common groundwork of explanation"—to solve complex problems in a fashion relevant and instructive today.[85] Wilson encouraged borrowing metaphors in one field to an unrelated other field to explore shared frameworks of understanding between target and source domains and breaking down barriers between disciplines. His unorthodox approach allowed for a startling discovery: butterfly flight was influenced by electromagnetic energy. Wilson wrote "with the aid of appropriate instruments we can now view the world with butterfly eyes. Scientists have entered the visual world of animals and beyond because they understand the electromagnetic spectrum."[86]

Unexplored or uncharted domains—like cybersecurity—requires seeking out truth in other, ostensibly unrelated fields. Just as Wilson found answers to complex problems in the synthesis of multiple domains, combining ideas from physical security (using castles to model defense-in-depth), immunology (comparing anomaly response in user permissions to normal function, as immune systems detect abnormal behavior in cells), malacology (injecting a proactive vaccine into an external host), and arachnology (studying the environment to better understand mechanism of intrusion and infection), may allow breakthrough ideas in computer science. Wilson warns in his work on consilience that "medical researchers are locked in an arms race with the rapidly evolving pathogens that is certain to grow more intense. They are obliged to turn to a broader array of wild species in order to acquire new weapons of medicine in the twenty-first century."[87] The puzzles of love dart function and spider flight were solved by a different set of ideas because the ideas that formed solutions did not come from the set that contributed to the problem: the limits of perception or understanding that come from imprecise or inapt metaphor. Wilson's

[85] Edward Wilson, *Consilience: The Unity of Knowledge* (New York: Vintage: 1999), 15.

[86] Wilson, 57.

[87] Wilson.

mention of wild species in medicine suggests that computer defenses may find ideas in new source domains, such as immunology, epidemiology, arachnology, malacology, and a myriad of others. Abstractions in cybersecurity likewise require the use of a metaphor to conceptualize threats and develop solutions.

By understanding how to draw ideas and inspiration from other fields, like Wilson's example of scientists seeing with butterfly eyes, one may connect new source ideas to the cybersecurity target domains and mitigate the dangers Bogost warns against. Consider a team that adopts both the castle metaphor and a butterfly eyes metaphor: using the same tools adopted from electrical engineers to look at the computer system's emissions. Computers move data from a hard disk to memory and back again and create electromagnetic emissions in the process. These emissions are not perceptible to human sight or hearing but would be easily detected by butterflies (electromagnetic sight). Looking at the world through butterfly eyes allowed researchers, who developed a specially constructed receiver, to interpret these emissions as signals intelligence. Those signals are then decrypted, allowing researchers to exfiltrate data.[88] Understanding how these patterns are deciphered allows security researchers to consider new defenses: perhaps adding random electromagnetic interference or practicing emissions control as one would on a submarine. Defense-in-depth and the adoption of the castle metaphor led to an evolution in the computer virus, because defense-in-depth is effective. Just as the French realized they could not batter down the walls at Chateau Gaillard, virus writers realize they cannot penetrate a network directly. Instead, they must employ an approach common to castles and biological threats: find a way inside, then launch an attack. In turn, security researchers must be resourceful in using a myriad of novel metaphors to challenge the understanding of a dominant metaphor and augment existing ideas with fresh approaches. Failing to do so leaves vulnerabilities to be exploited, with potential results that include loss of sensitive data, classified information, or even physical damage to critical infrastructure.

[88] Zetter, "Researchers Hack Air-gapped Computer with Simple Cell Phone."

B. RAISING THE BAY

A key passage of Wilson's text on consilience explains "the key to the exchange between [domains] is not hybridization, not some unpleasantly self-conscious form of scientific art or artistic science, but reinvigoration of interpretation with the knowledge of science and its proprietary sense of the future."[89] As threats emerge, the best defense will reinterpret threats with metaphors from diverse and divergent source domains simultaneously, rather than treating those domains as mutually exclusive; those metaphors should come from voices in every field. This thesis introduces the use of metaphor to conceive risks and augment dominant metaphors in complex cybersecurity problems but does not fully explore the wide range of domains to be studied or present solutions to be implemented. Future research in areas of epidemiology (drawing experience from contact tracing and social distancing to limit the spread of the computer virus), machine learning (comparing and consolidating blacklist and whitelist data sets), and human interface (resolving exploits related to computer systems that inherently trust human input) may mark the beginning of a new and exciting period in computer science and a return to the spirit of innovation and improvisation embraced by Hopper and Cohen.

