Malware is Called Malicious for a Reason:
The Risks of Weaponizing Code

CyCon 2014: Sixth Annual International Conference on Cyber Conflict
NATO Cooperative Cyber Defence Center of Excellence
Tallinn, Estonia, June 2014

Stephen Cobb
Research Department
ESET North America
San Diego, USA

Andrew Lee
Office of the CEO
ESET North America
San Diego, USA

Abstract: The allure of malware, with its tremendous potential to infiltrate and disrupt digital systems, is understandable. Criminally motivated malware is now directed at all levels and corners of the cyber domain, from servers to endpoints, laptops, smartphones, tablets, and industrial control systems. A thriving underground industry today produces ever-increasing quantities of malware for a wide variety of platforms, which bad actors seem able to deploy with relative impunity. The urge to fight back with "good" malware is understandable. In this paper we review and assess the arguments for and against the use of malicious code for either active defense or direct offense. Our practical experiences analyzing and defending against malicious code suggest that the effect of deployment is hard to predict with accuracy. There is tremendous scope for unintended consequences and loss of control over the code itself. Criminals do not feel restrained by these factors and appear undeterred by moral dilemmas like collateral damage, but we argue that persons or entities considering the use of malware for "justifiable offense" or active defense need to fully understand the issues around scope, targeting, control, blowback, and arming the adversary. Using existing open source literature and commentary on this topic we review the arguments for and against the use of "malicious" code for "righteous" purposes, introducing the term "righteous malware". We will cite select instances of prior malicious code deployment to reveal lessons learned for future missions. In the process, we will refer to a range of techniques employed by criminally-motivated malware authors to evade detection, amplify infection, leverage investment, and execute objectives that range from denial of service to information stealing, fraudulent, revenue generation, blackmail and surveillance. Examples of failure to retain control of criminally motivated malicious code development will also be examined for what they may tell us about code persistence and life cycles. In closing, we will present our considered opinions on the risks of weaponizing code.

Keywords: malware, weaponize, malicious code, active defense, cyber conflict

1. Introduction

On November 23 of 2013, news reports appeared stating that the National Security Agency of the United States (NSA) had installed malware on 50,000 computers around the world.[1] Three days later, Langner published a comprehensive analysis of Stuxnet.[2] Regardless of whether you agreed with all of Langner's conclusions, or regarded the reports of NSA malware deployment as fact or an erroneous allegation, these events served as a powerful reminder that the use of malicious code for nation state purposes is no longer a theoretical concern, but a present reality with serious socio-political and economic consequences. We will mention just some of these consequences as we argue that there is an urgent need for broader understanding of the merits and pitfalls of malicious code deployment, whether for cyber offense, active cyber defense, or cyber espionage, including legal and illegal surveillance for nation state or law enforcement purposes.

Numerous events over the last twenty years have demonstrated that malicious code has great potential as a means of infiltrating and disrupting digital systems of all kinds, for all manner of motives. Online markets now exist within which criminals and countries alike can acquire all of the means necessary for a malware campaign. With access to malware now easier than ever, the use of malicious code for either active defense or direct offense holds great fascination for nation states. Commercial suppliers are emerging to meet the demand, such as KEYW and Endgame.[3] Yet the literature of cyber conflict frequently notes that the deployment of malicious code by nation states is problematic.[4]

Unfortunately, detailed descriptions of the exact nature of the problems posed by weaponizing code are hard to find, a situation that we consider to be a problem in itself because it tends to create the impression that the objections to malware deployment are addressable. In turn, this could lead to the assumption that deployment of malicious code by nation states is inevitable. In the context of human conflict, to ascribe inevitability to an act that in reality requires a conscious decision is to court danger. Nation states can chose not to deploy malicious code and we will argue that more of them may make that choice if the problems inherent in malicious code deployment are better understood.

Clearly, more light must be shed on these issues at all levels, from the citizenry to the military, to the body politic. In this paper we elucidate the problems inherent in malicious code deployment by nation states and law enforcement agencies by first reviewing a list of reasons for thinking that a "good virus" is a bad idea. However, we distinguish the idea of a good virus designed to perform acts widely seen as beneficial, like backing up databases or patching systems, from code written to perform acts that benefit the deployer to the detriment of the target. We propose the term "righteous malware" for the latter. We also propose that any plans to deploy righteous malware be checked against the list of objections to good viruses, and then further evaluated relative to addition considerations that we present.

[1] "NSA infected 50,000 computer networks with malicious software," NRC, Nov. 23, 2013. Available: http://www.nrc.nl/nieuws/2013/11/23/nsa-infected-50000-computer-networks-with-malicious-software

[2] R Langner, "To Kill a Centrifuge: A technical analysis of what Stuxnet's creators tried to achieve," Nov. 2013. Available: http://www.langner.com/en/resources/papers

[3] M Riley and A Vance, "Cyber Weapons: The New Arms race," BusinessWeek, Jul. 20, 2011. Available: http://www.businessweek.com/magazine/cyber-weapons-the-new-arms-race-07212011.html

[4] Tallinn Manual, p.53

After considering the possible benefits of righteous malware we will conclude with an attempt to understand why some people still favor deploying malware in spite of longstanding objections from those who deal with malware on a daily basis.

