

INFORMATION WARFARE ON AN EVOLVING BATTLEFIELD

A Thesis

Presented to the

Faculty of

San Diego State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Homeland Security

by

Daniel Louis Gold

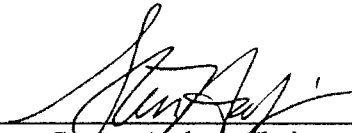
Fall 2012

SAN DIEGO STATE UNIVERSITY

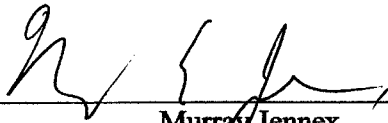
The Undersigned Faculty Committee Approves the

Thesis of Daniel Louis Gold:

Information Warfare on an Evolving Battlefield



Steven Andres, Chair
Homeland Security Program



Murray Jennex
Department of Management Information Systems



Allen Greb
International Security and Conflict Resolution Program

8-6-12

Approval Date

Copyright © 2012
by
Daniel Louis Gold
All Rights Reserved

DEDICATION

This work is dedicated to my parents, Sue and Marvin Gold, my sisters, Michelle and Diane, and my brother Peter Gold.

All warfare is based on deception.

-Sun Tzu

ABSTRACT OF THE THESIS

Information Warfare on an Evolving Battlefield

by

Daniel Louis Gold

Master of Science in Homeland Security

San Diego State University, 2012

Since the dawn of civilization warfare has been intrinsic to man's existence. War has shaped cultures, given rise to empires, and decimated entire populations. The nature of warfare is a constant evolution. Since the end of the Second World War, the medieval construct of opposing nation-state actors facing each other in open battle has effectively become obsolete. The balance of power established during the cold war characterized warfare's evolution as increasingly asymmetrical. The looming threat of mutually assured destruction gave rise to proxy-wars and covert actions. The terrorist attacks of September 11th 2001 were a testament to this progression. As such, warfare in the 21st century can be described as a multi-dimensional amalgam including the elements of global economics, the mass media, and terrorism. Intelligence, both human and signals based, has always played a central role in warfare. Due to the exponential growth of technology in recent decades, the tools of war have developed rapidly. Nearly all facets of modern warfare in the information age are now supported by the use of computers and the Internet. Whether motivated by politics, ideology, or monetary gain, criminals, spies, and terrorists all utilize computer based attacks. This work presents an analysis of emerging computer based threats in an effort to address the ongoing problem of securing our national digital infrastructure. Specifically, the case study presented will focus on the Stuxnet worm, which was arguably the first cyber-weapon to inflict significant kinetic damage. This study will examine Stuxnet, the Iranian nuclear program, and the potential that exists for similar threats to industrial control systems within the United States.

TABLE OF CONTENTS

	PAGE
ABSTRACT.....	vi
LIST OF TABLES.....	viii
LIST OF FIGURES	ix
ACKNOWLEDGEMENTS.....	x
CHAPTER	
1 INTRODUCTION	1
Information Warfare and Cyber Security	1
Statement of the Problem.....	22
Purpose of the Study	24
Limitations of the Study.....	24
Definition of Terms.....	24
Methodology.....	26
2 IRAN'S NUCLEAR ASPIRATIONS AND THE WAR ON TERROR.....	28
3 INDUSTRIAL CONTROL SYSTEMS.....	34
4 THE STUXNET WORM.....	43
5 CONCLUSIONS AND RECOMMENDATIONS	63
REFERENCES	75
APPENDIX	
A STUXNET TIMELINE	79
B PARTIAL ATTACK GRAPH.....	81
C CLUSTERS OF INFECTION BASED ON INITIAL INFECTION.....	83
D STUXNET VARIANTS	85
E DUQU INSTALLATION PROCESS.....	87

LIST OF TABLES

	PAGE
Table 1. List of Exports	48
Table 2. List of Resources	49
Table 3. Process Injection	50
Table 4. Comparison of Stuxnet and Duqu.....	65

LIST OF FIGURES

	PAGE
Figure 1. Global infection by W32.Stuxnet	23
Figure 2. Typical SCADA system	35
Figure 3. First generation SCADA architecture	36
Figure 4. Second generation SCADA architecture	37
Figure 5. Third generation SCADA architecture	38
Figure 6. Components of the SIMATIC PCS 7	39
Figure 7. Siemens' protocol for high security sites	40
Figure 8. Stuxnet P2P communication.....	46
Figure 9. Control flow of export 15	51
Figure 10. Infection routine flow	52
Figure 11. Command and control	55
Figure 12. Compromising the site's network	58
Figure 13. Geographic distribution of W32.Duqu	64
Figure 14. Double firewalled DMZ	67

ACKNOWLEDGEMENTS

First, I would like to thank my parents, Marvin and Sue Gold, for their patience and support in furtherance of my academic goals. I would also like to thank the department heads of San Diego State University's Interdisciplinary Homeland Security program, Dr. Jeffrey McIllwain and Dr. Eric Frost, for the quality of instruction they provided during my program of study. I would like to thank the members of my thesis committee for their involvement in this work: Steve Andres, for introducing me to the emerging field of cyber-security and advising me in the completion of this work, Murray Jennex for his instruction in the field of Management Information Systems, and Allen Greb, who apart from his involvement in my thesis committee, also served as my advisor during my undergraduate studies in San Diego State University's International Security and Conflict Resolution program.

CHAPTER 1

INTRODUCTION

In recent decades, cyberspace has grown to impact nearly every aspect of human existence. It is increasingly relied upon by citizens and policy-makers, as well as the military and federal agencies. Despite these facts, the importance of securing cyberspace is often overlooked. It is widely accepted that hackers, criminals, and foreign governments utilize the Internet for illicit purposes, but few understand the true nature of the threats facing the nation. Some describe the threat of cyber-attacks as exaggerated, while others warn of a digital doomsday. The truth is likely somewhere in between. Regardless, many find the subject matter confusing, due to its technical nature. The introductory portion of this paper will convey the essential aspects in the discourse of cyber-security, and provide a foundation for understanding the issues inherent in the field of information warfare. After this foundation is established, a review of the political theory surrounding the Iranian nuclear program will be conducted. A primer on industrial control systems will be presented along with a case-study of the Stuxnet computer worm and its effects on these critical systems. Finally, conclusions will be drawn as to the implications for blowback involved in Stuxnet's release, especially in regard to securing domestic industrial control systems.

INFORMATION WARFARE AND CYBER SECURITY

The birth of information theory is an appropriate starting point in the history of information warfare. Information theory was pioneered by Nobel Prize winning mathematician Claude Shannon in the 1930's and 40's. During Shannon's studies at MIT, "he did important work showing how logic could be applied to the design of relay circuits--in short, that the true-and-false of Boolean logic could be the same as the on-and-off of an electric switch."¹ After receiving his PhD, Shannon went on to work at Bell Labs, where his

¹ Charles A. Gimon, "Heroes of Cyberspace: Claude Shannon," InfoNation, <http://www.skypoint.com/members/gimonca/shannon.html> (accessed May 2, 2012).

research effectively laid the groundwork for the practical application of binary code in electronic communications.

In 1948, Shannon published “The Mathematical Theory of Communication” in the Bell System Technical Journal, along with Warren Weaver. This surprisingly readable document is the basis for what we now call information theory--a field that has made all modern electronic communications possible.²

Shannon’s theory of communication reduced all information quantitatively into units called “bits.” The term “bit” is an abbreviation of “binary digit,” which is represented as either a 1 or a 0. In terms of computing, eight bits make up a “byte,” which is the number of bits needed to encode a single letter. Shannon defined the capacities of communication channels by their ability to accurately relay a maximum number of bits free from error. During his time at MIT, Shannon’s “work on encryption led to the system used by Roosevelt and Churchill for transoceanic conferences, and inspired his pioneering work on the mathematical theory of cryptography,”³ which culminated in his 1949 publication “Communication Theory of Secrecy Systems.”

As time progressed, Shannon’s theories were applied to telephone systems and eventually enabled the development of the Arpanet in 1969. This principal connection of two computer systems, one at UCLA, and the other at Stanford Research Institute, was the predecessor to the modern Internet. The Arpanet grew as nodes popped up at other Universities across the nation. In 20 years, there were 100,000 systems linked together via the Arpanet.⁴

Shortly after the Arpanet’s creation, the first hackers began to emerge. In 1971 Vietnam War veteran John Draper, aka “Captain Crunch,” discovered that a toy whistle, the prize from a Cap’n Crunch cereal box, perfectly matched the 2600 Hz tone that telephone operators used to switch long-distance calls. This discovery prompted the creation of “blue boxes,” which were electronic units that matched the same 2600 Hz tone. These illicit

² Ibid.

³ Robert Calderbank and Neil J. A. Sloane, “Claude Shannon 1916-2001,” *Nature* 410 (2001): 768, <http://www2.research.att.com/~njas/doc/ces5.html> (accessed May 2, 2012).

⁴ Steven Andres, “Cyber Timeline,” <http://homelandsecurity.sdsu.edu/690/timeline> (accessed June 2, 2012).

devices were popularized by Apple Computers' founders Steve Wozniak and Steve Jobs. This practice of stealing free long-distance was referred to as phone "phreaking."

In 1974 Vinton Cerf and Robert Kahn published their paper "A Protocol for Packet Network Intercommunication," in which they described their development of the Transmission Control Protocol (TCP).

A packet communication network includes a transportation mechanism for delivering data between computers or between computers and terminals. To make the data meaningful, computer and terminals share a common protocol. Several protocols have already been developed for this purpose. However, these protocols have addressed only the problem of communication on the same network. In this paper we present a protocol design and philosophy that supports the sharing of resources that exist in different packet switching networks.⁵

Around the same time, Robert Metcalfe and his team at Xerox developed the Ethernet system of local area networking (LAN), which remains the dominant platform for computer networking today. These innovations in networking laid the foundation for the modern Internet.

In the 1980's, the first personal computers designed by IBM and Apple began to gain popularity. This was a key turning point in the history of computer hacking, as computers quickly became ubiquitous in businesses, and eventually homes around the world. One of the earliest hacking incidents occurred in 1983 when a group of teenagers, calling themselves "the 414's," were able to "break into several government computers, including a non-classified computer at the Los Alamos National Laboratory in New Mexico."⁶

The Morris worm, one of the first instances of a computer worm attack, took place in 1988. Robert Morris, a computer science student at Cornell University, claimed he did not write the program for malevolent purposes, but rather to measure the Internet. Regardless, after the worm caused significant damage to civilian and government systems Morris was convicted under the Computer Fraud and Abuse act of 1986.

On November 2, 1988, Robert Morris, Jr., a graduate student in Computer Science at Cornell, wrote an experimental, self-replicating, self-propagating program called

⁵ Vinton G Cerf and Robert E. Kahn , "A Protocol for Packet Network Intercommunication," *IEEE Transaction on Communications* 22 (1974): 1, <http://ece.ut.ac.ir/Classpages/F84/PrincipleofNetworkDesign/Papers/CK74.pdf> (accessed May 3, 2012).

⁶ Timeline: The U.S. Government and Cybersecurity, *Washington Post*, May 16, 2003. <http://www.washingtonpost.com/wp-dyn/articles/A50606-2002Jun26.html> (accessed May 5, 2012).

a worm and injected it into the Internet. He chose to release it from MIT, to disguise the fact that the worm came from Cornell. Morris soon discovered that the program was replicating and re-infecting machines at a much faster rate than he had anticipated---there was a bug. Ultimately, many machines at locations around the country either crashed or became catatonic. When Morris realized what was happening, he contacted a friend at Harvard to discuss a solution. Eventually, they sent an anonymous message from Harvard over the network, instructing programmers how to kill the worm and prevent re-infection. However, because the network route was clogged, this message did not get through until it was too late. Computers were affected at many sites, including universities, military sites, and medical research facilities. The estimated cost of dealing with the worm at each installation ranged from \$200 to more than \$53,000.⁷

The same year that the Morris worm was released, The US government established its Computer Emergency Response Team (CERT) which serves as “a central reporting center for Internet security problems.”⁸ US-CERT is still a functional entity to this day, and has inspired the creation of similar centers in nations around the globe.

In 1989, *The Cuckoo's Egg*, by Cliff Stoll was published. This is an essential piece of literature for anyone interested in the field of computer security. The book follows Stoll's experiences tracking down a hacker while working as a system manager at Lawrence labs in Berkeley California.

He discovered a 75-cent accounting error in the files which recorded the use of the computer's resources. Someone had not only used 75 cents worth of computer time, but had erased this 75 cents from one of the accounting files. What eventually became important was not the 75 cents, but the fact that someone (with user-name “Hunter”) was trying to cover his tracks. Cliff deleted Hunter, and the hacker logged on under another account, again causing an accounting error of a few cents. The hacker was scary for two reasons. First, he must have made himself super-user, in order to create the Hunter account, and erase accounting data. And second, he was using the Berkeley computer to hack into computers on the Arpanet and Milnet, research and military computer networks. Eventually, the FBI gets interested, and they track the hacker to Hannover, West Germany. And they find that he is selling military info to the KGB.⁹

Five years later, in 1994, Winn Schwartau's landmark work, *Information Warfare*, was published. In this book Schwartau defines information warfare (IW) as:

Actions taken to achieve information superiority by affecting adversary information, information-based processes, information systems, and computer-based networks while

⁷ Brendan P Kehoe, *Zen and the Art of the Internet* (New Jersey: Prentice Hall, 1992), <http://groups.csail.mit.edu/mac/classes/6.805/articles/morris-worm.html> (accessed May 5, 2012).

⁸ Timeline: The U.S. Government and Cybersecurity, *Washington Post*, May 16, 2003. <http://www.washingtonpost.com/wp-dyn/articles/A50606-2002Jun26.html> (accessed May 5, 2012).