With reinterpretation and reimagining driving the potential for an unbounded solution set, I hope this thesis reinvigorates and redoubles the commitment to search for answers to the wicked problems of the computer virus in diverse fields and numerous domains.

[89] Wilson, *Consilience*.

LIST OF REFERENCES

Aristotle. *Poetics* 21, 1457b, 6–7.

Bachrach, Bernard S. "Medieval Siege Warfare: A Reconnaissance." *Journal of Military History* 58, no. 1 (January 1994): 119. ProQuest.

Bogost, Ian. "Programmers: Stop Calling Yourselves Engineers." *The Atlantic*, November 5, 2015. https://www.theatlantic.com/technology/archive/2015/11/programmers-should-not-call-themselves-engineers/414271/.

Byres, Eric J. "Defense in Depth." *InTech; Durham* 59, no. 6 (December 2012): 38–40. ProQuest.

Cohen, David. "Ditching the Air-Gapping Myth. Power-Grid." July 23, 2017. https://www.power-grid.com/td/ditching-the-air-gapping-myth.

Cohen, Fred. "Computer Viruses: Theory and Experiments." *Computers & Security* 6, no. 1 (1987): 22–35.

Colburn, Timothy R., and Gary M. Shute. "Metaphor in Computer Science." *Journal of Applied Logic* 6, no. 4 (December 2008): 526–33. https://doi.org/10.1016/j.jal.2008.09.005.

Conway, Bevil R., and Ted Gibson. "Languages Don't All Have the Same Number of Terms for Colors—Scientists Have a New Theory Why." *The Conversation*. Accessed June 2, 2018. http://theconversation.com/languages-dont-all-have-the-same-number-of-terms-for-colors-scientists-have-a-new-theory-why-84117.

Drexler, Madeline. *What You Need to Know about Infectious Disease*. Washington, DC: National Academies Press, 2010. https://www.ncbi.nlm.nih.gov/books/NBK209710/.

Dumas, Alexandre. *The Count of Monte Cristo*. New York: Penguin Books, 2001.

Forrest, Stephanie. "Biology and Computers: Drawing Parallels between Immunology and Cyber-Security." *SC Media UK*, February 23, 2017. https://www.scmagazineuk.com/opinion/biology-and-computers-drawing-parallels-between-immunology-and-cyber-security/article/637267/.

Forrest, Stephanie, and Catherine Beauchemin. "Computer Immunology." *Immunological Reviews* 216, no. 1 (April 2007): 176–197. https://doi.org/10.1111/j.1600-065X.2007.00499.x.

Frincke, Deborah A., and Matt Bishop. “Guarding the Castle Keep: Teaching with the Fortress Metaphor.” *IEEE Security & Privacy Magazine* 2, no. 3 (May 2004): 69–72. https://doi.org/10.1109/MSP.2004.13.

Geary, James. *I Is an Other: The Secret Life of Metaphor and How It Shapes the Way We See the World*. New York: Harper, 2011.

Gentili, Pier Luigi. *Untangling Complex Systems: A Grand Challenge for Science*. 1st ed. Boca Raton, FL: CRC Press, 2018.

Gibson, William. “25 Years of Digital Vandalism.” *New York Times*, sec. Opinion. January 27, 2011. https://www.nytimes.com/2011/01/27/opinion/27Gibson.html.

Granneman, Scott. “Infected in 20 Minutes.” August 19, 2004. https://www.theregister.co.uk/2004/08/19/infected_in20_minutes/.

Greenberg, Andy. “Shopping for Zero-Days: A Price List for Hackers’ Secret Software Exploits.” Forbes, March 13, 2012. https://www.forbes.com/sites/andygreenberg/2012/03/23/shopping-for-zero-days-an-price-list-for-hackers-secret-software-exploits/.

Hak5 Gear. “USB Rubber Ducky.” Accessed June 2, 2018. https://hakshop.com/products/usb-rubber-ducky-deluxe.

Holliday, Wendy, and Northern Arizona University. “Frame Works: Using Metaphor in Theory and Practice in Information Literacy.” *Comminfolit* 11, no. 1 (2017): 4–20. https://doi.org/10.15760/comminfolit.2017.11.1.44.