2. DEFINING MALWARE AND MOTIVES

For a working definition of malicious code we thought it fitting to use the one provided by the National Security Agency of the United States (NSA) in its 2007 publication: *Guidance for Addressing Malicious Code Risk*.[5] We note that this document borrows from the Committee for National Security Systems (CNSS) Instruction 4009 *National Informational Assurance (IA) Glossary*,[6] signed in 2006 by Lieutenant General Michael Hayden. The entry for malicious code reads: "software or firmware intended to perform an unauthorized process that will have adverse impact on the confidentiality, integrity, or availability of an IS [Information System]."

The NSA document goes on to clarify that malicious code includes both unauthorized software that has an adverse effect, and authorized software that, when used improperly, has an adverse effect, noting: "This may include software in which exploitable faults have been intentionally included." Clearly, this view of malicious code encompasses logic bombs and backdoors coded into software and firmware during design and development, as well as the more commonly discussed phenomena such as viruses, worms, and Trojans. One could argue that it also includes causing industrial control software to increase the speed of an electric motor, such as you might find in a centrifuge.

The meat of the NSA's guidance on malware is found in the section headed "Malicious Code in the Software Life Cycle" which reviews threats, vulnerabilities, and mitigation strategies across the seven life cycle stages listed in Table I.

TABLE I THE SOFTWARE LIFE CYCLE IN SEVEN STAGES

1. Acquisition
2. Requirements
3. Design
4. Construction.
5. Testing
6. Installation (delivery, distribution, installation)
7. Maintenance (operation, maintenance, and disposal)

[5] *Guidance for Addressing Malicious Code Risk*, NSA, 2007.

[6] *National Informational Assurance (IA) Glossary*, CNSS National Security Systems Instruction 4009, 2006.

What is striking about this table is that the general public, and possibly too many information and communication technology (ICT) professionals, think of malicious code as being a stage seven problem. Despite this popular perception of malware as something inserted into systems after they are installed, for the purposes of this paper we will use malware to refer to all malicious code, not least because the NSA itself is alleged to have deployed backdoors in hardware, presumably at stage three or four.[7]

The idea of code that automatically inserts itself into a computer system at stage seven has been around almost as long as computers themselves. We refer to the concept of the "good virus," sometimes referred to as the "beneficial virus," self-replicating which does something positive, like encrypt files or patch code, in a fully automated and unsupervised manner.[8] However, both goodness and benefit are in the eye of the beholder, or in this case, in the opinion of the system owner on which the automated code is running. If you discern an unauthorized process on your network and find that its function is to email all of your engineering drawings to another country you are not likely to call it good or beneficial, in your opinion it is malicious.[9] Of course, the recipient of your drawings may find the arrangement beneficial and consider the code that delivers them to be good, even though it is, by all definitions, malware.

For this reason we introduce a new term to assist in the discussion of malware used for allegedly legitimate purposes: righteous malware. The following definition of righteous malware adds the aspect of motive to the purpose of the code: software or firmware deployed with intent to perform an unauthorized process that will impact the confidentiality, integrity, or availability of an information system to the advantage of a party to a conflict or supporter of a cause. We use the terms conflict and cause to distinguish righteous malware from malicious code that is motivated purely by financial gain. The party might be a person or group of persons, such as a nation state or agent thereof, or non-state actors, or even so-called hacktivists. What they have in common is the belief that their use of malware is justified, despite the fact that owners of systems and data impacted by the code are unlikely to agree.

While the concept of righteous malware is very different from that of good viruses, we assert that the persistent allure of the latter contributes to the persistence of the notion that malware can be deployed in a controlled manner to achieve, at least in the eyes of the deployer, beneficial results, such as hindering the process of enriching uranium that might be used to build nuclear weapons.

3. THE GOOD VIRUS PROBLEM

The allure of using self-replicating computer code to perform beneficial tasks dates back at least as far as the 1980s when it was explored by Dr. Fred Cohen.[10] Some early virus writing efforts were inspired

[7] .T Simonite, "NSA's Own Hardware Backdoors May Still be a "Problem from Hell", Oct. 8, 2013. Available: http://www.technologyreview.com/news/519661/nsas-own-hardware-backdoors-may-still-be-a-problem-from-hell/

[8] C. Peikari, "Fighting Fire with Fire: Designing a "Good" Computer Virus," Informit, Jun. 2011. Available: http://www.informit.com/articles/article.aspx?p=337309&seqNum=2

[9] R. Zwienenberg, "ACAD/Medre.A 10000's of AutoCAD files leaked in suspected industrial espionage," We Live Security, Jun. 21, 2012. Available: http://www.welivesecurity.com/2012/06/21/acadmedre-10000s-of-autocad-files-leaked-in-suspected-industrial-espionage

[10] F. Cohen, "Computational Aspects of Computer Viruses," *Computers & Security*, 8, 1989, pp. 325–344.

by this concept.[11] Unfortunately, the results ranged from annoying to expensive. However, the idea of beneficial viruses has proved surprisingly immune to discouragement, prompting antivirus researchers to make repeated public statements of the problem in an effort at dissuasion, most notably in 1994, when Vesselin Bontchev, then a research associate at the Virus Test Center of the University of Hamburg, published an article titled: *Are "Good" Computer Viruses Still a Bad Idea?*[12]

Despite the many changes in the technology landscape that have occurred in the two decades since that paper was published, it is still a useful starting point for understanding objections to the deployment of malware. We think that a review of problems with the release of self-replicating code that was created to do good makes a convenient starting point for assessing the virtue of employing any kind of code designed to execute without permission or through deception.