⁹ Jim Loy, “The Cuckoo's Egg - by Clifford Stoll,” review of *The Cuckoo's Egg* by Clifford Stoll, Jim Loy Books, 1997, <http://www.jimloy.com/books/cuckoo.htm> (accessed May 5, 2012).

defending one's own information, information-based processes, information systems, and computer-based networks.¹⁰

Schwartau further divides IW into three classes. The first class is personal, which largely refers to the violation of electronic privacy and the targeting of personal data such as financial records for purposes such as identity theft, fraud, or blackmail. The second class is corporate, which refers to theft of intellectual property, trade secrets, corporate bank information, and disinformation campaigns. Schwartau's third class is global information warfare. Schwartau describes global IW in the following passage:

A well financed, dedicated adversary has the capability—and I emphasize the word capability—to wage war against nation states and political or economic spheres of influence as never before. We will find that international conflict may well be waged on the world's information highways or on our own National Information Infrastructure. We must begin to defend ourselves now.¹¹

A notable cyber-attack that occurred in 1998 was given the name “Moonlight Maze,” in which analysts uncovered the unauthorized accessing of computer systems within universities, research facilities and the federal government. This attack compromised unclassified networks within NASA, the Department of Defense (DOD), and Department of Energy (DOE). Moonlight Maze was believed to have originated in Russia.

Highly placed sources said that the invaders were systematically marauding through tens of thousands of files -- including maps of military installations, troop configurations and military hardware designs. The Defense Department traced the trail back to a mainframe computer in the former Soviet Union but the sponsor of the attacks is unknown and Russia denies any involvement.¹²

“Code Red” was the moniker given to a computer worm that infected numerous networks in 2001. The name “Code Red” was adopted due to the tagline “Hacked by the Chinese.” that defaced the infected websites. Supposedly, Code Red Mountain Dew was also the beverage of choice for the analysts that discovered the worm. The worm's spread disrupted Internet service for businesses and individuals across the globe. It also caused a Denial of Service (DoS) on government systems.

¹⁰ Winn Schwartau, *Information Warfare, Chaos on the Electronic Superhighway* (New York: Thunder's Mouth Press, 1994), 7.

¹¹ *Ibid.*, 20.

¹² *Frontline PBS*, “Cyber War,” April 24, 2003, <http://www.pbs.org/wgbh/pages/frontline/shows/cyberwar/warnings/> (accessed May 10, 2012).

A specific denial of service attack built into the worm prompted the White House to change its IP address. In addition, Code Red Worm's resulting denial of service led DOD to block TCP port 80 traffic originating from non .mil networks destined for the Pentagon and other DOD networks. As a result, non-DOD customers attempting to access DOD web sites were blocked or experienced a severe degradation of service when trying to access government sites. This resulted in numerous customer complaints ranging from the inability to bid on government contracts to difficulties in accessing personnel web sites to apply for government jobs—some complaints even escalating to the congressional level.¹³

The damage from this attack could have been mitigated if the human element had been more proactive in applying Microsoft's security patches. Additionally, the impact of this attack should have served as a catalyst for cyber-security reform, as it highlighted the vulnerability of US networks a decade ago.

Although Microsoft made a patch available nearly a month before the initial outbreak of the Code Red worm, the massive number of infected systems demonstrates the ongoing problem of the failure of system administrators to keep their systems up-to-date with the most recent security patches.¹⁴

Another widely felt cyber-attack was the NIMDA Worm, which was also released in 2001, closely following the September 11th attack on the world trade center. Due to this fact, many suspected links to Al-Qaeda. These suspicions however, were found to be lacking in evidence. Regardless, the NIMDA worm had a significant impact, causing billions of dollars of damage.

The Nimda worm ripped through the U.S. financial sector one week after the Sept. 11, 2001 terrorist attacks. Nimda, which is "admin" spelled backwards, was a mass-mailing worm that exploited vulnerabilities in Microsoft software. It was notable because of its sophistication. It could replicate itself several ways -- by infecting e-mail programs, copying itself onto computer servers, or afflicting users who downloaded infected Web pages. Nimda was also significant for its speed and potency -- it affected millions of computers and slowed the Internet. Officials do not believe it was related to the Sept. 11 attacks.¹⁵

In 2003, the next escalation of cyber-attack took the form of SQL Slammer, also known as the Sapphire Worm. Paul Boutin's 2003 Article, "Slammed! An inside view of the worm

¹³ John C. Dolak, "Security Essentials: The Code Red Worm," SANS Institute (2001), 3. http://www.sans.org/reading_room/whitepapers/malicious/code-red-worm_85 (accessed May 10, 2012).

¹⁴ Ibid.

¹⁵ *Frontline PBS*, "Cyber War," April 24, 2003, <http://www.pbs.org/wgbh/pages/frontline/shows/cyberwar/warnings/> (accessed May 10, 2012).

that crashed the Internet in 15 minutes,” from *Wired Magazine* describes how the global economic damages caused by SQL Slammer totaled over a billion dollars.

Slammer's attack was ruthless and quick, spreading hundreds of times faster than the Code Red virus or Nimda worm. The tiny worm hit its first victim at 12:30 am Eastern standard time. By 12:33 am, the number of slave servers in Slammer's replicant army was doubling every 8.5 seconds. By 12:45 am, huge sections of the Internet began to wink out of existence. Three hundred thousand cable modems in Portugal went dark, and South Korea fell right off the map: no cell phone or Internet service for 27 million people. Five of the Internet's 13 root-name servers - hardened systems, all - succumbed to the squall of packets. Corporate email systems jammed. Web sites stopped responding. Emergency 911 dispatchers in suburban Seattle resorted to paper. Continental Airlines, unable to process tickets, canceled flights from its Newark hub. Lost revenue spilled over halfway into the next week. Total cost of the bailout: more than \$1 billion.¹⁶

Another important consequence of SQL Slammer's release was its effect on the industrial control systems of Ohio's Davis-Besse nuclear power plant. Although the plant was not operational at the time, SQL Slammer's infection of the plant's network illustrates vulnerabilities in critical US infrastructure. Kevin Poulsen described this incident in his 2003 article, “Slammer Worm Crashed Ohio Nuke Plant Network.”

The Slammer worm entered the Davis-Besse plant through a circuitous route. It began by penetrating the unsecured network of an unnamed Davis-Besse contractor then squirmed through a T1 line bridging that network and Davis-Besse's corporate network. The T1 line, investigators later found, was one of multiple ingresses into Davis-Besse's business network that completely bypassed the plant's firewall, which was programmed to block the port Slammer used to spread. Users noticed slow performance on Davis-Besse's business network at 9:00 a.m., Saturday, January 25th, at the same time Slammer began hitting networks around the world. From the business network, the worm spread to the plant network, where it found purchase in at least one un-patched Windows server. According to the reports, plant computer engineers hadn't installed the patch for the MS-SQL vulnerability that Slammer exploited.¹⁷

MS Blaster, another worm launched in 2003, was less damaging than the NIMDA Worm, but was definitely a cause for concern, as it was the first attack to exploit a Remote Procedure Call (RPC) flaw. An analysis by the Antivirus company F-Secure states:

The worm spreads in a 6176 byte executable named MSBLAST.EXE to Windows 2000 and Windows XP systems unless recent Windows security patches have been applied. Windows NT 4 and Windows 2003 might also be affected but these systems appear to be playing a lesser role in the spread of the worm. The worm launches a command shell and uses TFTP to connect to other infected systems to download the worm's executable.

¹⁶ Paul Boutin, “Slammed! An inside view of the worm that crashed the Internet in 15 minutes,” *Wired* (November 7, 2003), <http://www.wired.com/wired/archive/11.07/slammer.html> (accessed May 10, 2012).

¹⁷ Kevin Poulson, “Slammer worm crashed Ohio nuke plant network,” *Security Focus* (2003), <http://www.securityfocus.com/news/6767> (accessed May 10, 2012).

Blaster will scan addresses in the Internet to locate vulnerable Windows machines using TCP/TDP port 135. Once found, it will copy itself over and modify the system so the worm will be executed every time the machine is started. The worm will keep on replicating from every infected machine. Unsuccessful propagation attempts may crash vulnerable computers, or render them unstable.¹⁸

MS Blaster was programmed to launch automated attacks against the website windowsupdate.com, but apparently that site redirected to windowsupdate.Microsoft.com, which allowed for slightly easier mitigation. On a humorous note, MS Blaster contained the text string, “Billy Gates why do you make this possible? Stop making money and fix your software!!”¹⁹ Microsoft did, in fact, release a patch prior to the mass infection. Unfortunately, this patch was also widely unimplemented.

All of these progressively sophisticated attacks represent an evolution of the threat of computer worms in regard to size, complexity of design, and real-world impacts. While the responsibility of actively mitigating such attacks lies predominantly within the IT community, the US government is increasingly becoming cognizant of the necessity of policymakers’ active involvement in securing the national digital infrastructure.

In 2009, President Obama addressed the issue of global information warfare and highlighted the importance of domestic cyber-security efforts to the American people:

From now on, our digital infrastructure -- the networks and computers we depend on every day -- will be treated as they should be: as a strategic national asset. Protecting this infrastructure will be a national security priority. We will ensure that these networks are secure, trustworthy and resilient. We will deter, prevent, detect, and defend against attacks and recover quickly from any disruptions or damage.²⁰

The President outlined five key areas: First, developing “a new comprehensive strategy to secure America's information and communications networks...ensuring a coordinated approach across government, and accountability in federal agencies.”²¹ Second, “working with state and local governments and the private sector to ensure an organized and

¹⁸ John Leyden, “Blaster worm spreading rapidly,” The Register (August 12, 2003), http://www.theregister.co.uk/2003/08/12/blaster_worm_spreading_rapidly (accessed May 10, 2012).

¹⁹ Ibid.

²⁰ Barack Obama, “Remarks by the President on Securing Our Nation's Cyber Infrastructure,” The White House (March 29, 2009), http://www.whitehouse.gov/the_press_office/Remarks-by-the-President-on-Securing-Our-Nations-Cyber-Infrastructure/ (accessed May 11, 2012).

²¹ Ibid.

unified response to future cyber incidents.”²² Third, to “strengthen the public/private partnerships as the vast majority of critical information infrastructure in the United States is owned and operated by the private sector.”²³ Fourth, the “continued investment in cutting-edge research and development. Fifth, “a national campaign to promote cyber-security awareness and digital literacy.”²⁴

These areas of focus are clearly essential to the continued development of domestic cyber-security. However, little is mentioned regarding the specifics of how exactly these goals will be met. In truth, cyber-security is an emerging field that is clouded by ambiguity. This is partially due to the terminology involved, which is often misunderstood by legislators and the general public alike. Moreover, the discourse on the subject is convoluted by semantics, which tend to overstate the veracity of many of the threats in cyberspace.

Analysis of cyber-security issues has been weakened by the lack of agreement on terminology and the use of exaggerated language. An attack or an incident can include anything from an easily identified phishing attempt to obtain password details, a readily detected virus or a failed log-in to a highly sophisticated multi-stranded stealth onslaught.²⁵

At any given moment, multitudes of active threats abound in cyberspace. In order to properly address these threats, it is necessary to distinguish their nature and potential impacts.

There is a broad range of hostile or malicious action in cyberspace – crime, espionage, attacks, and political action. The identity of those who engage in these actions can be indeterminate, and these activities, at some level, often overlap. This does not justify, however, a similar blurring and imprecision in our discussions of cyber conflict. We can reduce this blurring by disaggregating the different kinds of conflict.²⁶

Cyber-war is a term often used in the political arena. According to a report by the Organization for Economic Cooperation and Development “a true cyber-war is an event with

²² Ibid.

²³ Ibid.

²⁴ Ibid.

²⁵ Peter Sommer and Ian Brown, *Future Global Shocks: Reducing Systemic Cybersecurity* (Organization for Economic Co-operation and Development, January 14, 2011), 6. <http://www.oecd.org/dataoecd/57/44/46889922.pdf> (accessed May 11, 2012).

²⁶ James A. Lewis, “Thresholds for Cyberwar,” Center for Strategic and International Studies (September 2010), 3. http://csis.org/files/publication/101001_ieee_insert.pdf (accessed May 11, 2012).

the characteristics of conventional war, but fought exclusively in cyberspace.²⁷” A definition from the US Army’s Cyber Operations Handbook describes cyber-war as:

Premeditated use (or threat) of disruptive activities against computers and/or networks, with the intention to cause harm or further social, ideological, religious, political or similar objectives, or to intimidate any person in furtherance of such objectives.²⁸

These definitions are extremely vague, and many cyber-threats could be categorized within their bounds. In *Thresholds for Cyberwar*, James Lewis narrows the definition to include a necessary component of force, specifically involving a nation-state.

An act of war involves the use of force for political purposes by or against a state. Force involves violence or intimidation (the threatened of the use of force). These are useful thresholds for deciding when an event in cyberspace is an act of war or justifies the use of force. If there is no violence, it is not an attack. If there is no threat of violence, it is not the use of force. In making this distinction, it is important to note the role of clandestine or covert activities. If an opponent intends for a cyber exploit to be undetected, and if the exploit does not inflict physical damage or destruction, it is not intimidation, not the use of force, and not an attack.²⁹

Lewis’ definition is logical but doesn’t distinguish cyber-war as something apart from standard warfare, which has always employed developing technologies. For example, when tank warfare was introduced, it was not referred to as a separate domain. Tanks may have revolutionized the battlefield, but they were merely another tool in a constantly expanding arsenal.

The use of network technologies and the exploitation of cyberspace for intelligence and attack has become a normal part of military activity. Cyber warfare will involve disruption of crucial network services and data, damage to critical infrastructure, and the creation of uncertainty and doubt among opposing commanders and political leaders. Cyber attack provides an ability to strike both tactical and strategic targets from a distance using inexpensive systems. Cyber attacks are unlikely to be decisive and will not by themselves produce victory, particularly against a large and powerful opponent. But they do offer strategic advantage and will be part of future military conflict.³⁰

It is obvious that any modern war would necessarily include a strategy for offensive and defensive maneuvering in cyberspace, but the prospect of a war being fought exclusively in cyberspace seems doubtful. According to the OECD:

²⁷ Sommer and Brown, *Future Global Shocks: Reducing Systemic Cybersecurity*, 6.

²⁸ *US Army Cyber Operations and Cyber Terrorism Handbook 1.02* (Army Training and Doctrine Command, August 15, 2005): II, 2, <http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA439217> (accessed May 11, 2012).

²⁹ Lewis, “Thresholds for Cyberwar,” 3.

³⁰ *Ibid.*, 7.

