National Defence Défense nationale

Canadian Forces College

Collège des Forces Canadiennes

THE CYBER *ENTSCHEIDUNGSPROBLEM*: OR WHY CYBER CAN'T BE SECURED AND WHAT MILITARY FORCES SHOULD DO ABOUT IT

Maj F.J.A. Lauzier

JCSP 41

Exercise *Solo Flight*

Disclaimer

Opinions expressed remain those of the author and do not represent Department of National Defence or Canadian Forces policy. This paper may not be used without written permission.

PCEMI 41

Exercice *Solo Flight*

Avertissement

Les opinons exprimées n'engagent que leurs auteurs et ne reflètent aucunement des politiques du Ministère de la Défense nationale ou des Forces canadiennes. Ce papier ne peut être reproduit sans autorisation écrite.

Canada

CANADIAN FORCES COLLEGE – COLLÈGE DES FORCES CANADIENNES
JCSP 41 – PCEMI 41
2014 – 2015

EXERCISE *SOLO FLIGHT* – EXERCICE *SOLO FLIGHT*

THE CYBER ENTSCHEIDUNGSPROBLEM: OR WHY CYBER CAN'T BE SECURED AND WHAT MILITARY FORCES SHOULD DO ABOUT IT

Maj F.J.A. Lauzier

Word Count: 4920

Compte de mots : 4920

INTRODUCTION

Nothing is an absolute and there is no certainty in this world, especially in the field of computer science. What holds true today could be easily proven wrong tomorrow without any look back to the old truths. One concept that has not been shown to be wrong is A.M. Turing theory on the fact that no computer program could prove that another program is free of bug[1]. Since this still hold true in that field, the customary extrapolation in other aspect of computer technology is again adequate.

Considering the axiom brought by this theory, that no computer program can be proven fault-free and that the cyber domain is constituted of numerous software programs, this essay thesis is then that it would be an unachievable objective to secure cyberspace in any way. Based on the fact that any bug could be exploited in order to modify the initial purpose of a program[2] and that a modification in a program could lead to it being used against our own military or governmental force, the current view of the occidental civilization on the cyber space is wrong. Securing completely this domain is thus a flawed conception and is bound to fail.

Three parts will be used to prove that thesis. The first one will explain Turing theory on mathematical machines, exposing how he proposed that no machine could prove another machine, and thus computer software, error-free. The concept of the

---

[1] A. M. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem" (Graduate, Princeton University, 1936), 36.
[2] Thomas M. Chen, Lee Jarvis and Stuart MacDonald, *Cyberterrorism: Understanding, Assessment, and Response* (New York: Springer, 2014), 34.

Entscheidungsproblem will also be covered in order to provide the depth required to analyze the issue.

Secondly, this essay will review the current emerging military doctrine governing the cyber environment. In this fashion, the nature of the usage of that domain will be exposed, showing that there is a constant psyche of securing it while denying its use by our adversary[3]. This will show us that the current mindset does not take into consideration Turing's theory and subsequently the idea that no computer system can be secured totally. We will also review some outlier on the subject to maintain objectivity, although they are few and between. Parallel will then be drawn between cyber-attacks in the private sector and the false sense of protection granted by software.

In the last section, this essay will explore ideas that the military could exploit or use in order to mitigate the situation exposed in the previous sections. Especially, it will be explained that the first step to use efficiently the cyber domain is to let go of the initial perception about its security and that the bulk of the work needs to be done culturally upon the leaders of the field[4]. Then it will explain what that paradigm shift could provide to the western military and security forces. This section will end be also providing some more technically oriented solutions to the whole concept while exposing modus operando of some hacker groups. The next section will thus be the foundation of this essay and will cover the history of the theory at the core of these ideas and why it matters today.

---

[3] Alan D. Campen, Douglas H. Dearth and R. Thomas Goodden, *Cyberwar: Security, Strategy, and Conflict in the Information Age* (Fairfax, Va.: AFCEA International Press, 1996), 106.
[4] United States. Defense Science Board. Task Force on Resilient Military Systems and the Advanced Cyber Threat, *Task Force Report: Resilient Military Systems and the Advanced Cyber Threat* (Washington, DC: Office of the Under Secretary of Defense for Acquisition, Technology and Logistics, 2013), 110.

The Entscheidungsproblem

The Entscheidungsproblem (decision problem in German) is a way to describe an algorithm that would take as an input a program and would provide as an output a true or false flag. That flag would represent if the program would be universally valid, meaning that the provided program would never encounter a state such as it is not being able to self-resolve. This concept was provided as a challenge by David Hilbert in 1928[5]. He thus contented that such a program should be able to exist and extended an invitation to the mathematician community to find it[6].