One reason to use Bontchev's list is that it summarizes extensive input from a group of antivirus experts. Bontchev asked participants in VirusL/comp.virus,[13] an electronic forum dedicated to discussions about computer viruses, to list all the reasons why they thought the idea of a "beneficial" virus was flawed. From their responses Bontchev produced “a systematized and generalized list of those reasons” of which there were twelve, grouped into three categories: technical, ethical and legal, and psychological. The reasons are presented in Table II.

TABLE II REASONS WHY GOOD VIRUSES ARE A BAD IDEA

Technical Reasons	
Lack of Control	Spread cannot be controlled, unpredictable results
Recognition Difficulty	Hard to allow good viruses while denying bad
Resource Wasting	Unintended consequences (typified by the “Morris Worm”)
Bug Containment	Difficulty of fixing bugs in code once released
Compatibility Problems	May not run when needed, or cause damage when run
Effectiveness	Risks of self-replicating code over conventional alternatives
Ethical and Legal Reasons	
Unauthorized Data Modification	Unauthorized system access or data changes illegal or immoral
Copyright and Ownership Problems	Could impair support or violate copyright of regular programs
Possible Misuse	Code could be used by persons will malicious intent
Responsibility	Sets a bad example for persons with inferior skills, morals

[11] For example, the 1982 Xerox Worm designed to enable distributed computation, see D. Harley, R. Slade, et al, *Viruses Revealed*, Osborne/McGraw-Hill, 2006, p. 56.

[12] V. Bontchev, “Are ‘Good’ Computer Viruses Still a Bad Idea?” Proc. EICAR’94 Conf., pp. 25-47.

[13] Virus-L and comp.virus were a mailing list and online forum respectively, now archived at Google group, located at https://groups.google.com/forum/#!forum/alt.comp.virus

Psychological Reasons	
Trust Problems	Potential to undermine user trust in systems
Negative Common Meaning	Anything called a virus is doomed to be deemed bad

We recommend that anyone considering the deployment of malicious code, either for offense or active defense, use this table as a basic checklist of concerns that need to be addressed (a more advanced checklist will be supplied later).

Consider a scenario in which a nation state is considering deployment of a virus designed to analyze cyber attacks against the deployer's systems, then identify the systems that are the source of the attack, and attempt to disable those systems in a counter attack.[14]

How does this plan measure up to the checklist? Frankly, we see problems in all twelve areas but will highlight just a few. Firstly, we doubt that such a program could be written in a way that would: rule out unanticipated actions that interfered with the attack code control mechanisms (Lack of Control); and prevent unanticipated and harmful reactions in all systems traversed during or after the counter attack (Compatibility Problems). We further doubt that this code could achieve its objective without detection, which would result in it being blocked by commercial antivirus programs (Recognition Difficulty Problem[15]).

While legal niceties (Unauthorized Data Modification) and excessive use of resources (Resource Wasting) may not bother the nation state behind the counter attack code, these are issues that may bother its citizens if the program comes to light. Spending taxpayer money to create code which is quickly co-opted by criminals to attack taxpayers (Possible Misuse) is also likely to be very unpopular. Of course, if the makers of the code solve all of these problems and achieve a successful deployment that defeats a serious attacker, the project may appease criticism in the area of Responsibility. However, a lack of success could undermine confidence in technology (Trust Problems) and lead to economic contraction.[16] Clearly the road to successful malware deployment is fraught with problems, as many failed malicious code campaigns attest.[17]

Of the above problems, the one that seems to have received the most attention in the literature of cyber conflict is control. However, even the most compelling examination of whether or not adequate levels of control over malware are achievable acknowledges that controls cannot prevent all problems: "Despite the care with which cyber weapon controls may be developed, there is always the possibility

[14] A scenario akin to the anti-viral virus referenced by Enn Tyugu, "Command and Control of Cyber Weapons," 4th International Conference on Cyber Conflict, NATO CCDCOE, 2012, p. 334.