It is unlikely that there will ever be a true cyber-war. The reasons are: many critical computer systems are protected against known exploits and malware so that designers of new cyber-weapons have to identify new weaknesses and exploits; the effects of cyber-attacks are difficult to predict – on the one hand they may be less powerful than hoped but may also have more extensive outcomes arising from the interconnectedness of systems, resulting in unwanted damage to perpetrators and their allies. More importantly, there is no strategic reason why any aggressor would limit themselves to only one class of weaponry.³¹

Cyber-espionage clearly must be separate from cyber-war, as nation-states spy on each other constantly during peacetime. Surveillance technology for example, used by a nation-state, would not be an act of cyber-war. Simply put, “cyber-espionage is not a few keystrokes away from cyber-war, it is one technical method of spying.”³²

Cyber-crime can be described as the unauthorized use of computer networks and associated technologies for a personal, profit, or political motive. Potentially, any crime using a computer could be classified as cyber-crime. Some common examples are online fraud, cyber-stalking, and data theft.

Cyber-crime is crime that is enabled by, or that targets computers. Some argue there is no agreed-upon definition for “cyber-crime” because “cyberspace” is just a new specific instrument used to help commit crimes that are not new at all. Cyber-crime can involve theft of intellectual property, a violation of patent, trade secret, or copyright laws. However, cyber-crime also includes attacks against computers to deliberately disrupt processing, or may include espionage to make unauthorized copies of classified data.³³

Cyber-terrorism would presumably have a necessary ideological component, distinguishing it from common criminal activity. There is no widely accepted definition for cyber-terrorism, but many argue that its intent would include leveraging the loss of life or economic loss in furtherance of an ideology.

Various definitions exist for the term cyber-terrorism, just as various definitions exist for the term terrorism. Security expert Dorothy Denning defines cyber-terrorism as “politically motivated hacking operations intended to cause grave harm such as loss of life or severe economic damage.” The Federal Emergency Management Agency (FEMA) defines cyber-terrorism as “unlawful attacks and threats of attack against computers, networks, and the information stored therein when done to intimidate or coerce a government or its people in furtherance of political or social objectives.”³⁴

³¹ Sommer and Brown, *Future Global Shocks: Reducing Systemic Cybersecurity*, 7, 8.

³² Ibid.

³³ Clay Wilson, “Botnets, Cybercrime, and Cyberterrorism,” Congressional Research Service (January 29, 2008), 4. <http://www.fas.org/sgp/crs/terror/RL32114.pdf> (accessed May 14, 2012).

³⁴ Ibid., 6.

Terrorists certainly use the Internet as a propaganda tool, and modern technology may assist them in their strategies, but true terrorism solely through cyberspace seems somewhat unlikely. It is doubtful that any small group would be able to cause significant kinetic damage through cyber-attacks alone. Regardless, a major concern among policymakers is the possibility that an insurgent group or rogue state might use cyberspace to attack the US homeland. According to the OECD, this threat is currently somewhat unrealistic:

While the tools are cheap, cyber attack is expensive as it depends on reconnaissance of network targets to find vulnerabilities and this reconnaissance must be periodically refreshed as networks change, add new equipment or software, or are reconfigured. A small nation or insurgent group that is plugged into the cyber underworld may be able to access such information, or to hire mercenaries. The key elements of a cyber attack capabilities are preparation and a fast 'refresh cycle' for targeting. The ability to design and manage a large-scale attack might still be beyond their capacity, but harassment attacks against specific targets – Washington D.C., for example - would be possible. It is also likely that as digital network applications and processing capabilities continue to expand, there may be commercial services and programs that can be adapted to provide small groups with the necessary reconnaissance and planning capabilities.³⁵

However, as technology progresses and costs fall, the threat of cyber-attacks by smaller groups will most certainly increase. The Federal Bureau of Investigation admits that terrorists have been largely ineffective in terms of cyber-attacks, but also stresses the evolving nature of the threat.

To date, the Federal Bureau of Investigation (FBI) reports that cyber-attacks attributed to terrorists have largely been limited to unsophisticated efforts such as email bombing of ideological foes, or defacing of websites. However, it says their increasing technical competency is resulting in an emerging capability for network based attacks.³⁶

A major issue in securing cyberspace is the task of attribution. Uncovering the source of a cyber-attack can prove extremely difficult, as the boundaries between state and non-state actors are often undefined. Additionally, it is difficult to ascertain whether an attack originated from a specific geographic location or a proxy thereof.

A key distinguishing feature of cyber-attacks is that it is often very difficult to identify the actual perpetrator because the computers from which the attack appears to originate will themselves have been taken over and used to relay and magnify the attack commands. This is known as the problem of attribution. An important consequence is that, unlike in conventional warfare, a doctrine of deterrence does not work – because the target for retaliation remains unknown. As a result, defense against cyber-weapons has to

³⁵ Sommer and Brown, *Future Global Shocks: Reducing Systemic Cybersecurity*, 9.

³⁶ Wilson, "Botnets, Cybercrime, and Cyberterrorism," 4.

concentrate on resilience – preventative measures plus detailed contingency plans to enable rapid recovery when an attack succeeds.³⁷

The difficulty of attribution serves as a tool for those engaging in cyber-crime and espionage, as their goal is to remain undetected. Cyber-terrorism however, would presumably be more obvious, as terrorists are often motivated to announce their attacks, or claim credit after the fact.

Many proponents of increased federal cyber-security measures argue that cyber attacks are comparable to weapons of mass destruction. In *Cyber War*, former presidential cyber-security advisor Richard A. Clarke and Richard K. Knake, from the Council on Foreign Relations, argue that the US infrastructure is extremely vulnerable to cyber attacks. They address the issue from the perspective of a worst-case scenario.

Several thousand Americans have already died, multiples of that number are injured and trying to get to hospitals. In the days ahead, cities will run out of food because of the train-system failures and the jumbling of data at trucking and distribution centers. Power will not come back up because nuclear plants have gone into secure lockdown and many conventional plants have had their generators permanently damaged. High-tension transmission lines on several key routes have caught fire and melted. Unable to get cash from ATMs or bank branches, some Americans will begin to loot stores. In all the wars America has fought, no nation has ever done this kind of damage to our cities. A sophisticated cyber war attack by one of several nation-states could do that today, in fifteen minutes, without a single terrorist or soldier appearing in this country.³⁸

Standing in contrast to Clarke and Knake, Jerry Brito and Tate Watkins condemn this alarmist perspective as “cyber-doom.” They critique Clarke’s view, describing his rhetoric as threat inflation.

The picture they paint includes the collapse of the government’s classified and unclassified networks, refinery fires and explosions in cities across the country, the release of “lethal clouds of chlorine gas” from chemical plants, the midair collision of 737s, train derailments, the destruction of major financial computer networks, suburban gas pipeline explosions, a nationwide power blackout, and satellites in space spinning out of control.³⁹

Brito and Watkins argue that there is little evidence in support of these predictions. They recognize the difficulty of presenting such evidence, as much of it would be classified

³⁷ Sommer and Brown, *Future Global Shocks: Reducing Systemic Cybersecurity*, 7.

³⁸ Jerry Brito and Tate Watkins, *Loving the Cyberbomb?* (Fairfax: Mercatus. April 26, 2011), 10. http://mercatus.org/sites/default/files/publication/WP1124_Loving_cyber_bomb.pdf (accessed May 14, 2012).

³⁹ Ibid.

information. Regardless, they point out the fact that Clarke's assumptions are based primarily on a number of Distributed Denial of Service (DDoS) attacks.

The only verifiable evidence they present to support the possibility of a cyber doomsday relates to several well-known distributed denial of service (DDoS) attacks. A DDoS attack works by flooding a server on the Internet with more requests than it can handle, thereby causing it to malfunction. For example, the web server that hosts www.gmu.edu has a certain limited bandwidth and processing capacity with which to serve George Mason University's home page to visitors. If several dozen persons were browsing university web pages and simultaneously requested GMU's homepage, the server would likely perform perfectly well. However, if the server encountered a hundred thousand requests for the home page every second, it would be overwhelmed and would likely shut down.⁴⁰

These attacks utilize botnets, which are networks of personal computers that act in concert, often unbeknownst to the PC owner. However, voluntary opt-in botnets, are often utilized as well. Opt-in botnets have been effectively employed by hacktivists, and cyber-protestors. These attacks are often coordinated through social networking sites like Facebook.

A person carrying out a DDoS attack will almost certainly employ a botnet to cause the massive flood of requests on the attacked server. A botnet is a network of computers that have been compromised without their users' knowledge, usually through a computer virus. The attacker remotely controls these computers and commands them to carry out the attack. Experts have estimated that over 25 percent of personal computers are compromised and form part of a botnet.⁴¹

This type of attack is more properly categorized as a weapon of mass distraction. Causing a website to crash may be annoying, but a DDoS attack can't inflict the type of damage described in Clarke's doomsday scenario.

Slightly more dangerous than weapons of mass distraction, are weapons of mass disruption. These attacks target critical infrastructure, such as the electrical grid or essential utilities. A weapon of mass disruption could seriously impede the nation but it would not cause direct casualties. This type of attack is more practically feasible than a true digital pearl harbor. Nonetheless, Brito and Watkins argue that there is little evidence in support of any serious threat in this regard, criticizing the report by the Center for Strategic and International Studies, *Securing Cyberspace for the 44th Presidency*.

⁴⁰ Ibid., 7.

⁴¹ Ibid.

An enemy able to take down our electric, communications, and financial networks at will could be a serious national security threat. And it may well be the case that the state of security in government and private networks is deplorable. But the CSIS report advances no reviewable evidence to substantiate this supposed threat. There is no evidence in the report that opponents have “mapped vulnerabilities” and “planned attacks.” The probing of DoD computers and the specific cases of cyber espionage that the report cites do not bear on the probability of a successful attack on the electrical grid.⁴²

In truth, the differences between the types of cyber-weapons available, as well as the outcomes of their use, are commonly misunderstood by policy-makers, as well as the general public.

The deployment of cyber-weapons is already widespread and in an extensive range of circumstances. Cyberweapons include: unauthorised access to systems (hacking), viruses, worms, trojans, denial-of-service, distributed denial of service using botnets, root-kits and the use of social engineering. Outcomes can include: compromise of confidentiality/theft of secrets, identity theft, web-defacements, extortion, system hijacking and service blockading. Cyberweapons are used individually, in combination and also blended simultaneously with conventional kinetic weapons as force multipliers. It is a safe prediction that the use of cyberweaponry will shortly become ubiquitous.⁴³

Phishing is a common technique employed by cyber-criminals. Phishing generally involves sending communications from a masked identity, impersonating a trusted user of a network. Typically, the attacker requests their password be reset, or given to them by a system administrator. Spear-phishing is phishing with a specific target. Microsoft describes common instances of phishing on their website:

They might appear to come from your bank or financial institution, a company you regularly do business with, such as Microsoft, or from your social networking site. They might appear to be from someone in your email address book. They might ask you to make a phone call. Phone phishing scams direct you to call a phone number where a person or an audio response unit waits to take your account number, personal identification number, password, or other valuable personal data. They might include official-looking logos and other identifying information taken directly from legitimate websites, and they might include convincing details about your personal history that scammers found on your social networking pages. They might include links to spoofed websites where you are asked to enter personal information.⁴⁴

Targeted attacks are not, however, limited to the distribution of phishing emails. Anonymous, the infamous hacktivist collective, recently conducted a targeted assault on HBGary Federal,

⁴² Ibid., 9.

⁴³ Sommer and Brown, *Future Global Shocks: Reducing Systemic Cybersecurity*, 7.

⁴⁴ “How to Recognize Phishing Email Messages or Links,” Microsoft (2011), <http://www.microsoft.com/security/online-privacy/phishing-symptoms.aspx> (accessed May 15, 2012).

a government cyber-security contractor, after CEO Aaron Barr attempted to publicly name members of the group.

HBGary Federal CEO Aaron Barr thought he had unmasked the hacker hordes of Anonymous and was preparing to name and shame those responsible for co-ordinating the group's actions, including the denial-of-service attacks that hit MasterCard, Visa, and other perceived enemies of WikiLeaks late last year. When Barr told one of those he believed to be an Anonymous ringleader about his forthcoming exposé, the Anonymous response was swift and humiliating. HBGary's servers were broken into, its e-mails pillaged and published to the world, its data destroyed, and its website defaced.⁴⁵

According to the group of hackers, a 16 year-old girl, masquerading as HBGary founder Greg Hoglund, gained access to rootkit.com through a chat session with an administrator. The Anonymous hacker, using Hoglund's e-mail account, was able to deceive the network administrator of the HBGary associated website.

Contained within Greg's mail were two bits of useful information. One: the root password to the machine running Greg's rootkit.com site was either "88j4bb3rw0cky88" or "88Scr3am3r88." Two: Jussi Jaakonaho, "Chief Security Specialist" at Nokia, had root access. Vandalizing the website stored on the machine was now within reach. The attackers just needed a little bit more information: they needed a regular, non-root user account to log in with, because as a standard security procedure, direct ssh access with the root account is disabled. Armed with the two pieces of knowledge above, and with Greg's e-mail account in their control, the social engineers set about their task... To be fair to Jussi, the fake Greg appeared to know the root password and, well, the e-mails were coming from Greg's own e-mail address. But over the course of a few e-mails it was clear that 'Greg' had forgotten both his username *and* his password. And Jussi handed them to him on a platter.⁴⁶

It is surprising that Anonymous was able to deceive the network administrator of an established firm so easily. The fact that Greg not only forgot his password, but also his user-name should definitely have raised suspicions. However, Anonymous was using a trusted email address, and they were able to dump all the password records from rootkit.com using another type of attack called Sequential Query Language (SQL) injection. SQL injection generally involves the inputting of SQL commands into unchecked input fields by an attacker.

The hbgaryfederal.com CMS was susceptible to a kind of attack called SQL injection. In common with other CMSes, the hbgaryfederal.com CMS stores its data in an SQL

⁴⁵ Peter Bright, "Anonymous speaks: The Inside Story of the HBGary Hack," (2011), <http://arstechnica.com/tech-policy/news/2011/02/anonymous-speaks-the-inside-story-of-the-hbgary-hack.ars/3> (accessed May 15, 2012).