If such an algorithm could exist, it would then be possible to validate any algorithm that could be thought of, thus solving it without a doubt. The application of such a theory would be tremendous in the field of computer science. It would mean for example that computer software could be proven bug free by a faultless party, another computer program[7].

Working under that assumption, two independent mathematicians came up with proof that such a program could not exist within the law of mathematics. The first one of them is Alonzo Church, with his concept of effective calculability based on his lambda-calculus methodology[8]. Although this theory was sound and presented everything that was needed to prove the point in its thesis, it was unable to apply the theory to the realm

---

[5] Turing, *On Computable Numbers, with an Application to the Entscheidungsproblem*
[6] Andrew Hodges, *Turing: A Natural Philosopher,* Vol. III (London: Phoenix, 1997), 27.
[7] Andrew Hodges, *Alan Turing,* Centenary ed. (Princeton, N.J.: Princeton University Press, 2012), 362.
[8] Turing, *On Computable Numbers, with an Application to the Entscheidungsproblem*

of computer program and thus was only able to prove it within the realm of mathematical theorems[9]. The second mathematician to delve into the subject was Alan Turing.

Independently, Turing draw the same conclusion that Church, as such that a theorem that prove other theorem could not exist. In order to prove his point, he used a completely different theory. While Church used the lambda-calculus model, Turing created the concept of an alternate model of computation that he called an *Automatic machine* or *a-machine*, which evolved over time and changed name to Turing Machines[10]. While Church theory was applicable to the mathematical field, Turing's concept was more adaptive and was able to represent also a computer program. His machine was actually a virtual representation of a manipulation machine that moves symbol on a tape according to its programming. The Turing machine was not supposed to be a practical computer machine but instead a way for scientist to understand the different state a machine go into while responding to a program[11].

Once they both had published their works through their own universities, both Church and Turing continued to work on their theories and eventually joined forces, thus coining the Church-Turing algorithm in 1936. Finally, Turing completed his theory by expending it to take into consideration the halting problem, thus making it universal to any computer program[12]. The halting problem is the fact that no machine, either physical or theoretical, can discern between a long delay in computation or the simple fact that that the program is in an infinite loop state that will never conclude.

---

[9] Hodges, *Turing: A Natural Philosopher*, 23.
[10] A. M. Turing, "Solvable and Unsolvable Problems," *Science News* 31 (1954), 1-3-4.
[11] Hodges, *Turing: A Natural Philosopher*, 34.
[12] Hodges, *Alan Turing*, 217.

Thus reducing his whole theory to the halting problem, based on his Turing machines, Turing rested his case and was not proven wrong in the eighty years since the discovery of his theory[13].

In 1970, Yuri Matiyasevich, a Russian mathematician, presented his doctoral thesis at the Leningrad Department of the Steklov Institute of Mathematics (LOMI) on Hilbert's tenth problem. He thus proved that through a negative solution that Hilbert's problem was thus unsolvable[14]. This work was done independently of Turing's former work but served also to validate the claim of the former mathematician, even after his death. By solving the issue through a negative proof, Matiyasevich was demonstrating at the same time that Turing was right on his assertion that no machine was able to discern the real status of a halting state in a program[15].

As a counterpoint, deriving from Turing's concept was the attempt to create Automated Theorem Proving (ATP) machines or program in order to prove wrong Turing, Church and Matiyasevich concepts. By using artificial intelligence, the Logic Theory Machine team of Allen Newell, Herbert A. Simon and J. C. Shaw tried to achieve just this[16]. The breakthrough in computer logic of that team and the ones that carried on with their work was impressive and most significant in the field of artificial intelligence, computer science and mathematics. They were thus able to create a system that was able, with minimal user input, to prove a theorem right with the correct mathematic formulas to support the claim. Using heuristic guidance, the initial system was able to prove 38 out of

---

[13] Ibid, 348.
[14] Troy Vasiga, "How to Test Programs," University of Waterloo, https://cs.uwaterloo.ca/~tmjvasig/CS134Testing.html (accessed 04/21, 2015).
[15] Hodges, *Turing: A Natural Philosopher*, 46.
[16] Hodges, *Alan Turing*, 237.

the first 52 problems of the *Principia Mathematica,* the keystone document of logical theory[17].

Although the ATP was able to operate within its required parameters and proved immensely successful, there was no way in the whole of the research on the subject to take away the human aspect. There was always a need for a mathematical operator to enter logical hints unto the system in order to eliminate the complex solutions that would cause the evaluated machine to go into a suspended state. As the aim of solving the Entscheidungsproblem was to bring a standalone program, machine or mathematical theory that would be able to solve on its own another such mathematical statement, machine or program, even the ATP was unable to achieve that aim.