[15] Despite headlines to the contrary, commercial antivirus products frequently detect, identify, and block previously unknown malware, including that deployed by government entities. See R. Lipovsky, "German Policeware: Use the Farce…er, Force…Luke," We Live Security, Oct. 10, 2011. Available: http://www.welivesecurity.com/2011/10/10/german-policeware-use-the-farce-er-force-luke/

[16] S. Cobb, "NSA and Wall Street: online activity shrinks, changes post-Snowden," We Live Security, Nov. 4, 2013. Available: http://www.welivesecurity.com/2013/11/04/nsa-wall-street-online-activity-shrinks-post-snowden/

[17] S. Cobb, "When malware goes bad: an historical sampler," We Live Security, Nov. 31, 2013. Available: http://welivesecurity.com/2013/11/30/when-malware-goes-bad-an-historical-sampler

of undesired effects such as affecting the wrong target. The ability to control malware is only as good as the intelligence informing its development".[18]

A large part of that intelligence involves knowing the environment in which your malware will seek to achieve its righteous ends. Yet this process may not be able to fully anticipate every eventuality. What if the target changes some of the software or hardware it is running just moments before or after the malware is deployed? Is the malware going to be smart enough to detect such changes and shut itself down? When you look at the experience of the commercial software industry, which conducts a massive amount of pre-launch product testing, you see that every product launch plan invariably includes support staff and engineers standing by to deal with the inevitable problems that simply could not be predicted.

We realize that proponents of righteous malware could counter this analysis by asserting the following: If anything goes wrong it will not be a problem because nobody will know it was us. This assertion reflects a common misunderstanding of the attribution problem, which is defined as the difficulty of accurately attributing actions in cyber space. While it can be extremely difficult to trace an instance of malware or a network penetration back to its origins with a high degree of certainty, that does not mean "nobody will know it was us." There are people who know who did it, most notably those who did it. If the world has learned one thing from the actions of Edward Snowden in 2013, it is that secrets about activities in cyber space are very hard to keep, particularly at scale, and especially if they pertain to actions not universally accepted as righteous.

Before moving on from the good virus checklist we should note that attribution was not listed as a problem for "the beneficial virus" back in 1994. After all, many virus writers proudly claimed their creations precisely because they thought they had created something beneficial (or at least functional with no intentional ill effects). Only when illegal activities rose to the fore as the primary motive for virus writing did malicious code attribution become an issue, initially for purposes of prosecution. Attribution becomes a critical issue when malicious code is used for cyber espionage or cyber attack, although it may not perceived to be a problem by those who deploy malware for motives they deem righteous. Responsibility for malware can be plausibly denied (with varying degrees of success, see Mandiant report[19]), or it can be tacitly acknowledged if you want to make a point (as may have been the case with Stuxnet[20]). And, indeed, there are good reasons why an agency involved in such attacks might wish to claim responsibility for an attack, though this is something of a double-edged sword. The fact remains that the perpetrators know who they are, and one day they may talk.

4. RIGHTEOUS MALWARE

Of course, a lot has changed in the two decades since Bontchev's paper laid out the reasons why a consensus of antivirus researchers think a virus designed with the best of intentions is a bad idea. As

[18] D. Raymond, G. Conti, et al, "A Control Measure Framework to Limit Collateral Damage and Propagation of Cyber Weapons," 5th International Conference on Cyber Conflict, 2013.

[19] Mandiant, "APT1: Exposing One of China's Cyber Espionage Units," Feb. 2013. Available: http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf

[20] R Langner, ibid, p.16

active participants in the antivirus community, we have not observed any change in that consensus over the years and we have heard the reasons against intentional malware deployment reiterated many times, yet we continue to see malware intentionally released into the wild with what its deployers believe to good intentions, such as waging "war on terror" and "war on drugs".[21]

One development we have observed over the last twenty years is an increase in the use of malicious code that is not self-replicating and therefore, one could argue, not subject to all of the problems ascribed to viruses and worms. We will concede that deploying righteous malware that is designed to work without self-reproductive abilities will address some of the problems we have listed, but this design choice also limits the capabilities of the malware. Furthermore, it does not mean that the malware will not be reproduced, either inadvertently (for example, when an infected system is cloned or archived), or intentionally (by someone who has discovered it and wants to re-use it).[22]

Another change in recent years has been the growth of criminal enterprises founded on the exploitation of all kinds of malicious code. There is now a well-established system of markets in which to buy and sell all of the components necessary to carry out a malware campaign, from system infection through to mule services for turning purloined data into cash.[23] Division of labor and specialization have enabled advances in efficiency and expertise not seen when a malware campaign has to be conducted end-to-end by a single campaigner (known in the last century as simply a virus writer).[24]

The rapid evolution of a market-based malware industry has turned the Possible Misuse problem identified in 1994 into an Inevitable Misuse problem today. It is not an exaggeration to say that the efforts by nation states to develop righteous malware fuel the criminal enterprise of malware production, delivery, and exploitation, to say nothing of making a market in zero day vulnerabilities.[25] Even when code itself is not re-used, techniques observed in weaponized malware may be quickly appear in criminal malware. For example, the creators of Stuxnet are widely considered to be pioneers in the use of stolen code-signing certificates to facilitate the spread of malware.[26] Today, the practice is mainstream and found in malware targeting the financial assets of consumers and corporations around the world.[27] Stuxnet also highlighted the benefits of modular malware design in which an existing infection could be enhanced with additional capabilities. Today, all the best banking malware sports a modular framework able to accept new tasking, leveraging the investment in infection to maximize

[21] Reuters, "U.S. directs agents to cover up programs used to investigate Americans," Aug. 5, 2013. Available: http://www.reuters.com/article/2013/08/05/us-dea-sod-idUSBRE97409R20130805

[22] R. Langner, ibid, p. 20.