⁴⁶ Ibid.

database, retrieving data from that database with suitable queries. Some queries are fixed—an integral part of the CMS application itself. Others, however, need parameters. For example, a query to retrieve an article from the CMS will generally need a parameter corresponding to the article ID number. These parameters are, in turn, generally passed from the Web front-end to the CMS. SQL injection is possible when the code that deals with these parameters is faulty. Many applications join the parameters from the Web front-end with hard-coded queries, then pass the whole concatenated lot to the database. Often, they do this without verifying the validity of those parameters. This exposes the systems to SQL injection. Attackers can pass in specially crafted parameters that cause the database to execute queries of the attackers' own choosing.⁴⁷

HBGary was not wholly incompetent, as their databases did not contain raw password data, instead listing hash values. Hashing is system of password encryption that replaces the password with an alphanumeric sequence for added security.

The attackers grabbed the user database from the CMS—the list of usernames, e-mail addresses, and password hashes for the HBGary employees authorized to make changes to the CMS. In spite of the rudimentary SQL injection flaw, the designers of the CMS system were not completely oblivious to security best practices; the user database did not store plain readable passwords. It stored only hashed passwords—passwords that have been mathematically processed with a hash function to yield a number from which the original password can't be deciphered. The key part is that you can't go backwards—you can't take the hash value and convert it back into a password. With a hash algorithm, traditionally the only way to figure out the original password was to try every single possible password in turn, and see which one matched the hash value you have.⁴⁸

Unfortunately, this system is not fool proof. In order to crack the hashed passwords obtained through SQL injection, Anonymous used databases of pre-computed passwords called rainbow tables.

A technique first published in 2003 gave password crackers an alternative approach. By pre-computing large sets of data and generating what are known as rainbow tables, the attackers can make a trade-off: they get much faster password cracks in return for using much more space. The rainbow table lets the password cracker pre-compute and store a large number of hash values and the passwords that generated them. An attacker can then look up the hash value that they are interested in and see if it's in the table. If it is, they can then read out the password.⁴⁹

As it turns out, HBGary was using one of the most common hashing systems, called MD5, which was highly susceptible to cracking through rainbow tables. Moreover, they did not employ salting, a technique in which extra variables are added to the user's password prior to hashing.

⁴⁷ Ibid.

⁴⁸ Ibid.

⁴⁹ Ibid.

The best known and most widely supported (hashing system) is probably MD5, which is quick to compute and produces an output that is only 128 bits (16 bytes) per hash. These factors together make it particularly vulnerable to rainbow table attacks. A number of software projects exist that allow the generation or downloading of MD5 rainbow tables, and their subsequent use to crack passwords. As luck would have it, the hbgaryfederal.com CMS used MD5. What's worse is that it used MD5 badly: there was no iterative hashing and no salting. The result was that the downloaded passwords were highly susceptible to rainbow table-based attacks.⁵⁰

After obtaining the user-names and passwords of the company's email administrators in this way, the hackers were able to create an effective subterfuge, and with a little social engineering, convinced Jussi to allow them access. An excerpt from Symantec's website describes social engineering based attacks in detail:

Most articles on the topic of social engineering begin with some sort of definition like "the art and science of getting people to comply to your wishes" (Bernz), "an outside hacker's use of psychological tricks on legitimate users of a computer system, in order to obtain information he needs to gain access to the system" (Palumbo), or "getting needed information (for example, a password) from a person rather than breaking into a system" (Berg). In reality, social engineering can be any and all of these things, depending upon where you sit. The one thing that everyone seems to agree upon is that social engineering is generally a hacker's clever manipulation of the natural human tendency to trust. The hacker's goal is to obtain information that will allow him/her to gain unauthorized access to a valued system and the information that resides on that system.⁵¹

Buffer overflow is another type of attack commonly employed by hackers and cyber-criminals. Maciej Ogorkiewicz & Piotr Frej describe this type of attack in an article from WindowsSecurity.com:

Broadly speaking, buffer overflow occurs anytime the program writes more information into the buffer than the space it has allocated in the memory. This allows an attacker to overwrite data that controls the program execution path and hijack the control of the program to execute the attacker's code instead the process code... Programs written in C language, where more focus is given to the programming efficiency and code length than to the security aspect, are most susceptible to this type of attack.⁵²

Another common type of attack, known as a drive-by download, occurs when a website attempts to run JavaScripts that automatically download and install malicious

⁵⁰ Ibid.

⁵¹ Sarah Granger, "Social Engineering Fundamentals," Symantec Security Response (2010), <http://www.symantec.com/connect/articles/social-engineering-fundamentals-part-i-hacker-tactics> (accessed May 16, 2012).

⁵² Maciej Ogorkiewicz and Piotr Frej, "Analysis of Buffer Overflow Attacks," Windows Security (2008), http://www.windowsecurity.com/articles/Analysis_of_Buffer_Overflow_Attacks.html (accessed May 16, 2012).

software (Malware), directly to your computer.

Spy-ware vendors frequently use automated installations of ActiveX controls to distribute their software via web sites. These automated installations are initiated when web surfers land on pages that include HTML code to start the download and installation process. These installations may also be initiated by pop-ups spawned by web pages that users visit. Web users often find these “drive-by-downloads” confusing and disorienting, and it is little wonder that many of them would carelessly click through pop-ups on web sites with very little understanding of the programs they are in fact allowing to be installed on their PCs.⁵³

Malware is a broad term, which can be used to describe many types of malicious software, including viruses, worms, trojan horses, and spy-ware. The above quote describes an instance of drive-by downloading of spy-ware, which typically collects small amounts of personal data, unbeknownst to the user. Many online advertisers utilize these types of programs.

BGP Hijacking is another type of cyber-attack that could conceivably be utilized for purposes of cyber-espionage. This attack was recently employed by the Chinese.

The tactic exploits the Internet routing protocol BGP (Border Gateway Protocol) to let an attacker surreptitiously monitor unencrypted Internet traffic anywhere in the world, and even modify it before it reaches its destination. The attack exploits BGP to fool routers into re-directing data to an eavesdropper’s network. Anyone with a BGP router (ISPs, large corporations or anyone with space at a carrier hotel) could intercept data headed to a target IP address or group of addresses. The method conceivably could be used for corporate espionage, nation-state spying or even by intelligence agencies looking to mine internet data without needing the cooperation of ISPs.⁵⁴

As such, BGP Hijacking should be of particular concern to US intelligence services.

Despite warnings from IT security professionals, the Chinese were able to successfully implement this tactic in 2010. “For 18 minutes in April, China’s state-controlled telecommunications company hijacked 15 percent of the world’s Internet traffic, including data from U.S. military, civilian organizations and those of other U.S. allies.”⁵⁵

Although BGP Hijacking is less than covert, cyber-espionage tactics are of particular concern in regard to China. Presumably, the Chinese have already successfully acquired

⁵³ Eric L. Howes, “The Anatomy of a Drive-by-Download,” (2004), <http://www.spywarewarrior.com/uiuc/dbd-anatomy.htm> (accessed May 16, 2012).

⁵⁴ Kim Zetter, “Revealed: The Internet’s Biggest Security Hole,” *Wired* (2008), <http://www.wired.com/threatlevel/2008/08/revealed-the-in/> (accessed May 16, 2012).

⁵⁵ Stew Magnuson, “Cyber Experts Have Proof That China Has Hijacked U.S.-Based Internet Traffic,” *National Defense Magazine* (2010), <http://www.nationaldefensemagazine.org/blog/Lists/Posts/Post.aspx?ID=249> (accessed May 16, 2012).

significant amounts of data from US networks. An example of this is the recently photographed J-20 stealth fighter produced by the Chinese, which exhibits many features of American aircrafts.

One question that may go unanswered for a long time concerns the degree to which cyber-espionage has aided the development of the J-20. U.S. defense industry cyber-security experts have cited 2006—close to the date when the J-20 program would have started—as the point at which they became aware of what was later named the advanced persistent threat (APT), a campaign of cyber-intrusion aimed primarily at military and defense industries and characterized by sophisticated infiltration and exfiltration techniques.⁵⁶

There is some evidence that the networks of Boeing-Lockheed subcontractors were compromised on multiple occasions.

Between 2009 and early 2010, Lockheed Martin found that ‘six to eight companies’ among its subcontractors ‘had been totally compromised—e-mails, their networks, everything,’ according to Chief Information Security Officer Anne Mullins.⁵⁷

Other glaring examples of Chinese cyber-espionage are Titan Rain and the GhostNet network. Titan Rain was the name given to an incursion, presumably initiated by the Chinese for purposes of cyber-espionage. In “Inside the Chinese Hack Attack,” an article from Time Magazine, Nathan Thornberg describes the incident.

Hackers breaking into official U.S. networks are not just using Chinese systems as a launch pad, but are based in China, sources tell TIME. Their story: Sometime on November 1st, 2004, hackers sat down at computers in southern China and set off once again on their daily hunt for U.S. secrets. Since 2003 the group had been conducting wide-ranging assaults on U.S. government targets to steal sensitive information, part of a massive cyber-espionage ring that U.S. investigators have codenamed Titan Rain.⁵⁸

Initially, the attackers involved in Titan Rain used a scanning program to find vulnerabilities in US computers, specifically targeting military networks. Thornberg goes on to describe the impact of the attacks.

They hit hundreds of computers that night and morning alone, and a brief list of scanned systems gives an indication of the breadth of the attacks. At 10:23 p.m. pacific standard time (PST), they found vulnerabilities at the U.S. Army Information Systems Engineering Command at Fort Huachuca, Arizona. At 1:19 am PST, they found the same hole in computers at the military's Defense Information Systems Agency in Arlington, Virginia.

⁵⁶ Roger Knecht, “China Clones US Stealth Fighters,” (January 12, 2011), <http://www.rogerknecht.com/2011/01/12/china-clones-us-stealth-fighters/> (accessed May 18, 2012).

⁵⁷ Ibid.

⁵⁸ Nathan Thornburgh, “Inside the Chinese Hack Attack,” Time (August 25, 2005) <http://www.time.com/time/nation/article/0,8599,1098371,00.html> (accessed May 18, 2012).

At 3:25 am, they hit the Naval Ocean Systems Center, a defense department installation in San Diego, California. At 4:46 am PST, they struck the United States Army Space and Strategic Defense installation in Huntsville, Alabama. As with prior attacks, the targeted networks were unclassified systems; the military's classified networks are not connected directly to the Internet. But even unclassified systems store sensitive information and provide logistics support throughout the armed forces. Government analysts say the attacks are ongoing, and increasing in frequency. But whether the Titan Rain hackers are gathering industrial information or simply testing their ability to infiltrate a rival nation's military systems, the U.S. government is taking the threat very seriously.⁵⁹

The GhostNet network compromised computer systems in Tibet, Taiwan and several other countries. An article from the New York Times describes GhostNet as:

A broader operation that, in less than two years, has infiltrated at least 1,295 computers in 103 countries, including many belonging to embassies, foreign ministries and other government offices, as well as the Dalai Lama's Tibetan exile centers in India, Brussels, London and New York.⁶⁰

A major game-changer in the realm of cyber security was the recent Stuxnet attack on Iran's nuclear facilities, which will be discussed at length in the case study to follow. It was recently confirmed that Israel and the United States were behind this revolutionary cyber-weapon, which crippled the Iranian nuclear program by destroying over a thousand centrifuges. The Stuxnet worm targeted specific industrial control systems called SCADA (Supervisory Control and Data Acquisition) systems that were separated from the Internet. A method called a candy-drop was presumably employed in which USB thumb drives were used to bridge the air-gap that often isolates sensitive systems.

What's most interesting about Stuxnet isn't how smart its authors were; it's how dumb they guessed we all would be. How did the worm's creators expect to get it inside some of the most secure installations in the world? After all, sensitive machines often operate behind an air gap - that is, their networks are physically separated from the Internet and other dangerous networks where viruses can roam freely. Getting anything inside one of these zones requires the complicity of an employee. That's exactly what Stuxnet got, because its authors designed the worm to piggyback on the perfect delivery system - the ubiquitous, innocent-looking USB flash drive, the planet's most efficient vector of viruses, worms, and other mal-ware. What makes USB drives so great at carrying mal-ware? They're the mosquitoes of the digital world - small, portable, and everywhere, so common as to be nearly invisible.⁶¹

⁵⁹ Ibid.

⁶⁰ John Markoff, "Vast Spy System Loots Computers in 103 Countries," *New York Times*, March 29, 2009. <http://www.nytimes.com/2009/03/29/technology/29spy.html> (accessed May 19, 2012).

⁶¹ "The Hazard of USB Drives," Biz IT Newsletters, (October 14, 2010) <http://www.biznuzz.com/newsletters/?action=article&article=71> (accessed May 19, 2012).

This type of attack has been employed on numerous occasions, against a multitude of targets. Moreover, The United States and its allies have fallen victim to USB candy drops several times, at one point, prompting a ban on the use of thumb drives in military networks.⁶²

STATEMENT OF THE PROBLEM

The Stuxnet worm was brought to the attention of the public in 2010 after being discovered by Belarusian Internet security firm VirusBlokAda. To date, the majority of computer based threats have been motivated by profit and espionage. Stuxnet, however, was clearly developed for the purpose of information warfare. It was revolutionary insofar as it was the first computer virus to cause significant kinetic damage, as it was responsible for rendering over a thousand centrifuges in Iran's Natanz nuclear facility inoperable.⁶³ It was recently confirmed that Stuxnet was developed in a joint effort between US and Israeli intelligence services for the sole purpose of crippling the Iranian uranium enrichment program.⁶⁴

Stuxnet's code targets Siemens' SCADA systems, specifically the systems' Programmable Logic Controllers (PLCs). Although Stuxnet was developed to target the PLCs used with nuclear centrifuges, many computer security experts argue that it could be re-engineered to attack a host of other industrial operations that utilize similar systems. As Stuxnet's code is now easily obtainable, national security concerns have been voiced regarding its potential use by hackers and terrorist organizations. The recent appearance of the W32.Duqu Trojan has legitimized such concerns to a degree, as it greatly resembles Stuxnet's source code. Duqu's purpose however, seems geared towards data collection and reporting. As such, it can be inferred that the Duqu Trojan may be laying the groundwork for future attacks.

⁶² Ibid.

⁶³ Nicolas Falliere, Liam O. Murchu, and Eric Chien, "W32.Stuxnet Dossier," Symantec Security Response (February 2011): 55, http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 5, 2012).