The opposing view on the concept will highlight that a simple program can be proven fault free. Although this is true, it will only be true if a human being does that demonstration. For simple programs, proof can be provided directly by a mathematician or a machine alike. When confronted by a complex program, a proofer will need special software to prove the program bug-free and he will need to introduce hint to the testing software in order to eliminate some situations that the program won't be able to handle[18]. Knowing this, there is thus no way to confirm with absolute certainty that a program will not present itself with a buggy behaviour down the round as some aspect of the testing were dismissed by a human in order to finalize the proofing process.

---

[17] Bernard Linsky, *The Evolution of Principia Mathematica* (Cambridge ; New York: Cambridge University Press, 2011), 122.
[18] Vasiga, *How to Test Programs*

The conclusion we can draw from all that mathematical and computer science background is that no program can prove that another program[19] won't generate infinite halt state. As the inherent quality of bug-free software is that there won't be any halting state in the program[20], it is then established that no program could declare another program bug free. Armed with that knowledge, the next section will now delve on the implication of Turing theory's and applications on cyberspace and how they impact the view onto that domain. This will also lead to an examination on the way cyberspace is perceived from a security standpoint.

Turing in cyber space

When Turing was developing his mathematical theory in 1936, he had no way of knowing how it could be extended to the computer science field of today. His aim was solely to prove a mathematical point but the application of it were shrouded by the ignorance of what was to come.

Now that we know that no algorithm can be proven right by another algorithm and that "a computer program, or just a program, is a sequence of instructions, written to perform a specified task on a computer"[21] and that those instructions derive from algorithms, it is logical that no computer program can be proven fault free. By extension, the cyber domain, also known as cyberspace to some, is the environment created by the interconnection of computer programs through a worldwide networking backbone,

---

[19] Ibid.
[20] Turing, *Solvable and Unsolvable Problems*, 1-3-4.
[21] Ralph M. Stair and George Walter Reynolds, *Principles of Information Systems*, 10th ed. (Boston, Mass.: Course Technology, Cengage Learning, 2012), 132.

commonly known as the Internet.[22] The number of program composing cyberspace is only limited by the programming will of the humankind or artificial intelligence machines that can create program[23]. This creates a cyberspace with an infinite number of programs, created or waiting to be created.

When correlating the two previous facts, first, that no computer program can be proven bug-free, and second, that cyberspace is composed by an infinite set of computer programs, the conclusion that there is an infinite subset of bugs in cyberspace can be drawn.

As we previously saw with Turing's machine theory, bugs are actually a state for which an algorithm is not concluding his intended work and thus entering a state of perpetual suspension[24], making it appears unresponsive. As the different algorithm composing a computer program cannot be proven fault-free, there is then no way of asserting that a program has no hidden bugs that could lead to a halting state.

It is in those possible bugs that remain the weakness of cyberspace. It is through them that opposition forces (OPFOR) leverage their capacity to discover and use programs to their advantages. It is through the exploitation of these bugs that the OPFOR gain unauthorized access to information or have computer programs behave in a malicious way, either confined to cyberspace or with an explicit impact on the physical

---

[22] Howard A. Schmidt, *Patrolling Cyberspace: Lessons Learned from a Lifetime in Data Security*, 1st ed. (N. Potomac, MD: Larstan Pub., 2006), 27.
[23] Alan D. Campen, Douglas H. Dearth and Armed Forces Communications and Electronics Association, *Cyberwar 2.0: Myths, Mysteries and Realities* (Fairfax, Va.: AFCEA International Press, 1998), 117.
[24] Turing, *Solvable and Unsolvable Problems*, 1-3-4.

domain[25]. One of the best examples of this is the STUXNET attack that will be described below.

The STUXNET exploit is now known to be an attempt by foreign agents, presumably the United States of America and Israel, to infiltrate electronically the computer programs controlling the Natanz uranium enrichment plant in Iran[26]. The aim was to create a computer program, commonly known as a virus, in order to infect the control module of the centrifuge of the plant and in the end, have the control module render the centrifuge unusable. The real challenge was to bring the virus to the centrifuge control modules as they were on a closed network with no ties to the external world. There was thus an air gap to bridge and that is where the whole complexity of the endeavour was[27].

In order to achieve their aim, the sponsor of the virus had a six steps plan in place. The first one was to target five different companies that had ties with the enrichment plant in order to have their computer infected with the virus. This was done through various means, notably the use of Universal Serial Bus (USB) memory stick and stolen electronic security certificate. The theft of the certificate is one of the strong hints that analysts like Kim Zetter take as a proof of nation-state involvement in the project[28]. The usage of the certificate was critical as the virus was able to appear legitimate to the targeted system, thus evading common anti-virus software.