Holyoak Keith J., and Paul Thagard. “The Analogical Mind.” *American Psychologist*, no. 52 (1997): 35–44.

Isaacson, Walter. “Grace Hopper, Computing Pioneer.” *Harvard Gazette* (blog). December 3, 2014. https://news.harvard.edu/gazette/story/2014/12/grace-hopper-computing-pioneer/.

Jackson, Ronald L., and Michael A. Hogg. “Sapir-Whorf Hypothesis.” In *Encyclopedia of Identity*. Vol. 1, edited by Ronald L. Jackson and Michael A. Hogg, 652–654. Thousand Oaks, CA: SAGE Publications, Inc., 2010. https://doi.org/10.4135/9781412979306.n207.

Kephart, Jeffrey O., Steve R. White, and Dave M. Chess. “Computers and Epidemiology.” *IEEE Spectrum* 30, no. 5 (1993): 20–26.

Kimura, Kazuki, Kaito Shibuya, and Satoshi Chiba. “The Mucus of a Land Snail Love-Dart Suppresses Subsequent Matings in Darted Individuals.” *Animal Behavior* 85, no. 3 (March 2013): 631–55. https://doi.org/10.1016/j.anbehav.2012.12.026.

Kovecses, Zoltan. “Cognitive Linguistics.” School of English and American Studies, Eötvös Loránd University Budapest, 2013. http://seas3.elte.hu/VLlxx/kovecses.html.

Krupin, Kathryn. “Spiders Fly Riding Electric Currents.” Last updated February 9, 2021. https://asknature.org/strategy/spiders-surf-on-electric-fields/.

Kushner, David. “The Real Story of Stuxnet.” IEEE Spectrum: Technology, Engineering, and Science News, February 26, 2013. https://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet.

Lakoff, George, and Mark Johnson. *Metaphors We Live By*. Chicago: University of Chicago Press, 2003.

Lapointe, Adriane. *When Good Metaphors Go Bad: The Metaphoric ‘Branding’ of Cyberspace*. Washington, DC: Center for Strategic and International Studies, 2011. https://csis-prod.s3.amazonaws.com/s3fs-public/legacy_files/files/publication/110923_Cyber_Metaphor.pdf.

Leonard, Janet. “The ‘Love-Dart’ in Helicid Snails: A Gift of Calcium or a Firm Commitment?” *Journal of Theoretical Biology* 159, no. 4 (December 21, 1992): 513–21. https://doi.org/10.1016/S0022-5193(05)80695-2.

Levy, Steven. *Hackers: Heroes of the Computer Revolution*. Sebastopol, CA: O’Reilly Media, 2010.

Lodi, Monica, and Joris M. Koene. “The Love-Darts of Land Snails: Integrating Physiology, Morphology and Behaviour.” *Journal of Molluscan Studies* 82, no. 1 (February 1, 2016): 1–10. https://doi.org/10.1093/mollus/eyv046.

Martins, Ralph. “Love Hurts: What Happens When Snails Stab Their Mates.” March 10, 2015. https://www.nationalgeographic.com/animals/article/150310-snails-reproduction-sex-animals-science-evolution.

Metcalf, Leigh, Jonathan M. Spring, and CERT Network Situational Awareness. *Everything You Wanted to Know about Blacklists but Were Afraid to Ask*. Publication CERTCC-2013-39. Pittsburgh, PA: CERT, 2013.

Morley, Erica L., and Daniel Robert. “Electric Fields Elicit Ballooning in Spiders.” *Current Biology* 28, no. 14 (July 23, 2018): 2324–2330.e2. https://doi.org/10.1016/j.cub.2018.05.057.

Newman, Lily Hay. “Why Governments Won’t Let Go of Secret Software Bugs.” WIRED, May 16, 2017. https://www.wired.com/2017/05/governments-wont-let-go-secret-software-bugs/.

Patterson, Dan. “U.S. Grapples with Controlling ‘Cyber-Munitions’ While Recruiting 6,000 New Cyber-Warriors.” TechRepublic. Accessed June 3, 2018. https://www.techrepublic.com/article/u-s-grapples-with-controlling-cyber-munitions-while-recruiting-6000-new-cyber-warriors/.