⁶⁴ David E. Sanger, "Obama Order Sped Up Wave of Cyberattacks Against Iran," *New York Times*, June 1, 2012. http://www.nytimes.com/2012/06/01/world/middleeast/obama-ordered-wave-of-cyberattacks-against-iran.html?_r=4&pagewanted=1 (accessed June 15, 2012).

While Stuxnet infection has spread out over the globe, the worm essentially lies dormant until it comes into contact with specific frequency converter drives manufactured only in Iran and Finland. The targeted nature of the worm is made evident by the fact that the majority of worldwide infections occurred within Iran (See Figure 1.).

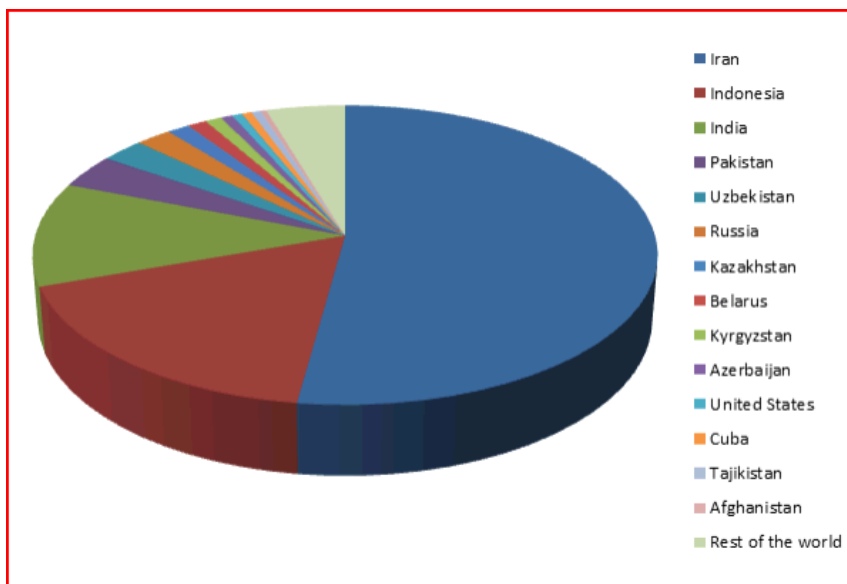


Figure 1. Global infection by W32.Stuxnet. Source: Matrosov, Aleksandr, Eugene Rodionov, David Harley, and Juraj Malcho. "Stuxnet Under the Microscope." ESET (2010). http://go.eset.com/us/resources/white-papers/Stuxnet_Under_the_Microscope.pdf (accessed June 13, 2012).

The major question this study seeks to answer is: Does the Stuxnet worm present a threat to US industrial control systems? The minor questions this study will address are: What is the Stuxnet worm and how does it spread? Why is the release of Stuxnet a significant event in the arena of information warfare? How does Stuxnet differ from previous cyber-threats? What is the potential blowback involved in Stuxnet's release now that its source code is easily obtainable in cyberspace? Finally, how can US industrial control systems be safeguarded from future attacks?

PURPOSE OF THE STUDY

This study is significant largely due to the fact that the Stuxnet worm is the first cyber-weapon of its kind. Many other types of computer based threats have been employed

by foreign governments, hacktivists, and cyber-criminals, but Stuxnet is different insofar as it was a targeted attack, employed by a nation-state, that caused significant kinetic damage. It can thus be considered a proverbial first-strike in terms of malicious code targeting industrial control systems, for the purpose of IW. While Stuxnet has been analyzed in depth by IT professionals, no clear answer has been provided as to how vulnerable the systems within our borders are, should Stuxnet be reverse engineered and modified to attack US infrastructure. This purpose of this study is to examine Stuxnet, with an eye to the potential for similar attacks within the US.

LIMITATIONS OF THE STUDY

This study is limited primarily by the availability of data on the subject of cyber-security, information warfare and the Stuxnet worm itself. Much of the subject matter involved is classified. The willingness of foreign nations to divulge information is also questionable in regard to the effects of computer based threats such as Stuxnet, as well as their respective capacities for in-kind responses by way of offensive cyber-attacks. As such, this study will focus on unclassified and open-source data, largely provided by private security companies and US government entities. The majority of this data has been collected from public domains via the Internet.

DEFINITION OF TERMS

- Information Warfare – Actions taken to achieve information superiority by affecting adversary information, information-based processes, information systems, and computer-based networks while defending one's own information, information based processes, information systems, and computer-based networks.⁶⁵
- Cyber War – Premeditated use (or threat) of disruptive activities against computers and/or networks, with the intention to cause harm or further social, ideological, religious, political or similar objectives, or to intimidate any person in furtherance of such objectives.⁶⁶

⁶⁵ Schwartau, *Information Warfare, Chaos on the Electronic Superhighway*, 7.

⁶⁶ *US Army Cyber Operations and Cyber Terrorism Handbook 1.02*, II, 2.

- Virus – A virus is a code fragment that copies itself into a larger program, modifying that program. A virus executes only when its host program begins to run. The virus then replicates itself, infecting other programs as it reproduces.⁶⁷
- Worm – A destructive program that replicates itself throughout a single computer or across networks (both wired and wireless). It can do damage by sheer reproduction, consuming internal disk and memory resources within a single computer or by exhausting network bandwidth. It can also deposit a Trojan that turns a computer into a zombie for spam and other malicious purposes. Very often, the terms “worm” and “virus” are used synonymously; however, worm implies an automatic method for reproducing itself in other computers.⁶⁸
- Trojan Horse – A Trojan horse is a code fragment that hides inside a program and performs a disguised function. It's a popular mechanism for disguising a virus or a worm.⁶⁹
- Rootkit – A type of Trojan that keeps itself, other files, registry keys and network connections hidden from detection. It enables an attacker to have “root” access to the computer, which means it runs at the lowest level of the machine. A rootkit typically intercepts common API calls. For example, it can intercept requests to a file manager such as Explorer and cause it to keep certain files hidden from display, even reporting false file counts and sizes to the user.⁷⁰
- IW – Information Warfare
- ICS – Industrial Control System
- SCADA – Supervisory Control and Data Acquisition
- HMI – Human Machine Interface
- MTU – Master Terminal Unit
- PLC – Programmable Logic Controller
- RTU – Remote Terminal Unit
- VFD – Variable Frequency Drive
- PROFIBUS – Process Field Bus
- DOD – Department of Defense

⁶⁷ Deborah Russel and G.T. Gangemi, *Computer Security Basics* (Sebastopol: O'Reilly & Associates, 1994), 57.

⁶⁸ “PC Mag Encyclopedia,” PC Magazine, (2012), http://www.pcmag.com/encyclopedia_term/0,2542,t%3Dworm&i%3D54874,00.asp (accessed May 19, 2012)

⁶⁹ Russel and Gangemi, *Computer Security Basics*, 57.

⁷⁰ “PC Mag Encyclopedia,” PC Magazine, (2012), http://www.pcmag.com/encyclopedia_term/0,2542,t%3Dworm&i%3D54874,00.asp (accessed May 19, 2012)

- DOE – Department of Energy
- DHS – Department of Homeland Security
- CIA – Central Intelligence Agency
- NSA – National Security Agency
- OECD – Organization for Economic Cooperation and Development
- SQL – Sequential Query Language
- CMS – Computer Management System
- DDoS – Distributed Denial of Service
- TCP – Transmission Control Protocol
- BGP – Border Gateway Protocol
- RPC – Remote Procedure Call
- ISP – Internet Service Provider
- USB – Universal Serial Bus
- URL – Uniform Resource Locator
- IT – Information Technology
- LAN – Local Area Network
- WAN – Wide Area Network
- P2P – Peer to Peer
- C&C – Command and Control
- VPN – Virtual Private Network
- MD5 – Message Digest Algorithm
- SHA – Secure Hashing Algorithm
- API – Application Programming Interface
- DMZ – Demilitarized Zone
- NPT – (Nuclear) Non-Proliferation Treaty
- IAEA – International Atomic Energy Association

METHODOLOGY

The methodology for this study resembles a typical computer security analysis. The majority of computer based attacks can be examined across five variables: Threat, Exploit,

Vulnerability, Motive and Countermeasure. Typically, a cyber-threat employs exploits against vulnerabilities in a target to obtain a given motive, (be it political or financial) and eventually a countermeasure is employed for mitigation. Prior to the analysis, background on Iran's nuclear aspirations and American involvement in the Middle East will be provided and set within the context of current political theory. The study will then perform a threat assessment, describing the nature of Stuxnet and its target. The exploits Stuxnet uses to affect industrial control systems will be discussed, along with the associated vulnerabilities the worm exploits. The motive for Stuxnet's creation and distribution will be examined. Finally, countermeasures for a Stuxnet type infection will be recommended. The majority of data collected for this study is qualitative by nature, but a limited amount of quantitative data will be presented in regard to Stuxnet's spread.

CHAPTER 2

IRAN'S NUCLEAR ASPIRATIONS AND THE WAR ON TERROR

Iran's nuclear aspirations and former President George W Bush's War on Terror⁷¹ can be commonly categorized as strategic political maneuvers intended to increase domestic security respectively for each nation. The security policy of a given nation is a product of its historical background. Nations, as rational actors, have always sought to secure themselves. However, the framework of the current global power structure is unique to the modern era. Since the end of the cold war, America has become the sole hegemonic power in the world. For the first time, one nation is at the vanguard of international security. Thus, America cannot be judged merely on the efficacy with which it secures its own national interest. Similarly, Iran's unique geographic, cultural and economic roles stand to reason that elements beyond simple self-interest are intrinsic to its security policy as well. To effectively analyze the security policies of a nation, one must take into account its situation across multiple dimensions. One must examine a given nation's security policy holistically, including national interests, international pressures, treaty obligations, as well as moral and cultural imperatives. In support of this position, the relevance of the dominant political paradigms set forth by realists and liberals alike must be re-examined, along with emerging theories, such as social constructivism.

Currently, the United States occupies a decisively unique role in global affairs. As the world's sole hegemon, The US is belied by a long history of international policy which, arguably, can be considered a continuation of previous policies stretching back to the nation's birth. Since the beginning of the twentieth century, many have labeled the US as an essentially imperialist power. A self-interested actor, it continually seeks to increase its

⁷¹ The term coined by George W Bush to describe the international campaign against terrorist networks and their nation state sponsors particularly in regard to combating Islamic extremism. The term also denotes Bush era policies initiated in response to the Sept. 11th 2001 al-Qaeda attacks on the World Trade Center.

power and economic interests through the securing of new markets. Its involvement in the Middle East has been popularly characterized as a means of securing national oil interests. This assumption is not without basis. Specifically, when examining the historical background of Iranian/US relations it becomes evident that the current Iranian administration is a direct reaction to prior American involvement. Historically there are many examples of US installation of puppet governments serving to protect American interests abroad. In 1953, when Iranian Prime Minister Mohammad Moseddeq was ousted by coup, it was largely to protect British and American oil interests in Iran. Dissatisfied with this pro-western arrangement, Iran eventually embraced the Ayatollah Khomeini's overthrow of the Shah in 1979. In the 1980s, economic and military aid was supplied to freedom fighters under the Reagan doctrine. During this era, aid was supplied in part by the covert sales of arms to Iran. Labeled the Iran-Contra Affair, the US funneled these arms monies to revolutionaries in Nicaragua. More importantly, the US effectively played both sides of the coin during the Iran-Iraq war. Given the history of American involvement, it stands to reason that Iran should, as a rational actor, be duly concerned with securing its own regional and global interests. Claims of Western imperialism, especially in relation to the US roles in Israel and Iraq, are not entirely devoid of merit.⁷²

The policy decisions set forth by President George W Bush's War on Terror were cause for great concern in the Middle East. From many Middle Easterner's perspectives, Bush's policies entailed treaty violations, constitutional hypocrisy, and morally reprehensible acts. America's support for Israel, involvement in Afghanistan, and occupation of Iraq were also serious regional concerns for Iran. Due to these facts, it stands to reason that Iran might seek nuclear weapons in an attempt to deter invasion and increase its diplomatic bargaining power. Iran's dissatisfaction with the Non-Proliferation Treaty (NPT) and subsequent stalemates with the International Atomic Energy Association (IAEA), it claims, are rooted in the lack of support for the NPT by many nuclear powers, as well as NPT's failure to address vertical proliferation and actual arms reduction. Iran's official position is that it merely seeks to develop nuclear technology for peaceful purposes. However, given the nature of US and Israeli interests in the region, it is highly doubtful that this is the case.

⁷² Latha Vardarajan, "National Security Policy" (lecture, San Diego State University, 2009).

Kenneth Waltz, credited as the originator of neo-realism, describes his theoretical model as “structural realism,”

Kenneth Waltz’s formulation of a neorealist theory has had a profound effect on the field of security studies. Waltz’s theory is explicitly structural. It argues that the international state system molds states and defines the possibilities for cooperation and conflict.⁷³

Simply put, states are driven to act in response to pressure from global competitors. John J. Mearsheimer refers to his model of national security as “offensive realism,”

Its elements are few and can be distilled in a handful of simple propositions. For example, I emphasize that great powers seek to maximize their share of world power. I also argue that multipolar systems which contain an especially powerful state - in other words, a potential hegemon - are especially prone to war.⁷⁴

Essentially, the realist perspective argues that nations will always act in their own self-interest, and seek to occupy the role of hegemon. The War on Terror, taken *prima facie*, would make little sense to a realist in any other context. While terrorists clearly present a threat, they can be properly addressed through intelligence gathering, practical threat assessment, and covert actions. Realists rejected the legitimacy of the war in Iraq, viewing it as a colossal mistake. To a realist, war is simply a last resort, and should only be employed when victory is easily foreseeable. Both Mearsheimer and Waltz were signatories to a 2002 letter to President Bush urging against intervention in Iraq. They sighted the absence of an exit strategy, lack of WMD evidence, potential regional destabilization, the US ability to contain Saddam Hussein, and the superior threat of Al-Qaeda. In this regard, parallels can be drawn between the realist perspective and the security policies advocated by the Colin Powell doctrine in the run up to the first Gulf War.