---

[25] Chen, Jarvis and MacDonald, *Cyberterrorism: Understanding, Assessment, and Response*, 45.
[26] Kim Zetter, *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*, 1st ed. (New York: Crown Publishers, 2014a), 433.
[27] Jonh P. Geis II et al., "Deterrence in the Age of Surprise" Air War College), 24.
[28] Zetter, *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*, 227.

Once the initial systems were compromised, the next step was the search. The virus actively searched for the specific system it needed to compromises, an industrial control system made by the company Siemens. It was in that specific control that an exploitable bug was found by the creators of STUXNET and it was through it that the whole plan hinged. If the infected system had no matching software to the Siemens control, the virus turned itself off until such a system was introduced to him, in order to avoid detection[29]. If the specific system was found, the third step was for the virus to update itself with the latest version of itself, through the internet.

Then, the fourth step for the virus was to use as much as four zero-day vulnerabilities imbedded in it to infect the Siemens control systems. The fact that they were zero-day meant that these vulnerabilities were still unknown to the anti-virus companies and thus undetectable. This is another hint that indicates that this whole operation must have been state sponsored as it is extremely difficult to find zero-day vulnerabilities and keep them secret. So, to have four of them in only one product requires an amount of research only attainable through state-sponsored organisation[30].

Once it was installed in the Siemens control system, the fifth step for the virus was to take control of them through the different bug residing in the Siemens products. At first, the virus is not changing the normal behaviour of the centrifuge in order to obtain a

---

[29] Kim Zetter, "An Unprecedented Look at STUXNET, the World's First Digital Weapon," http://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/ (accessed 05/06, 2015).
[30] David Kushner, "The Real Story of Stuxnet," IEEE, http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet (accessed 05/01, 2015).

snapshot of normal operations. Once a thirty day evaluation is done, the virus enters its final stage, deceive and destroy[31].

In order to destroy the centrifuge, the virus is to have their rotors spin in unconventional fashion so has to create irreparable damage. All the while, the virus will replay normal behavior to the employees of the enrichment plan so that there is no suspicion as to what is really happening. Once the damage was done, the virus returned to step five, controlling the system while waiting for the right time to strike again[32].

When following the trail of STUXNET, it shows that all the unauthorized operations were made possible through bugs in the programming in place. The zero-days vulnerabilities were discovered bugs by the group in charge of STUXNET, the Siemens control tempered by the virus were exploited through buggy state controller and there is no way, as proved Turing with his theories that there is not any other such holes in the machine waiting to be found by astute computer analysts and programmers. This is a real challenge to all type of organizations but especially the military where security is one of the paramount tenants.

The current military approach toward cyber security is to make sure the environment is secured and that it cannot be exploited by anyone other than the friendly force[33]. This can be traced back to the military mandate of the Electronic Warfare (EW) units that is to "exploit the electromagnetic spectrum while denying its use to the

---

[31] Zetter, *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*, 287-298.
[32] Kushner, *The Real Story of Stuxnet*
[33] J. C. Walkling and Canadian Forces College, *Considerations: Canadian Forces' Efforts in the Electromagnetic Spectrum and Cyber Operating Environment*, Vol. JCSP/PCEMI 39-78 (Toronto, Ont.: Canadian Forces College, 2013), 89.

enemy[34]". Although this physical approach will succeed on the physical electromagnetic spectrum, simply transferring that school of thought to the cyber domain is less likely to be the best solution.

In his paper, McGuffin explains that the termed used in the military physical domain such as vital ground or center of gravity have no real meaning in the cyber realm[35]. It is then surprising that the bulk of the efforts of different government are on securing the cyberspace using either inappropriate or unrelated concept. It seems that the emphasis is put on ensuring security over something that will never be able to be fully secure because, as we now know, no program can be proven bug-free[36]. The next section will examine what are the tools and methods used by the OPFOR in that domain and compare them with our own.