[23] B. Krebs, "The value of a hacked email account," Krebs on Security, Jun. 2013. Available: http://krebsonsecurity.com/2013/06/the-value-of-a-hacked-email-account

[24] S. Cobb, "The Industrialization of Malware: One of 2012's darkest themes persists," We Live Security, Dec. 31, 2012. Available: http://www.welivesecurity.com/2012/12/31/the-industrialization-of-malware-one-of-2012s-darkest-themes-persists

[25] Tom Simonite, "Welcome to the Malware-Industrial Complex," MIT Technology Review, Feb. 13, 2013. Available: http://www.technologyreview.com/news/507971/welcome-to-the-malware-industrial-complex

[26] Tom Simonite, "Stuxnet Tricks Copied by Computer Criminals," MIT Technology Review, Sep. 12, 2012. Available: http://www.technologyreview.com/news/429173/stuxnet-tricks-copied-by-computer-criminals

[27] J. Boutin, "Code certificate laissez-faire leads to banking Trojans," We Live Security, Feb. 21, 2013. Available: http://www.welivesecurity.com/2013/02/21/code-certificate-laissez-faire-banking-trojans. Also R. Lipovsky, "Back to School Qbot, now Digitally Signed," We Live Security, Sep. 7, 2011. Available: http://www.welivesecurity.com/2011/09/07/back-to-school-qbot-now-digitally-signed

returns.[28] Having a hard time recruiting money mules to convert stolen banking credentials into cash? Push a distributed denial of service (DDoS) module to your network of compromised machines and rent them out.

There may be an even bigger re-use problem. We are not experts in military history, doctrine, or philosophy, so we are unaware of the correct word for the following category of weapons: the ones you deliver to your enemies in re-usable form. Examples we can think of are rocks, arrows, throwing spears, and non-returning boomerangs. These weapons are delivered intact, available for re-use by the recipients, assuming they, the recipients and the weapons, are not too badly damaged by the act of delivery. Whatever the correct term for this ancient category of weapon, we think it includes the most modern of weapons, righteous malware. In fact, it is perhaps true to say that righteous malware is unique in that you are giving away your weapons, tactics, and designs, simply by using them.[29]

Almost by definition, righteous malware is code that you deliver to the victim/target in working order, whether via email, browser exploit, USB key, firmware update, or embedded chipset. This raises the very real possibility that the recipient can discover the code, reverse engineer it, and use it against you. As Rustici has pointed out, the practical impossibility of knowing whether or not this has happened is just one of many ways in which cyber weapons differ from conventional weapons.[30] For example, satellite imagery cannot provide you with an early warning of a cyber attack. Your adversary cannot be seen marshaling cyber weapons on your borders, not least because there are no borders in cyberspace.

Ascertaining the cyber capabilities of potential adversaries is a non-trivial task further complicated by globally dispersed non-state actors and an international sub-culture of hackers for hire and malicious code delivery systems for purchase or rent. There is also a risk of tremendous inequality in targets. Take for instance, a terrorist group operating a malware network from an undeveloped or chaotic country with the intention of attacking infrastructure in a developed nation. The group may feel it has little to lose if it deploys righteous malware that provokes a cyber response. Is there enough digital infrastructure in their country for a retaliatory cyber-attack to have a punishing affect. Not only that, but when dealing with people who have little interest in preserving their own lives or the lives of others, cyber capabilities may not offer much deterrence.[31]

While there has been extensive discussion of cyber conflict relative to theories and codes of war, much of it directed at a goal we support, limiting the use of cyber weapons, we argue that righteous malware has already created fallout, at a level we can ill afford to ignore. Three months after the press started reporting on the Snowden papers, we asked a representative sample of American adults who use the Internet how the revelations had affected their sentiment, in general and with respect to specific aspects of Internet usage. About one in five agreed with this statement: "Based on what we have

[28] ESET, "Hesperbot: A New Advanced Banking Trojan in the Wild," Sep. 9, 2013. Available: http://www.welivesecurity.com/wp-content/uploads/2013/09/Hesperbot_Whitepaper.pdf

[29] A. Anghaie, "STUXNET: Tsunami of Stupid or Evil Genius?" Infosec Island, Jun. 1, 2012. Available: http://infosecisland.com/blogview/21507-Stuxnet-Tsunami-of-Stupid-or-Evil-Genius.html

[30] R. Rustici, "Cyberweapons: Leveling the International Playing Field," Parameters, Vol. XLI, No. 3, Autumn 2011. U.S. Army War College, p. 32.

[31] A. Lee, "Cyberwar: Reality, Or A Weapon of Mass Distraction?" Proceedings Of the 22nd Virus Bulletin International Conference, 2012, pp. 292 - 300.

learned about government surveillance I have done less banking online."[32] A similar percentage said they were less inclined to use email. We found that 14% had cut back on online shopping.