The neo-liberal view takes the position that the spread of democracy and free-market capitalism are essential to peace and stability. By this justification, the US war in Iraq was justified by moral imperative. Ousting a cruel dictator in favor of establishing democratic rule was clearly morally justifiable. Similarly, the liberal perspective argues that fighting the War on Terror was essential, as the core values of democracy were being challenged. Fascism and communism may have all but dissipated, but Islamic fundamentalism and

⁷³ Peter Katzenstein, *The Culture of National Security: Norms and Identity in World Politics* (New York: Columbia University Press, 1996), 12.

⁷⁴ John J Mearsheimer, *The Tragedy of Great Power Politics* (New York: W.W. Norton, 2001), 7.

terrorism remain serious security threats. For this reason, liberals argue that terrorists must be fought on their own terms, which may be morally reprehensible or unconstitutional. One could easily infer that the neo-liberal view would also advocate US military intervention in Iran to deter the current regime's aspirations for nuclear weapons, and support the secular revolutions of the Iranian populous.

The single underlying constant in US security policy is economic expansionism.⁷⁵ The US concern for finding and securing new markets is of essential importance. This principle is manifested by the US "strategy of openness," a series of successive policies characterized through globalization and free trade. Bacevich states that "an open world that adheres to the principles of free enterprise is a precondition for continued American prosperity"⁷⁶ In contrast, Vladimir Lenin argued that free trade does not, in fact, promote peace. Lenin's position is that the spread of liberal capitalism actually serves to promote global insecurity by concentrating power and wealth in the hands of the few. Undoubtedly, Lenin would view US involvement in the Middle East as an extension of imperialist policies stretching back to the turn of the 20th century.

Colonial possession alone gives the monopolies complete guarantee against all contingencies in the struggle with competitors, including the contingency that the latter will defend themselves by means of a law establishing a state monopoly.⁷⁷

Such a state monopoly was created in Iran when Moseddeq nationalized the oil industry, wresting control from British Petroleum. This action was the primary motivation for the coup of 1953, in which the US and Britain covertly overthrew Moseddeq. As such, the current Iranian regime would clearly support Lenin's argument.

Iran's status quo is a product of economic, political, social, and cultural factors unique to its location and history. To understand its security policy, or that of any nation, Peter Katzenstein offers a different perspective, social constructivism.

Katzenstein proposes that social constructivism focuses on two under-attended determinants of national security policy: the cultural institutional context of policy on the

⁷⁵ Andrew Bacevich, *American Empire* (Cambridge: Harvard University Press, 2004).

⁷⁶ *Ibid.*, 3.

⁷⁷ Vladimir Lenin, *Imperialism: The Highest Stage of Capitalism* (London: Pluto Press, 1996), 98.

one hand and the constructed identity of states, governments, and other political actors on the other.⁷⁸

According to Katzenstein, national security must be defined within its specific historical and political context. To understand the context of security policy, one must understand that national interests are not static, but dynamic in nature.

Katzenstein states that structural neo-realism and institutional neo-liberalism share a similar framework, susceptible to the same weakness...neorealist and neoliberal perspectives focus on how structures affect the institutional rationality of actors.⁷⁹

These perspectives both fall short in accounting for social facts, culture, and state identity.

Katzenstein argues that neo-liberalism takes as given actor identities and views ideas and beliefs as intervening variables between assumed interests and behavioral outcomes...neo-realism is too general and underspecified to tell us anything about the direction of balancing, let alone about the content of the national security policies of states.⁸⁰

According to social constructivism, the contention that one should judge the security policies of the US and Iran merely by how effectively the ruling parties have maximized self-interest for their respective nations is somewhat shortsighted. To judge a particular nation's security policies effectively, one must look at other variables, including history, state identity, and the ability of a nation to function within its region, as well as the international arena. These facts have become increasingly evident in the aftermath of the recent "Arab-spring" uprisings in the region.

One could argue that neither nation has become more secure as a result of their policies. The US is drowning in debt, overextended militarily, and making little headway in promoting global peace and stability. The current Iranian regime will likely pursue the development of nuclear weapons regardless of sanctions or popular revolutions. While the US traditionally seeks to deter non-nuclear states from acquisition of nuclear weapons, the current situation with Iran stands to reason that overt coercive measures will be ineffective.

According to Katzenstein the relationship between non-use of nuclear weapons and non-acquisition is explicitly embodied in the Nuclear Non-Proliferation Treaty and in various

⁷⁸ Katzenstein, *The Culture of National Security: Norms and Identity in World Politics*, 12.

⁷⁹ *Ibid.*, 13.

⁸⁰ *Ibid.*, 25, 26.

commitments by the nuclear powers not use nuclear weapons against non-nuclear powers who are party to the treaty.⁸¹

This premise implies that the nature of non-use is essentially normative. This non-use norm serves to further exemplify Iran's nuclear aspirations as illegitimate. Clearly, if Iran so chose, it could abandon its goal of acquisition, avoid sanctions, and use its increased economic power to protect itself by non-nuclear means. This is, however, unlikely due to normative reasons as well. Given the Iranian regime's cultural institutional context, this simply would not happen. American support for the nation of Israel is a major variable in this equation. In a struggle of diametrically opposed ideologies, Iran would be unlikely to submit to nuclear apartheid and bow to America's will. Regardless, the prospect of US military intervention in Iran should be nothing short of a measure of last resort. Lessons learned through the occupations of Afghanistan and Iraq were a testament to this fact. These looming realizations reinforce the validity of the employment of targeted covert strikes on Iran's nuclear program as being a more desirable course of action. Undoubtedly, there will be blowback involved with covert actions such as the assassination of nuclear physicists, or the release of directed cyber-weapons like Stuxnet, but these types of actions will ultimately cause less collateral damage to the Iranian population and delay, if not prevent future military intervention.

⁸¹ Ibid., 117.

CHAPTER 3

INDUSTRIAL CONTROL SYSTEMS

While Stuxnet was designed to specifically affect the control systems of Iran's uranium enrichment facilities, computer-based industrial control systems (ICSs) are found in a variety of industrial facilities around the world. Many of these operations involve critical infrastructure such as power production and distribution, water treatment, waste management, oil pipelines and refinement, as well as the manufacturing industry, airports, and even correctional facilities. As such, securing these sensitive systems from cyber attacks, whether motivated hacktivism, terrorism, or nation-state sabotage, is of the utmost importance. In order to properly convey security threats to industrial control systems, a basis must be provided as to the architecture of the systems themselves. As noted earlier in this work, SCADA stands for Supervisory Control and Data Acquisition. A 2004 Technical Information Bulletin from the National Communications System describes SCADA systems in the following manner:

These systems encompass the transfer of data between a SCADA central host computer and a number of Remote Terminal Units (RTUs) and/or Programmable Logic Controllers (PLCs), and the central host and the operator terminals. A SCADA system gathers information (such as where a leak on a pipeline has occurred), transfers the information back to a central site, then alerts the home station that a leak has occurred, carrying out necessary analysis and control, such as determining if the leak is critical, and displaying the information in a logical and organized fashion. These systems can be relatively simple, such as one that monitors environmental conditions of a small office building, or very complex, such as a system that monitors all the activity in a nuclear power plant or the activity of a municipal water system.⁸²

A typical SCADA system is comprised of a number of elements. A human machine interface (HMI), which is the computer system and associated software through which data is presented to the system operators. The operators monitor the system's data and send commands to the field devices through this interface. The field data interface devices are the

⁸² Dale Barr, *Technical Information Bulletin 04-1 Supervisory Control and Data Acquisition Systems* (Arlington: National Communications System, 2004), 4. http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf (accessed June 3, 2012).

RTUs and PLCs. These devices “interface to field sensing devices and local control switchboxes and valve actuators.”⁸³ They convert the signals from these sensors into digital data and transmit the data back to the Master Terminal Unit (MTU). The MTU is the “central host computer server or servers (sometimes called a SCADA Center).”⁸⁴ The MTU is essentially the intermediary between the HMI and the field devices. The final element is the communications infrastructure of the system, which is “used to transfer data between field data interface devices and control units and the computers in the SCADA central host. The system can be radio, telephone, cable, satellite, or any combination of these.”⁸⁵ Figure 2 is a visual representation of a typical SCADA system.

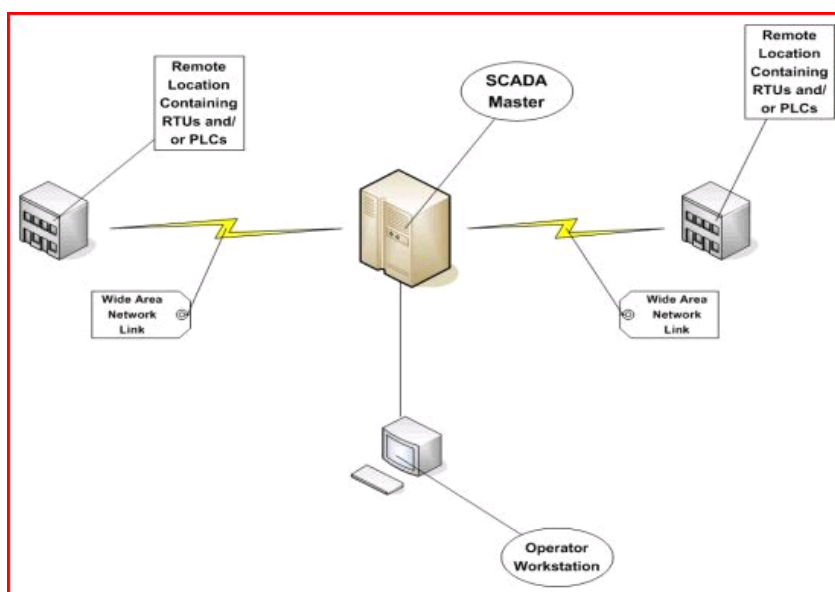


Figure 2. Typical SCADA system. Source: Barr, Dale. *Technical Information Bulletin 04-1 Supervisory Control and Data Acquisition Systems*. Arlington: National Communications System, 2004. http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf (accessed June 3, 2012).

⁸³ Ibid.

⁸⁴ Ibid.

⁸⁵ Ibid.

There have been three phases in the evolution of SCADA architecture thus far. The first generation of SCADA architecture is referred to as monolithic. Typical monolithic SCADA architecture is represented in Figure 3.

When SCADA systems were first developed, the concept of computing in general centered on mainframe systems. Networks were generally non-existent, and each centralized system stood alone. As a result, SCADA systems were standalone systems with virtually no connectivity to other systems. The Wide Area Networks (WANs) that were implemented to communicate with RTUs were designed with a single purpose in mind—that of communicating with RTUs in the field and nothing else. In addition, WAN protocols in use today were largely unknown at the time.⁸⁶

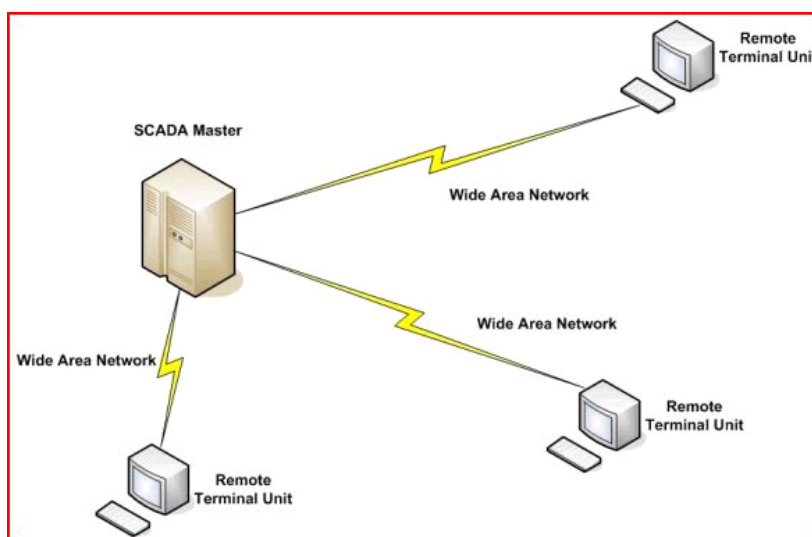


Figure 3. First generation SCADA architecture. Source: Barr, Dale. *Technical Information Bulletin 04-1 Supervisory Control and Data Acquisition Systems*. Arlington: National Communications System, 2004. http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf (accessed June 3, 2012).

Second generation SCADA systems are referred to as distributed systems. These systems evolved with the increasing implementation of local area networks (LANs). In this model, multiple workstations were interconnected, enabling real-time data-sharing. This enabled multiple HMIs to function within the system. See Figure 4 for a visual depiction of second generation SCADA architecture.

⁸⁶ Ibid., 10.

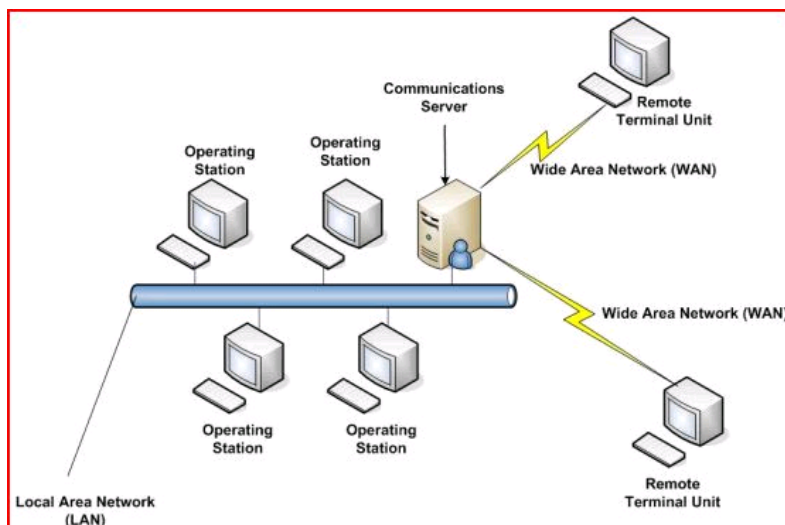


Figure 4. Second generation SCADA architecture. Source: Barr, Dale. *Technical Information Bulletin 04-1 Supervisory Control and Data Acquisition Systems*. Arlington: National Communications System, 2004. http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf (accessed June 3, 2012).