The stalker in the night

While our aim is to exploit cyber space to our advantage and deny the same to the opposing forces, their own goal is exactly the same. The exploitation of the cyber domain is critical to many operations of the OPFOR in the contemporary environment. Through a legitimate use of cyber, the OPFOR usually work on the informational plane to try to gain the upper hand on the psychological and informational operation fronts[37]. Although, militaries use the same tools and counteract the effect of the OPFOR, pushing their own

---

[34] United States Army, *ELECTRONIC WARFARE IN OPERATIONS*, *FM 3-36*, ed. Department of the Army (Whashington D.C. USA: Department of the Army, 2012), 2-2.
[35] W. C. McGuffin and Canadian Forces College, *Soldiers of FORTRAN: Militarization of the 5th Dimension*, Vol. JCSP/PCEMI 39-48 (Toronto, Ont.: Canadian Forces College, 2013), 52.
[36] Canada. Public Safety Canada, *Canada's Cyber Security Strategy: For a Stronger and More Prosperous Canada* (Ottawa: Govt. of Canada, 2010), 12.
[37] Alan D. Campen and Douglas H. Dearth, *Cyberwar 3.0: Human Factors in Information Operations and Future Conflict* (Fairfax, VA: AFCEA International Press, 2000), 88.

messaging strategies, this tempted the OPFOR to make illegitimate use of the environment or more simply and aptly conduct cyber warfare.

There is many example of cyber warfare since the beginning of this century and not all of them are well documented due to either a lack of evidence in the public domain or simply no pointers towards the aim and the methods, leaving only the results apparent[38]. This section will explain two of the most documented cyber-attacks to illustrate from which vectors the OPFOR is gaining access and relate the results to the previous statement that these attacks leverage more often than not a fault in a program, a bug[39].

The first example we will examine is the attack that led to the United States Armed Forces Operation BUCKSHOT YANKEE (Op BY). What led to Op BY was an attack by a foreign intelligence service on one of the sensitive information system of the United States government, more specifically the Secret Internet Protocol Router Network (SIPRNet) that is used to transfer classified information by the State and the Defence Department. It is now believed that the attack was conducted by Russian state sponsored agency but this has not been proven in the public domain or confirmed by any USA government officials[40].

---

[38] Thomas M. Chen and Army War College . Strategic Studies Institute, *An Assessment of the Department of Defense Strategy for Operating in Cyberspace* (Carlisle, PA: Strategic Studies Institute, U.S. Army War College, 2013), 24.
[39] Anthony J. Masys, *Networks and Network Analysis for Defence and Security* (Cham: Springer, 2014), 176.
[40] Ellen Nakashima, "Cyber-Intruder Sparks Response, Debate," Washington Post, http://www.washingtonpost.com/national/national-security/cyber-intruder-sparks-response-debate/2011/12/06/gIQAxLuFgO_story.html (accessed 05/06, 2015).

The threat that triggered Op BY was a very basic one. A malicious program, later named agent.btz, on a thumb drive was inserted on a SIPRNet terminal. The dumb drive was allegedly laid around parking lots near USA middle-eastern bases of operation. The program being a worm, it replicated itself into the affected computer, ready to infect any other thumb drives, replicating itself into the network endlessly. This replication was made possible because the malicious piece of code exploited a hole in the operating system (Microsoft Windows 7) that would use the *autorun* function of the system to install itself on the unprotected system. As we can see here again, it is through the exploitation of a fault in programming that this whole operation began[41].

Agent.btz aim was not only to replicate itself on any computer he encountered but also to exploit its environment by copying and ultimately sending classified material to its unconfirmed outside master. USA government confirmed that because SIPRNet is a closed network with only controlled access to the Internet, agent.btz was unable to send back its stolen information to its controller[42]. The effect of the program was even more pernicious as the government decided to ban the use of any thumb drive for almost 18 months, denying their usage to field commanders who sorely needed them to do legitimate information exchange on the battlefield. It took an enormous effort to clean the network and the amount of hours worked on that operation was as much damageable as the intended effect of the program.

---

[41] William J. III Lynn, "Defending a New Domain: The Pentagon's Cyberstrategy," *Foreign Affairs*, no. September/October 2010 (2010).
[42] Nakashima, *Cyber-Intruder Sparks Response, Debate*

Another example of vector of attack of the OPFOR is simply breaches in communication protocol over the Internet, like the case of the Estonian Denial of Service attack (DDoS) that happened in 2007. As of now, it has not been publicly stated by any parties involved but the running doubts are that Russian-backed hackers were the instigator of the attack[43].

Again, the attacker leveraged flaws in programming by the overloading of requests to numerous servers of the Estonian government and business sector. Not knowing how to handle that type of request, the servers were unable to resolve any request for almost a day. The servers were eventually patched and the hole in the programming improved to handle that type of overload but the damage to the government and the Estonian in general was already done[44].

As these two examples demonstrate, the principal vector of attack in the cyber realm is inevitably a program fault that is exploited by an opposing force. These can be mundane in nature but once exploited open access to the heart of a system, rendering it useless or even worse, completely exploitable, and under the control of a superior intellectual force[45]. The next section will present some of the main defenses philosophies that militaries use to circumvent these attacks and evaluate if there should be a change in the approach.