Whether this sentiment will lead to an ecommerce contraction remains to be seen. Our subjects said they were cutting back, not cutting off the Internet. We do not know if doubts will persist, but bear in mind that this sentiment was assessed before people heard about the following, all of which would tend to further exacerbate the problem: the NSA's mapping of Americans' social contacts, capturing of their address books and contact lists, hacking into connections between data centers owned by Yahoo and Google, and infecting 50,000 systems with righteous malware. All indications are that new and equally unsettling revelations will continue well into 2014.[33]

One more survey finding that should be cause for concern is that half of respondents said that they were now less likely to trust technology companies such as Internet service providers and software companies. One way to look at that number is as an erosion of public trust in the very entities to which people normally turn for help in securing their systems and protecting their digital domains. Ironically, the source of mistrust is the other place that people turn for protection: the government.

Trust in the very software that is designed to defeat malicious code has also been shaken. In October of 2013, a coalition of digital rights organizations and academics published an 'open letter' asking for clarification on vendor policies regarding cooperation with government agencies and/or law enforcement using state-sponsored Trojan code.[34] Historically, there is no evidence that any antivirus company had ever collaborated with any nation state or law enforcement agency to further the spread of righteous malware. The letter demonstrates the corrosive effect that revelations of government malware deployment can have on both trust and common sense. Several antivirus companies responded by pointing out they had already refuse to give passes to righteous malware.[35] Others pointed to their exposure of righteous malware in the past, and the improbability than any such software could be "allowed" by the antivirus industry.[36]

One term that keeps occurring to us as we look at the effect of righteous malware deployment on our industry and on the wider economy, is attrition. We fear that nations are at a tipping point, the downside of which is a slow but steady erosion of that essential building block of prosperous societies: trust. Malware of any kind eats away at trust in networked systems, the very systems that form the critical infrastructure and industrial fabric of developed countries. We are already seeing righteous malware deployment eroding trust in the institutions that deliver and defend that infrastructure.

While each new development of malicious code is met with new security measures, and the network continues to function for most people most of the time, each new round of attack and counter-

[32] S. Cobb, "Survey says 77% of Americans reject NSA mass electronic surveillance, of Americans," We Live Security, Oct. 29, 2013. Available: http://www.welivesecurity.com/2013/10/29/survey-says-77-of-americans-reject-nsa-mass-electronic-surveillance-of-americans

[33] S. Cobb, personal notes on Gen. R. Hayden's comments to The Ecommerce Summit, San Diego, Nov. 23, 2013.

[34] J Leyden, "Antivirus bods grilled: Do YOU turn a blind eye to government spyware, The Register," Nov. 5, 2013. Available: http://www.theregister.co.uk/2013/11/05/av_response_state_snooping_challenge

[35] M. Hypponen, "F-Secure Corporation's Answer to Bits of Freedom," News rom the lab, Nov. 6, 2013. Available: http://www.f-secure.com/weblog/archives/00002636.html

[36] R. Marko, A. Lee, et al, "ESET response to Bits of Freedom open letter on detection of government malware," We Live Security, Nov. 11. Available: http://www.welivesecurity.com/2013/11/11/eset-response-to-bits-of-freedom-open-letter-on-detection-of-government-malware/

measure further encumbers the technology and reduces its potential to deliver the continued productivity gains upon which much future economic growth is predicated. Not only that, but savvy operators will find other channels to avoid detection, while millions of the innocent will have their privacy and security compromised.

5. THE BENEFITS OF RIGHTEOUS MALWARE

Whether used for offense or active defense, malicious code can boast numerous advantages, in its own right or relative to conventional weapons. Malicious code is an essential component of cyber weaponry, which is envisioned by Rustici as leveling the international playing field.[37] We examine these benefits and counter some of the arguments against deployment of righteous malware listed in the preceding section.

A. *Less deadly than kinetic weaponry*

The argument has been made that using code instead of kinetic weapons is more humane.[38] Cyber-attacks, if used carefully, certainly seem as if they could provide tactical advantage in ways that are not physically harmful and that do not require troop deployments.[39] If one nation state is convinced it has to take action against another, surely it is better to threaten, or execute, an attack on the networked systems of its opponent, where the effects may range from inconvenient to life-threatening, but stop short of deadly force. The demoralizing effect of sustained inconvenience, like intermittent malware-induced power outages, should not be under-estimated. However, as Rustici has pointed out, the benefits of weaponized code in this context do not accrue equally to all nations.[40] In fact, they stack the cards against developed nations whose greater reliance on cyber everything leaves them most vulnerable to such attacks, and in favor of less cyber reliant nations that nevertheless have rich traditions of learning and innovation.

B. *Works well for espionage*

Undoubtedly, malware can greatly facilitate espionage. Electronic espionage can definitely strengthen a nation's hand against its enemies and appears to be less encumbered by international treaties and norms governing nation state behavior. However, espionage is not without political and economic risks for the countries that engage in it, as the world discovered in 2013. We do not know if revelations of large-scale electronic spying, including widespread use of righteous malware, will have long term negative effects on nations, or the commercial entities perceived to be enablers of this activity. We remain alert to signs of economic contraction, retaliatory network Balkanization, or other potential ill effects. The apparent impact of being seen as an enabler on the price of shares in Cisco, shown in Figure 1, is a useful visual reminder.[41]

[37] R. Rustici, ibid, p. 32.