Raines, Rebecca Robbins. *Getting the Message Through: A Branch History of the U.S. Army Signal Corps*. Washington, DC: Center for Military History, U.S. Army, 1996.

Riechmann, Deb. “Intel Official: Cyber Threat Warnings ‘Blinking Red.’” *Military*, July 14, 2018. https://www.military.com/daily-news/2018/07/14/intel-official-cyber-threat-warnings-blinking-red.html.

Savov, Vlad. “The Death of Garbage in, Garbage Out.” The Verge, August 16, 2016. https://www.theverge.com/2016/8/16/12499854/first-click-the-death-of-garbage-in-garbage-out.

Schneier, Bruce. “The Story behind the Stuxnet Virus.” Forbes, October 2010. https://www.forbes.com/2010/10/06/iran-nuclear-computer-technology-security-stuxnet-worm.html.

Smith, Brad. “The Need for Urgent Collective Action to Keep People Safe Online: Lessons from Last Week’s Cyberattack.” *Microsoft on the Issues* (blog). May 14, 2017. https://blogs.microsoft.com/on-the-issues/2017/05/14/need-urgent-collective-action-keep-people-safe-online-lessons-last-weeks-cyberattack/.

Sousa, Sonia, Paulo Dias, and David Lamas. “A Model for Human-Computer Trust.” In *9th Iberian Conference on Information Systems and Technologies*, 1–656. Barcelona, Spain, IEEE, 2014.

Stallings, William, and Lawrie Brown. *Computer Security: Principles and Practice*. 2nd ed. London: Pearson, 2011.

Sterling, Bruce. “The Dropped Drive Hack.” WIRED, June 29, 2011. https://www.wired.com/2011/06/the-dropped-drive-hack/.

Stewart, Michael J., Tianfang Wang, Joris M. Koene, Kenneth B. Storey, and Scott F. Cummins. “A ‘Love’ Dart Allohormone Identified in the Mucous Glands of Hermaphroditic Land Snails.” *Journal of Biological Chemistry* 291, no. 15 (April 8, 2016): 7938–50. https://doi.org/10.1074/jbc.M115.704395.

U.S. Congress. House of Representatives. *Hearing before the Subcommittee on Technology and Competitiveness of the Committee on Science, Space, and Technology*. 102nd Cong., 1st. sess., June 27, 1991. https://winnschwartau.com/wp-content/uploads/2019/06/Testimoney-1991-Computer-security_hearing.pdf.

Warner, Philip. *Sieges of the Middle Ages*. Havertown, PA: Pen and Sword, 2015.

Webster, Brian, Sonia Assil, and Marlène Dreux. "Cell-Cell Sensing of Viral Infection by Plasmacytoid Dendritic Cells." *ASM Journals, Journal of Virology* 90, no. 22 (October 28, 2016): 10050–10053. https://doi.org/10.1128/JVI.01692-16.

Wilson, Edward. *Consilience: The Unity of Knowledge*. New York: Vintage: 1999.

Woodside, Simon. "Defence in Depth: The Medieval Castle Approach to Internet Security." Medium, June 20, 2016. https://medium.com/@sbwoodside/defence-in-depth-the-medieval-castle-approach-to-internet-security-6c8225dec294.

Xie, Tao, Jim Blythe, Ross Koppel, and Sean Smith. "Science of Human Circumvention of Security." Information Trust Institute, 2019. http://publish.illinois.edu/science-of-security-lablet/science-of-human-circumvention-of-security/.

Yong, Ed. "Spiders Can Fly Hundreds of Miles Using Electricity." The Atlantic, July 5, 2018. https://www.theatlantic.com/science/archive/2018/07/the-electric-flight-of-spiders/564437/.

Zetter, Kim. "An Unprecedented Look at Stuxnet, the World's First Digital Weapon." WIRED, November 3, 2014. https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/.

———. "Nov. 10, 1983: Computer 'Virus' Is Born." *WIRED*, November 10, 2009. https://www.wired.com/2009/11/1110fred-cohen-first-computer-virus/.

———. "Researchers Hack Air-gapped Computer with Simple Cell Phone." WIRED, July 27, 2015. https://www.wired.com/2015/07/researchers-hack-air-gapped-computer-simple-cell-phone/.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California