---

[43] Chen and Army War College . Strategic Studies Institute, *An Assessment of the Department of Defense Strategy for Operating in Cyberspace*, 14.
[44] Matt Murphy, "War in the Fifth Domain," *The Economist*, no. July 2010 (2010).
[45] Lawrence V. Brown, *Cyberterrorism and Computer Attacks* (New York: Novinka Books, 2006), 53.

Shifting the paradigm, how can militaries deal with the Entscheidungsproblem

When confronted with the issue of protecting the cyber domain, most countries will take an approach akin to a ground military operation. They will try to put in places defenses, like trenches and obstacles, and will then dig in and make sure that no other forces go through their barricades. They will also prepare means to attack if any OPFOR comes to close from their position. This approach has always worked from the Romans to today and is the foundation of military defensive operations[46].

In the cyber space, the same is usually done. For example, the Canadian cyber security policy explains that the Government of Canada will harden the security of its infrastructure by having the necessary countermeasures in place to intercept incoming attacks. The government will also invest in research and development to come up with new protection solutions to shelter the critical infrastructure all the while helping private companies establish their own protection perimeter[47].

The USA strategy is relatively similar, although the focus is more oriented toward securing the country information as opposed to Canada's focus on securing to ensure economic prosperity[48]. In order to achieve their aim, the government agencies in charge of protecting the cyberspace will rely heavily on protective computer programs, installed on all critical informational infrastructures. That hardening is expected to provide the

---

[46] Campen, Dearth and Goodden, *Cyberwar: Security, Strategy, and Conflict in the Information Age*, 119.
[47] Canada. Public Safety Canada, *Canada's Cyber Security Strategy: For a Stronger and More Prosperous Canada*, 8.
[48] United States. Defense Science Board. Task Force on Resilient Military Systems and the Advanced Cyber Threat, *Task Force Report: Resilient Military Systems and the Advanced Cyber Threat*, 2.1.

resiliency required to withstand attacks and retaliate if it is deemed appropriate by their political masters[49].

Knowing now that there can be no fault free software, as proven by Turing through his mathematical proofs, and that any fault in software will be eventually used against that software to either remove it, render it useless or use it towards other design, it seems to be wishful thinking to hope that the government designed programs that should be there for a nation protection won't be circumvented in the same fashion.

A paradigm shift needs to happen if occidental countries hope to remain on top of the podium in this realm. By only creating a software and relying on it to operate as designed is naïve at best. As stated in USA Task Force on Cyber, the individual cyber culture must change in order to improve security as there is too much undeserved trust for programs and computer that should not receive it[50].

There needs to be a real change in mentality towards how cyber defense is seen. As opposed to installing a program and feeling secured once the install bar hits the completion point, our operators needs to continually challenge their perception and ensure constant verifications over the equipment they manage. It is easy to overlook strange behavior in the computer world just because it is a computer, although that strange behavior might be a malicious attack. In the USA, after Op BY, cyber became a matter of leadership, not a matter of system administrators[51]. This shift is required to happen at all

---

[49] Chen and Army War College . Strategic Studies Institute, *An Assessment of the Department of Defense Strategy for Operating in Cyberspace*, 16.
[50] United States. Defense Science Board. Task Force on Resilient Military Systems and the Advanced Cyber Threat, *Task Force Report: Resilient Military Systems and the Advanced Cyber Threat*, 9.0.
[51] Nakashima, *Cyber-Intruder Sparks Response, Debate*

level and in every type of employment. There is no difference between the clerk using his computer to file in a pay allotment and the soldier practicing shooting on a virtual system. Both use a computer to enable their activity and both needs to be aware of the potential of cyber-attack at their level.

It will only be through training and mentorship that this change can happen and there is no other way to begin than from the top of the hierarchy. System administration and security officer can give briefing all they want, it is through engagement with their peers and supervisors that change will come.

In order to be prepared to retaliate if the needs come, operators of system designated to do so must also be given as much room to explore the potential means to use them to do so. According to the Capability of the People's Republic of China to Conduct Cyber Warfare and Computer Network Exploitation report, the Chinese are fully exploiting their capability by enabling their operators[52]. To level that, we must ensure that we develop our own cyber operator creativity and enable them to use it without the constraint of a bureaucratic environment.