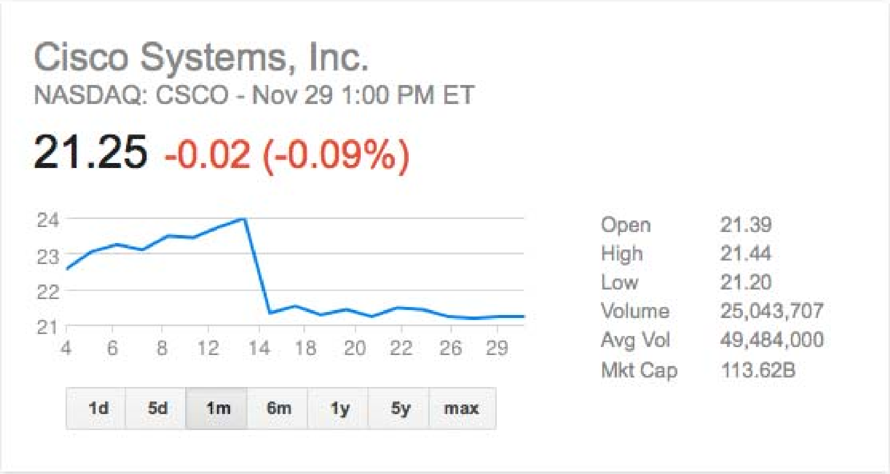
[38] D. Denning, "Obstacles and Options for Cyber Arms Controls," Paper presented at Arms Control in Cyberspace Conference, Berlin, Jun. 2001: "instead of dropping bombs on an enemy's military communication systems, for example, cyber forces could take down the system with a computer network attack, causing no permanent damage and no risk of death or injury to soldiers or civilians. The operation would be more humane and should be preferred over more destructive alternatives."

[39] J. Andress and S. Winterfeld, *Cyber Warfare: Techniques, Tactics and Tools for Security Practitioners*, Syngress, 2011.

[40] R. Rustici, ibid.

[41] D. Meyer, "Cisco's gloomy revenue forecast shows NSA effect starting to hit home," Gigaom, Nov. 14, 2013. Available: http://gigaom.com/2013/11/14/ciscos-gloomy-revenue-forecast-shows-nsa-effect-starting-to-hit

Figure 1 The November 14 “NSA effect” on Cisco stock



Cisco Systems, Inc.

NASDAQ: CSCO - Nov 29 1:00 PM ET

21.25 -0.02 (-0.09%)

Open	21.39
High	21.44
Low	21.20
Volume	25,043,707
Avg Vol	49,484,000
Mkt Cap	113.62B

C. *Less expensive that physical options*

Nation states have surely asked this question: Why spend billions to arm our country with sophisticated kinetic weapons and the trained soldiery needed to deploy them, when we can obtain malware-based cyber weapons for mere millions? Unfortunately, the allure of lower prices, particularly in terms of human cost, evaporates when cyber weapons are examined from a technical perspective. Many fall short of traditional definitions of weaponry and into various categories of strategic support for kinetic warfare, such as disabling or disrupting key infrastructure as an adjunct or precursor to kinetic attack.[42]

One can argue that the issue of cyber war is more properly considered an issue of security: systems security, network security, and due diligence on part of its operators. The majority of security breaches today—be they commercial, consumer, or military—are as a result of systems failure and human error, and the legal responses considered should perhaps be limited to such.[43] This problem lends itself to a situation of diminishing return, escalating cost and a strengthened enemy. It may be possible to gain a brief advantage initially, but this is soon lost if the enemy increases his own security posture in response.

Consider the case of Estonia, which came under digital attack in 2008. The damage was certainly quantifiable, but the end result was, that, paradoxically, the confident, even defiant, response by the Estonian government, and the prompt support lent by the North Atlantic Treaty Organization and the European Union, may have left Estonia in a stronger technical, political, and moral position after the attacks than before.[44] Therefore, expense cannot only be measured in development and deployment cost, but also in reputational and operational cost. There is also an issue of attribution. It stands to reason that one would want an enemy to recognize that an attack has been carried out, certainly if the purpose is deflection of further kinetic activity due to a show of strength. Strategically, this would

[42] A. Lee, ibid.

[43] T. Guo. "Shaping Preventive Policy in “Cyber War” and Cyber Security: A Pragmatic Approach" *J. Cyber Sec. Info. Sys.* 1-1.14 (2012). Available: http://works.bepress.com/tony_guo/2.

[44] T. C. Wingfield, “International Law and Information Operations,” in *Cyberpower and National Security*, F. D. Kramer, H. S. Starr, & K. L. Wentz (Eds., pp. 525-542). Washington DC: Potomac Books, 2009.

require exposure (as perhaps is the case with the US and Israel claiming responsibility for the Stuxnet malware). Such exposure though, raises the stakes, creating an arms race.