Distribution of system functionality across network-connected systems served not only to increase processing power, but also to improve the redundancy and reliability of the system as a whole. Rather than the simple primary/standby failover scheme that was utilized in many first generation systems, the distributed architecture often kept all stations on the LAN in an online state all of the time. For example, if an HMI station were to fail, another HMI station could be used to operate the system, without waiting for failover from the primary system to the secondary.⁸⁷

The third generation model is the networked SCADA system. These systems are similar to the second generation, but differ insofar as in previous models the networking technologies employed were typically proprietary, whereas third generation systems utilize open protocols (See Figure 5).

The major improvement in third generation SCADA systems comes from the use of WAN protocols such as the Internet Protocol (IP) for communication between the master station and communications equipment. This allows the portion of the master station that is responsible for communications with the field devices to be separated from the master station across a WAN. Vendors are now producing RTUs that can communicate with the master station using an Ethernet connection.⁸⁸

⁸⁷ Ibid., 12.

⁸⁸ Ibid., 13.

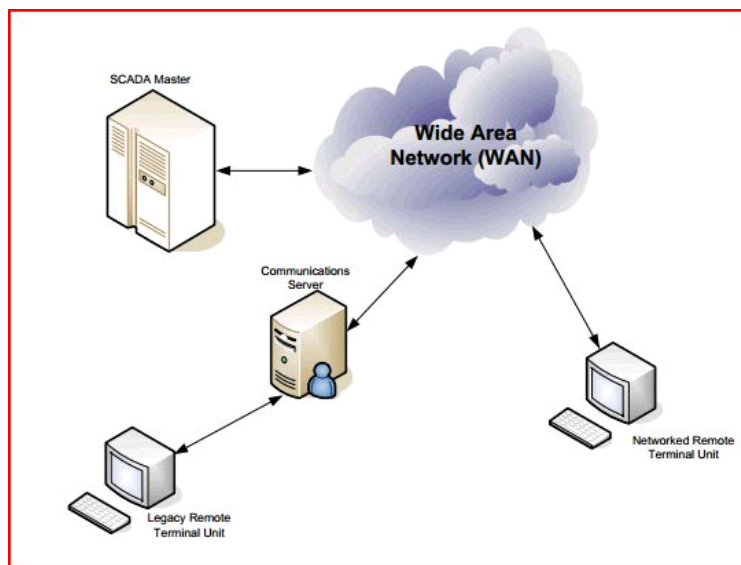


Figure 5. Third generation SCADA architecture.
Source: Barr, Dale. *Technical Information Bulletin 04-1 Supervisory Control and Data Acquisition Systems*. Arlington: National Communications System, 2004. http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf (accessed June 3, 2012).

Many third generation SCADA systems are linked with the networks of corporate offices and accessible through the Internet. This practice seems less than ideal, and makes securing these systems from remote attacks extremely important. However, one security advantage of networked SCADA architecture is that it is more resilient in terms of disaster mitigation. “By distributing the processing across physically separate locations, it becomes possible to build a SCADA system that can survive a total loss of any one location.”⁸⁹

The specific ICS architecture targeted by Stuxnet is the SIMATIC PCS 7 system developed by the German company Siemens. The system is comprised of an integrated package of software and hardware.

SIMATIC is a comprehensive term used by Siemens, which includes their complete portfolio of industrial automation solutions ranging from machine vision to distributed I/O systems and programmable controllers. SIMATIC WinCC is a specialized process visualization system that comprises the core SCADA system. It can be used with Siemens – branded control equipment, such as the S7 line of PLC’s or it can be used independently with other control products. The SIMATIC STEP 7 software environment

⁸⁹ Ibid.

is used specifically for the programming of the Siemens S7 line of controllers. An integrated solution, composed of S7 PLC's, WinCC visualization software, and STEP 7 configuration software, is then referred to as SIMATIC PCS 7. All computer software components run on Microsoft Windows operating systems, including XP, Server 2003, and Windows 7.⁹⁰

The SIMATIC PCS 7 system is divided into three subsystems: The Operator System, Automation System and Engineering System (See Figure 6).

The Operator System permits the secure interaction of the operator with the process under control of PCS 7. Operators can monitor the manufacturing process using various visualization techniques to monitor, analyze and manipulate data as necessary. The Automation System is the name given to the class of programmable logic controllers used with PCS 7. The Engineering System consists of software that is responsible for configuring the various PCS 7 system components.⁹¹

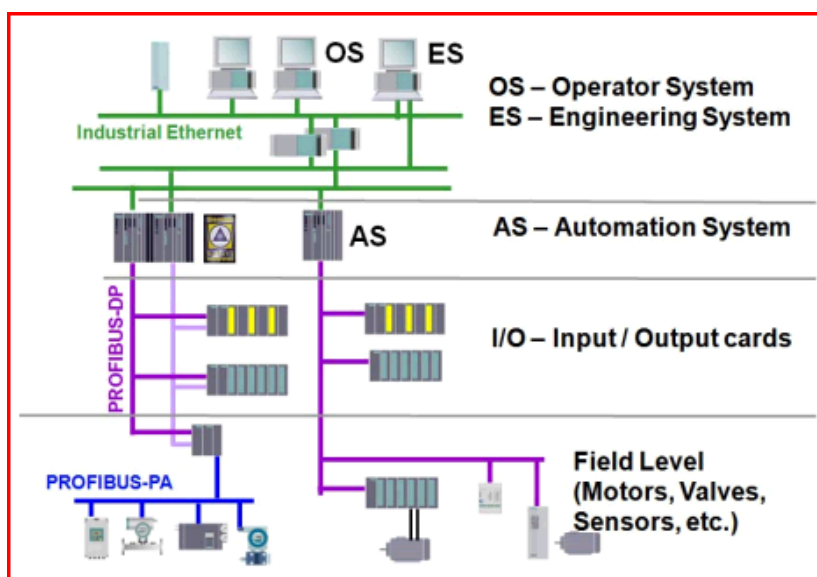


Figure 6. Components of the SIMATIC PCS 7. Source: Byres, Eric P., Andrew Ginter, and Joel Langill. *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems*. Alberta: Tofino Security/Abterra Technologies, 2011. <http://abterra.ca/papers/How-Stuxnet-Spreads.pdf> (accessed June 3, 2012).

⁹⁰ Eric P Byres, Andrew Ginter, and Joel Langill, *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems* (Alberta: Tofino Security/Abterra Technologies, 2011) <http://abterra.ca/papers/How-Stuxnet-Spreads.pdf> (accessed June 3, 2012).

⁹¹ Ibid., 8.

As many of the sites utilizing the SIMATIC PCS 7 would be considered high security operations, Siemens recommends a set of “best practice” security measures for users of the system. According to Siemens, a high security operation should be divided into at least four security zones. Figure 7 details a typical partitioning scheme for a high security industrial site:

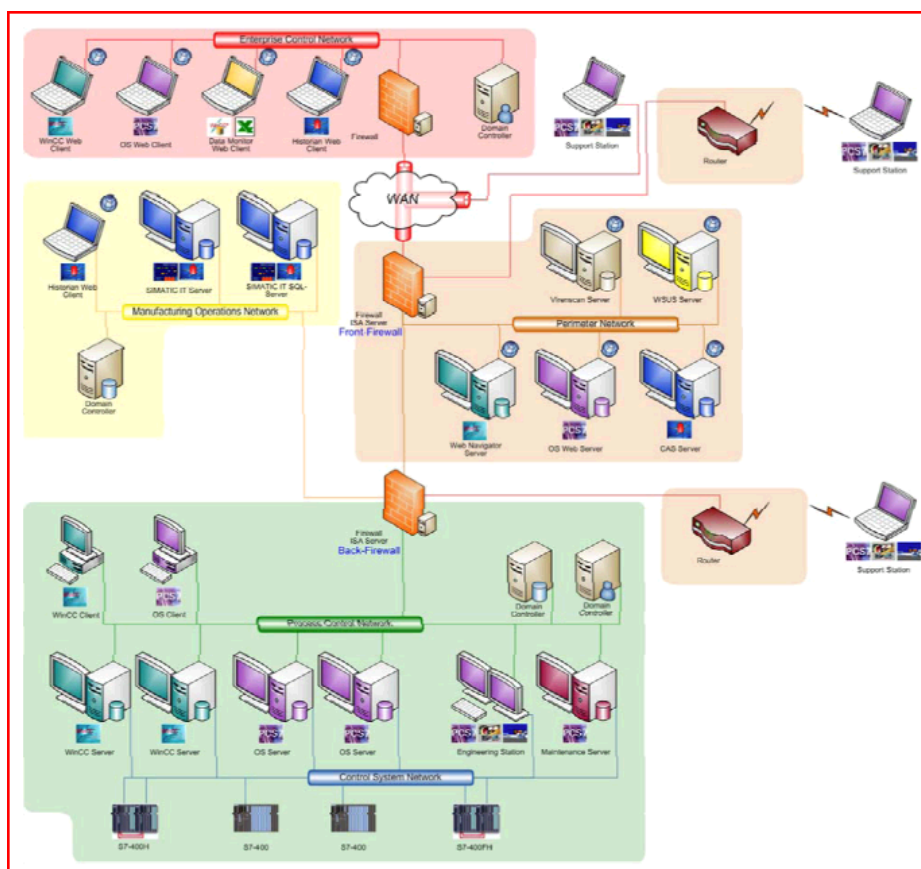


Figure 7. Siemens’ protocol for high security sites. Source: Byres, Eric P., Andrew Ginter, and Joel Langill. *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems*. Alberta: Tofino Security/Abterra Technologies, 2011. <http://abterra.ca/papers/How-Stuxnet-Spreads.pdf> (accessed June 3, 2012).

The first zone, highlighted in pink, is the Enterprise Control Network. This is the corporate network “which hosts most business users and business accounting and planning systems, such as Enterprise Resource Planning (ERP) systems.”⁹² The second zone,

⁹² Ibid., 9.

highlighted in yellow, is the Manufacturing Operations Network. This network “hosts the SIMATIC IT servers, which exchange information between the control system, the ERP system, and other important applications on the Enterprise Control Network.”⁹³ The third zone, highlighted in brown, is the Perimeter Network. This network “hosts servers that manage equipment in the ICS, and servers that provide information to end users on the Enterprise Control Network.”⁹⁴ Typically, the Perimeter Network is responsible for the servers that manage patching and security updates “Including Windows security updates and anti-virus updates. Many of the servers within this zone provide information to end users via web servers and web services. People sometimes refer to this zone as a demilitarized zone or DMZ.”⁹⁵ The fourth zone, highlighted in green, hosts two networks, the Process Control Network and the Control System Network.

The Process Control Network hosts the 24x7 plant operators on their Human Machine Interface (HMI) workstations, and is also connected to the WinCC/PCS 7 control system servers. The Control System Network is connected to a number of Programmable Logic Controllers (PLCs) and is also connected to the WinCC/PCS 7 control system servers. In a large facility, there are frequently multiple “green” zones, one for each control center or operating area.⁹⁶

Sites are connected together via the corporate WAN, which also interconnects the various security zones within each site. The networks are managed by corporate IT and separated by firewalls which protect each individual zone. Additionally, Microsoft Internet Security and Acceleration Servers are implemented in order to shield higher security networks.

Microsoft Internet Security and Acceleration (ISA) Servers protect the plant zones from the WAN. They also protect zones from each other. All traffic between security zones passes through an ISA server. Each ISA server hosts a number of functions, such as firewall services, network address translation, web proxies, virus scanning and secure web server publishing. All of the ISA servers are configured by default to block connections originating in less-trusted networks, such as the corporate WAN. The ISA servers allow connections, such as web services connections, from clients on less-trusted networks to selected servers, such as web servers, in the Perimeter Network.⁹⁷

⁹³ Ibid.

⁹⁴ Ibid.

⁹⁵ Ibid.

⁹⁶ Ibid.

⁹⁷ Ibid., 10.

These ISA servers also manage incoming connections via Virtual Private Networks (VPNs), which are represented in Figure 7 as support stations. VPNs are essentially encrypted communication tunnels utilizing public infrastructure, typically the internet, to allow remote connections through the Perimeter Network. Access through VPNs typically requires user authentication.

Support stations are used most commonly for remote engineering activities or vendor support activities. The stations may be at the site, or at a remote corporate site, connected indirectly to the corporate WAN, with their access into corporate networks other than the WAN mediated by either corporate firewalls or the ISA servers. The vendors may also be at other non-corporate remote sites, connecting directly to the ISA servers from quarantine zones served by routers. When these support stations access protected network zones through an ISA firewall, the firewall authenticates the VPN connection. If the vendor uses WinCC or other process applications that require access to the Process Control Network, the firewall allows a small number of connections, including WinCC and STEP 7 database connections, to protected servers.⁹⁸

The use of these multiple layers of security begs the question: How exactly was Stuxnet able to penetrate such a highly secured site? In solving this conundrum, there are a couple key variables which will be addressed in the next chapter. First, Stuxnet had the element of surprise. Nobody knew of its existence, therefore there were no pre-existing anti-virus signatures in vendor databases with which to compare. Second, Stuxnet exploited four previously unknown vulnerabilities in Microsoft's Windows OS.

⁹⁸ Ibid., 11.

CHAPTER 4

THE STUXNET WORM

Stuxnet is a computer worm. As defined earlier, a worm is a type of malicious code that has the ability to self replicate and spread across computer networks. Prior instances of this type of attack include the Morris worm, Nimda, Code Red, MS Blaster, SQL Slammer, and Conficker. The Stuxnet worm however, represents a significant point of departure from its predecessors. “Stuxnet represents the first of many milestones in malicious code history – it is the first to exploit four zero-day vulnerabilities, compromise two digital certificates, and inject code into industrial control systems and hide the code from the operator.”⁹⁹ It can be inferred that Stuxnet also represents a proverbial “first-strike” in the arena of cyber-war, as it caused significant kinetic damage, and was engineered by a nation-state (or combination thereof). A timeline in relation to Stuxnet’s discovery is provided in Appendix A.

Stuxnet was initially discovered by Belarusian security firm VirusBlokAda. It was reported to Microsoft in June of 2010. Bruce Dang was the lead analyst on the vulnerability testing team that Microsoft tasked with unraveling Stuxnet’s code. It was this team that uncovered Stuxnet’s unprecedented exploitation of four zero-day vulnerabilities in Microsoft’s Windows OS. A zero-day vulnerability, simply put, is a previously unknown flaw in a piece of software’s code. As zero-days are unknown, the developers of the software, Microsoft in this case, would not have already created a patch to fix the vulnerability. Dang describes the process of uncovering these vulnerabilities in a 2010 lecture at a meeting of Germany’s Chaos Computer Club.