This is not to say that formal training and development must be set aside, actually this is very far from it. Creativity must be enabled in the development of solutions but not to the detriment of good order and discipline on the usage of the means. The OPFOR is nothing if not well organized it its chaos. For example, the Anonymous Group is able to have an unknown membership of over thousands of operator that all act independently in

---

[52] Bryan Krekel, *Capability of the People's Republic of China to Conduct Cyber Warfare and Computer Network Exploitation* (Washington, D.C.: US-China Economic and Security Review Commission,[2009]), 37.

learning how to devise new vectors of attack but can be mustered quickly to achieve results as a group effort[53]. This needs to be emulated and not necessarily loathed upon.

CONCLUSION

When we look back at the initial thesis of this paper, based on Alan Turing 1936 work on the Entscheidungsproblem, we can only accept the conclusion that there exist no way to completely secure cyberspace. There will always be the possibility that some bug will be found in computer program code, even if they are not known now. This comes from the fact that there is no way to formally prove that a computer program will not encounter a halting state which will result in an exploitable bug.

Throughout this paper, examples of exploitation of bugs were given to explain figuratively how this fact could be leveraged, either by our own forces or by the opposing force. It was also presented that some simple program could be proven bug free but that were no way to have that kind of proof without a human intervention in order to dismiss the machine unprovable steps. This led then to the question regarding the current military approach towards cyber space.

The world governments' strategy on cyber defense is mainly to lock the door and assume that no one will be able to pick the lock. This will inevitably lead to a false sense of security form the people using the systems. The way to view the cyber Entscheidungsproblem must change. It is possible that our systems will be attacked or compromised. The OPFOR is sufficiently savvy to exploit our own bugs against us like

---

[53] Gabriella Coleman, *Hacker, Hoaxer, Whistleblower, Spy: The Many Faces of Anonymous* (London: Verso, 2014), 132-145.

we saw with OP BUCKSHOT YANKEE. The culture on the issue must change; militaries especially must stop believing they have an untouchable system and that they can't be attacked. The paradigm must shift from certainty to constant vigilance, with a mixture of doubt. In a culture exempt of doubt, even the biggest indicators of security breaches will be dismissed.

There is also a need for creativity from our cyber operators. They must be given the right to experiment and try as much as possible as this is the way the OPFOR is training their own. If we constrain them, we will have inflexible defenders that won't be able to identify opportunity in the cyber domain.

As a military institution, we must maintain these doubts and ensure vigilance. There may be a time when humans will be able to create the perfect program; exempt of bugs and possible exploitation, but this is not today. Until Turing is proven wrong, we must make sure that the systems are setup in layer within layer of security, thus removing the false sense of security. There is no absolute certainty, but there will be absolute failure if that fact is set aside in the name of compliancy.

BIBLIOGRAPHY

Brown, Lawrence V. *Cyberterrorism and Computer Attacks*. New York: Novinka Books, 2006.

Campen, Alan D. and Douglas H. Dearth. *Cyberwar 3.0: Human Factors in Information Operations and Future Conflict*. Fairfax, VA: AFCEA International Press, 2000.

Campen, Alan D., Douglas H. Dearth, and Armed Forces Communications and Electronics Association. *Cyberwar 2.0: Myths, Mysteries and Realities*. Fairfax, Va.: AFCEA International Press, 1998.

Campen, Alan D., Douglas H. Dearth, and R. Thomas Goodden. *Cyberwar: Security, Strategy, and Conflict in the Information Age*. Fairfax, Va.: AFCEA International Press, 1996.

Canada. Public Safety Canada. *Canada's Cyber Security Strategy: For a Stronger and More Prosperous Canada*. Ottawa: Govt. of Canada, 2010.

Chen, Thomas M. and Army War College . Strategic Studies Institute. *An Assessment of the Department of Defense Strategy for Operating in Cyberspace*. Letort Papers. Carlisle, PA: Strategic Studies Institute, U.S. Army War College, 2013.

Chen, Thomas M., Lee Jarvis, and Stuart MacDonald. *Cyberterrorism: Understanding, Assessment, and Response*. New York: Springer, 2014.

Colarik, Andrew M. and Lech Janczewski. *Cyber Warfare and Cyber Terrorism*. Hershey, PA: Idea Group Reference, 2007.

Coleman, Gabriella. *Hacker, Hoaxer, Whistleblower, Spy: The Many Faces of Anonymous*. London: Verso, 2014.

Deibert, Ronald J. and Canadian Defence and Foreign Affairs Institute. *Distributed Security as Cyber Strategy: Outlining a Comprehensive Approach for Canada in Cyberspace*. Calgary, Alta.: Canadian Defence & Foreign Affairs Institute, 2012.

Gallaher, Michael P., Albert N. Link, and Brent Rowe. *Cyber Security: Economic Strategies and Public Policy Alternatives*. Cheltenham, UK ; Northampton, MA: Edward Elgar, 2008.