Eventually, we may reach equilibrium, where we understand that use of our own cyber-weaponry will result in an equally destructive response from our enemy. The 'nightmare' scenario is one where our 'enemy' has less to lose in terms of connected infrastructure, a strong defensive posture, and an advanced cyber weapons deployment capability. This will certainly be a costly situation for an attacker.

6. CONCLUSIONS

We see many problems with, and arguments against, the deployment of malicious code by anyone for any purpose. We have shown that many of these objections have been raised before. We have discussed additional risks, some of which have recently been demonstrated in world events. We also note additional objections and obstacles from those seeking to understand the relationship between cyber weapons and concepts like the laws of armed conflict (LOAC),[45] *jus in bello*,[46] and *jus ad bello*.[47] Review of these is beyond the scope of this paper but we have included them in our summary table of questions to ask before proceeding with the deployment of righteous malware, Table III.

TABLE III CONSOLIDATED LIST OF RIGHTEOUS MALWARE QUESTIONS TO ASK

Control	Can you control the actions of the code in all environments it may infect?
Detection	Can you guarantee that the code will complete its mission before detection?
Attribution	Can you guarantee that the code is deniable or claimable, as needed?
Legality	Will the code be illegal in any jurisdictions in which it is deployed?
Morality	Will deployment of the code violate treaties, codes, and other international norms?
Misuse	Can you guarantee that none of the code, or its techniques, strategies, design principles will be copied by adversaries, competing interests, or criminals
Attrition	Can you guarantee that deployment of the code, including knowledge of the deployment, will have no harmful effects on the trust that your citizens place in its government and institutions including trade and commerce.

[45] J. Healey, "When 'Not My Problem' Isn't Enough: Political Neutrality and National Responsibility in Cyber Conflict," 4th International Conference on Cyber Conflict, NATO CCDCOE, 2012.

[46] R. Fanelli and G. Conti, "A Methodology for Cyber Operations Targeting and Control of Collateral Damage in the Context of Lawful Armed Conflict," 4th International Conference on Cyber Conflict, NATO CCDCOE, 2012. p. 327.

[47] Reese Nguyen, "Navigating Jus Ad Bellum in the Age of Cyber Warfare," 101 Cal. L. Rev. 1079 (2013). Available at: http://scholarship.law.berkeley.edu/californialawreview/vol101/iss4/4

Note that we are talking about objections to the deployment of righteous code, not its development. Detailed discussion of this important distinction is beyond the scope of this paper.

Frankly, we do not anticipate the imminent outbreak of outright cyber war, but we do anticipate that righteous malware will continue to be a serious problem. As one of the authors has previously stated: "Cyber-attack capabilities, then, seem most likely to be useful in the future precisely in the same ways as they are being used now: causing temporary and generally non-injurious disruption to systems, whether to embarrass, shame or disrupt organizations, or to steal useful information, and perhaps prevent or delay technological progress."[48] To this we would now add the risks of economic contraction and trust erosion that come from secret cyber operations, including the use of righteous malware, being made public.

Finally, we need to ask why nation states and law enforcement agencies persist in the deployment of righteous malware. Do those who are in a position to approve such deployments still think the potential benefits outweigh the risks? Naturally, we would argue that the risks have not been fully appreciated, a recurring problem in information assurance if risk assessment methodologies developed in simpler times are applied to rapidly evolving technology. When assessing the location for a proposed data center you can use historical tables to put a number on the likelihood of floods, high winds, and other threat events. But what about assessing risks to systems on which novel attacks are possible? Just because a country has never experienced a particular type of attack, such as malicious code damaging a critical infrastructure, does not mean the probably of this happening in the future is zero. Indeed, it is entirely possible and efforts are underway in many countries to defend against such eventualities.

The best place to find an explanation of why a government that openly acknowledges its vulnerability to cyber attack would simultaneously engage in cyber attack, as the US arguably has done, may be the Gerras critical thinking model,[49] which has already been applied to an exploration of the prudent limits of automated cyber attack.[50] The model is apt because it derives from a military setting and the two entities most heavily involved in decisions to deploy righteous malware in the US are, at the time of writing, under military command. We fear that one or more of the nine common logical fallacies enumerated by Gerras could lead to a damaging weaponized code deployment of which the right questions were not asked or critically answered.

Acknowledgments

The authors acknowledge the assistance of fellow ESET researchers including Lysa Myers, David Harley, Aryeh Goretsky, Cameron Camp, and Righard Zwienenberg.

[48] A. Lee ibid

[49] S. Gerras, "Thinking Critically About Critical Thinking: A fundamental guide for strategic leaders," Carlisle Barracks: U.S. Army War College, Department of Command, Leadership, and Management, August 2008. Available: http://www.au.af.mil/au/awc/awcgate/army-usawc/crit_thkg_gerras.pdf

[50] J. Caton, "Exploring the Prudent Limits of Automated Cyber Attack." 5th International Conference on Cyber Conflict, NATO CCDCOE, 2013.