Geis II, Jonh P., Grant T. Hammond, Harry A. Foster, and Theodore C. Hailes. "Deterrence in the Age of Surprise."Air War College, 2014.

Grauman, Brigid and Security and Defence Agenda. *Cyber-Security: The Vexed Question of Global Rules : An Independent Report on Cyber-Preparedness Around the World*.

Security & Defence Agenda Report. Brussels, Belgium: Security & Defence Agenda, 2012.

Hodges, Andrew. *Alan Turing*. Centenary ed. Princeton, N.J.: Princeton University Press, 2012.

———. *Turing: A Natural Philosopher*. The Great Philosophers. Vol. III. London: Phoenix, 1997.

Krekel, Bryan. *Capability of the People's Republic of China to Conduct Cyber Warfare and Computer Network Exploitation*. Washington, D.C.: US-China Economic and Security Review Commission, 2009.

Kushner, David. "The Real Story of Stuxnet." IEEE. Accessed 05/01, 2015. http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet.

Leavitt, David. *The Man Who Knew Too Much: Alan Turing and the Invention of the Computer*. Great Discoveries. 1st ed. New York: W. W. Norton, 2006.

Linsky, Bernard. *The Evolution of Principia Mathematica*. Cambridge ; New York: Cambridge University Press, 2011.

Lynn, William J. III. "Defending a New Domain: The Pentagon's Cyberstrategy." *Foreign Affairs* no. September/October 2010 (2010).

Masys, Anthony J. *Networks and Network Analysis for Defence and Security*. Lecture Notes in Social Networks. Cham: Springer, 2014.

McGuffin, W. C. and Canadian Forces College. *Soldiers of FORTRAN: Militarization of the 5th Dimension*. Masters Thesis (Canadian Forces College). Vol. JCSP/PCEMI 39-48. Toronto, Ont.: Canadian Forces College, 2013.

Murphy, Matt. "War in the Fifth Domain." *The Economist* no. July 2010 (2010).

Nakashima, Ellen. "Cyber-Intruder Sparks Response, Debate." Washington Post. Accessed 05/06, 2015. http://www.washingtonpost.com/national/national-security/cyber-intruder-sparks-response-debate/2011/12/06/gIQAxLuFgO_story.html.

Negroponte, John D., Samuel J. Palmisano, Adam Segal, Council on Foreign Relations. Independent Task Force on Defending an Open, Global, Secure, and Resilient Internet, and Council on Foreign Relations. *Defending an Open, Global, Secure, and Resilient Internet*. Independent Task Force Report. Vol. 70. New York, NY: Council on Foreign Relations, 2013.

Petzold, Charles. "“The Imitation Game” and Alan Turing’s Real Contribution to Computing." . Accessed 04/21, 2015. http://www.charlespetzold.com/blog/2014/12/The-Imitation-Game-and-Alan-Turings-Real-Contribution-to-Computing.html.

Schmidt, Howard A. *Patrolling Cyberspace: Lessons Learned from a Lifetime in Data Security*. 1st ed. N. Potomac, MD: Larstan Pub., 2006.

Stair, Ralph M. and George Walter Reynolds. *Principles of Information Systems*. 10th ed. Boston, Mass.: Course Technology, Cengage Learning, 2012.

Turing, A. M. "On Computable Numbers, with an Application to the Entscheidungsproblem." Graduate, Princeton University, 1936.

———. "Solvable and Unsolvable Problems." *Science News* 31, (1954): 1-3-4.

United States Army. *Electronic Warfare in Operations, Fm 3-36*, edited by Department of the Army. Whashington D.C. USA: Department of the Army, 2012.

United States. Defense Science Board. Task Force on Resilient Military Systems and the Advanced Cyber Threat. *Task Force Report: Resilient Military Systems and the Advanced Cyber Threat*. Washington, DC: Office of the Under Secretary of Defense for Acquisition, Technology and Logistics, 2013.

Vasiga, Troy. "How to Test Programs." University of Waterloo. Accessed 04/21, 2015. https://cs.uwaterloo.ca/~tmjvasig/CS134Testing.html.

Walkling, J. C. and Canadian Forces College. *Considerations: Canadian Forces' Efforts in the Electromagnetic Spectrum and Cyber Operating Environment*. Masters Thesis (Canadian Forces College). Vol. JCSP/PCEMI 39-78. Toronto, Ont.: Canadian Forces College, 2013.

Zetter, Kim. *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*. 1st ed. New York: Crown Publishers, 2014.

———. "An Unprecedented Look at STUXNET, the World's First Digital Weapon." . Accessed 05/06, 2015. http://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/.