The code that had been provided to the team was large — close to 1 MB of information, Dang said. A team of 20 to 30 people with expertise in various components of the Windows system was assembled and began quickly exchanging emails.¹⁰⁰

⁹⁹ Falliere, O Murchu, and Chien, “W32.Stuxnet Dossier,” 55.

¹⁰⁰ John Borland, “A Four-Day Dive Into Stuxnet’s Heart,” *Wired*, (December 27, 2010), <http://www.wired.com/threatlevel/2010/12/a-four-day-dive-into-stuxnets-heart/> (accessed June 5, 2012).

The first vulnerability Dang's team found related to one method by which Stuxnet propagates, the ubiquitous USB stick. As stated earlier, many infer that it was via USB stick that Stuxnet found its way into Iran's air-gapped Natanz uranium enrichment facility.

They traced the apparent problem to code that came from an infected USB stick. By exploiting a vulnerability in the Windows icon shortcut feature, or LNK files, the worm gained the ability to execute commands on the infected computer, but only with the current user's level of access.¹⁰¹

According to Dang, "this vulnerability had been known for several years by some people."¹⁰² Nonetheless, Microsoft neglected to create a patch. After noticing suspicious system (DLL) files emerging, Dang's team deepened their analysis and eventually uncovered a second zero-day. The implications of this vulnerability were far more dangerous than the first.

They noticed that extra drivers were being installed on their test computers, both in Windows XP and Windows 7 environments. Closer investigation showed that scheduled tasks were being added, and XML-based task files were being created and rewritten. Working with a colleague overseas, Dang discovered that the way Windows Vista and later operating systems stored and verified scheduled tasks contained a vulnerability that would give the attacking worm (which had already gained the ability to drop code with user-based access privileges) the ability to give itself far broader — and thus more dangerous — privileges on the infected computer. In short, the two flaws working together allowed the worm to gain code-execution privileges, and then to deepen those privileges to install a rootkit.¹⁰³

This vulnerability worked in tandem with the first zero-day to escalate user privileges and install a rootkit on the infected computer. As defined earlier, a rootkit has the ability to conceal itself, other processes and network connections, as well as provide administrator privileges. In this manner, Stuxnet hides itself from detection on infected removable drives prior to their use on another computer. Dang's team was able to mitigate this vulnerability "by changing the way the Vista and Windows 7 task scheduler uses hash values to verify files."¹⁰⁴ However, their work was far from finished. Eventually, they noticed an anomaly in the way a DLL file was being loaded in Windows 7 versus Windows XP.

The team identified a flaw in the way Windows XP systems are allowed to switch user keyboard layouts — from an English keyboard to a German configuration, for example. Once again, this allowed the worm to gain elevated privileges on the

¹⁰¹ Ibid.

¹⁰² Ibid.

¹⁰³ Ibid.

¹⁰⁴ Ibid.

infected computer. Smart, almost chillingly so, Dang said. The task-scheduling attack previously identified worked only on Vista and later systems. The keyboard layout attack worked only on XP. Some people somewhere had set their sights very broadly.¹⁰⁵

The fact that each of these exploits was designed to take advantage of vulnerabilities in different incarnations of Windows meant that the scope of the attack was large, as the worm had the ability to infect multiple platforms. The fourth zero-day was discovered after Dang's team was contacted by Kaspersky Labs.

The team got word from the Kaspersky Lab security company that there was strange "remote procedure call" traffic being sent over a network — a kind of communication that allows one computer to trigger activity on another, such as printing from a remote device. Dang and his team set up a mini-VPN, infected one computer, and went away. They came back to find their entire mini-network had been infected... They brought Microsoft's printer team in, and this time the problem proved simple to uncover. In 5 minutes they had traced the source: a print-spooler flaw that allowed remote guest accounts to write executable files directly to disk. A terrible flaw, but luckily fixed quickly. The flaw gave more insight into the attacker's intentions. The configuration vulnerable to this flaw was very uncommon in normal corporations, but allowed widespread infection within a network that was configured in this way.¹⁰⁶

This fourth zero-day is another method by which Stuxnet propagates. However, there are additional methods of propagation that Stuxnet employs aside from infected removable drives and print spooler services. Stuxnet also spreads via peer-to-peer (P2P) communication, as represented in Figure 8.

The P2P component works by installing an RPC server and client. When the threat infects a computer it starts the RPC server and listens for connections. Any other compromised computer on the network can connect to the RPC server and ask what version of the threat is installed on the remote computer. If the remote version is newer then the local computer will make a request for the new version and will update itself with that. If the remote version is older the local computer will prepare a copy of itself and send it to the remote computer so that it can update itself. In this way an update can be introduced to any compromised computer on a network and it will eventually spread to all other compromised computers.¹⁰⁷

¹⁰⁵ Ibid.

¹⁰⁶ Ibid.

¹⁰⁷ Falliere, O. Murchu, and Chien, "W32.Stuxnet Dossier," 25.

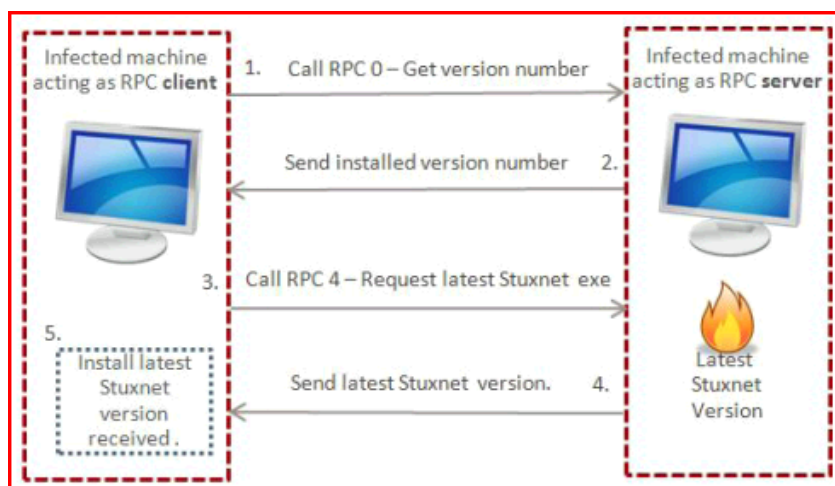


Figure 8. Stuxnet P2P communication. Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. “W32.Stuxnet Dossier.” Symantec Security Response (February 2011). http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

This method of propagation not only ensures that each infected computer is updated with the latest version of Stuxnet, but also allows Stuxnet to reach computers that are connected to each other via LAN but not connected to the Internet individually. Another pathway of propagation utilized by Stuxnet is through the infection of computers running Siemen’s WinCC software as its core SCADA system.

When it finds a system running this software it connects to the database server using a password that is hardcoded within the WinCC software. Once it has connected it performs two actions. First, Stuxnet sends malicious SQL code to the database that allows a version of Stuxnet to be transferred to the computer running the WinCC software and executes it, thereby infecting the computer that is running the WinCC database. Second, Stuxnet modifies an existing view adding code that is executed each time the view is accessed.¹⁰⁸

Not only does Stuxnet infect WinCC software, it also copies itself into STEP 7 project files and auto-executes each time these files are loaded. Considering the use of a hardcoded password from WinCC software, it could be inferred that someone with insider knowledge of Siemen’s PCS 7 systems was involved in Stuxnet’s design. Stuxnet also has the ability to spread through networks via shared resources.

¹⁰⁸ Ibid., 26.

Stuxnet can spread to available network shares through either a scheduled job or using Windows Management Instrumentation (WMI). Stuxnet will enumerate all user accounts of the computer and the domain, and try all available network resources either using the user's credential token or using WMI operations with the explorer.exe token in order to copy itself and execute on the remote share.¹⁰⁹

Stuxnet's final method of propagation is by way of Windows Server Service vulnerability MS08-067. This is the same vulnerability exploited by the Conficker worm in 2008. Being that a patch exists, this is not a zero-day. Regardless, any system that has not been patched remains susceptible.

MS08-067 can be exploited by connecting over Server Message Block (SMB) and sending a malformed path string that allows arbitrary execution. Stuxnet uses this vulnerability to copy itself to un-patched remote computers.¹¹⁰

Having examined Stuxnet's exploits and vectors for propagation, Stuxnet's architecture and method of installation will now be discussed. Stuxnet's architecture is complex, and somewhat difficult to describe in layman's terms. Symantec Security Response provides a comprehensive overview of Stuxnet's organization in their Win32.Stuxnet Dossier.

The heart of Stuxnet consists of a large .dll file that contains many different exports and resources. In addition to the large .dll file, Stuxnet also contains two encrypted configuration blocks. The dropper component of Stuxnet is a wrapper program that contains all of the above components stored inside itself in a section name "stub". This stub section is integral to the working of Stuxnet. When the threat is executed, the wrapper extracts the .dll file from the stub section, maps it into memory as a module, and calls one of the exports. A pointer to the original stub section is passed to this export as a parameter. This export in turn will extract the .dll file from the stub section, which was passed as a parameter, map it into memory and call another different export from inside the mapped .dll file. The pointer to the original stub section is again passed as a parameter. This occurs continuously throughout the execution of the threat. In this way every layer of the threat always has access to the main .dll and the configuration blocks. In addition to loading the .dll file into memory and calling an export directly, Stuxnet also uses another technique to call exports from the main .dll file. This technique is to read an executable template from its own resources, populate the template with appropriate data, such as which .dll file to load and which export to call, and then to inject this newly populated executable into another process and execute it.¹¹¹

As Symantec states, the main payload of Stuxnet is embodied in a DLL file which is constantly being re-circulated by a variety of exports. These exports represent a number of

¹⁰⁹ Ibid., 27.

¹¹⁰ Ibid., 28.

¹¹¹ Ibid., 12.

functions related to Stuxnet’s installation, propagation, and updating mechanisms. They are enumerated according to Table 1:

Table 1. List of Exports

DLL Exports	
Export #	Function
1	Infect connected removable drives, starts RPC server
2	Hooks APIs for Step 7 project file infections
4	Calls the removal routine (export 18)
5	Verifies if the threat is installed correctly
6	Verifies version information
7	Calls Export 6
9	Updates itself from infected Step 7 projects
10	Updates itself from infected Step 7 projects
14	Step 7 project file infection routine
15	Initial entry point
16	Main installation
17	Replaces Step 7 DLL
18	Uninstalls Stuxnet
19	Infects removable drives
22	Network propagation routines
24	Check Internet connection
27	RPC Server
28	Command and control routine
29	Command and control routine
31	Updates itself from infected Step 7 projects
32	Same as 1

Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. “W32.Stuxnet Dossier.” Symantec Security Response (February 2011).
http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

The other components of the DLL file are its resources. These are files that “the exports use in the course of controlling the worm.”¹¹² Symantec enumerates these resources according to Table 2:

¹¹² Ibid., 13.

Table 2. List of Resources

DLL Resources	
Resource ID	Function
201	MrxNet.sys load driver, signed by Realtek
202	DLL for Step 7 infections
203	CAB file for WinCC infections
205	Data file for Resource 201
207	Autorun version of Stuxnet
208	Step 7 replacement DLL
209	Data file (%windows%\help\winmic.fts)
210	Template PE file used for injection
221	Exploits MS08-067 to spread via SMB.
222	Exploits MS10-061 Print Spooler Vulnerability
231	Internet connection check
240	LNK template file used to build LNK exploit
241	USB Loader DLL -WTR4141.tmp
242	MRxnet.sys rootkit driver
250	Exploits Windows Win32k.sys Local Privilege Escalation (MS10-073)

Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. "W32.Stuxnet Dossier." Symantec Security Response (February 2011). http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

The injection technique employed by Stuxnet is extremely cunning. Stuxnet has the ability to search for many commercially available anti-virus products and inject itself into the trusted processes of these software suites. "When injecting into a trusted process, Stuxnet may keep the injected code in the trusted process or instruct the trusted process to inject the code into another currently running process."¹¹³ In this way, Stuxnet determines whether it can bypass a security product. If it cannot inject into the process of the anti-virus product itself, it will redirect to one of three trusted default Windows processes: Lsass.exe, Winlogon.exe, or Svchost.exe. "In addition, Stuxnet will determine if it needs to use one of the two currently undisclosed (zero-day) privilege escalation vulnerabilities before

¹¹³ Ibid., 14.

injecting.”¹¹⁴ Table 3 shows the security products Stuxnet searches for and the related target processes for injection:

Table 3: Process Injection

Process Injection	
Security Product Installed	Injection target
KAV v1 to v7	LSASS.EXE
KAV v8 to v9	KAV Process
McAfee	Winlogon.exe
AntiVir	Lsass.exe
BitDefender	Lsass.exe
ETrust v5 to v6	Fails to Inject
ETrust (Other)	Lsass.exe
F-Secure	Lsass.exe
Symantec	Lsass.exe
ESET NOD32	Lsass.exe
Trend PC Cillin	Trend Process

Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. “W32.Stuxnet Dossier.” Symantec Security Response (February 2011). http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

When Stuxnet’s payload DLL file initially installs it calls export 15 which completes an assortment of tasks.

It is responsible for checking that the threat is running on a compatible version of Windows, checking whether the computer is already infected or not, elevating the privilege of the current process to system, checking what antivirus products are installed, and what the best process to inject into is. It then injects the .dll file into the chosen process.¹¹⁵

If export 15 determines that the system is properly configured, it will then determine whether the machine is running a compatible version of Windows. If the machine is running Win2K

¹¹⁴ Ibid.

¹¹⁵ Ibid., 16.

Windows XP, Windows 2003, Vista, Windows Server 2008, Windows 7, or Windows Server 2008 R2, export 15 then checks to see if it has administrator privileges. If it does not, it will execute one of the two zero-day privilege escalation attacks, depending on the OS.

If the operating system is Windows Vista, Windows 7, or Windows Server 2008 R2 the currently undisclosed Task Scheduler Escalation of Privilege vulnerability is exploited. If the operating system is Windows XP or Windows 2000 the Windows Win32k.sys Local Privilege Escalation vulnerability (MS10-073) is exploited.¹¹⁶

The actions taken by export 15 are represented in Figure 9:

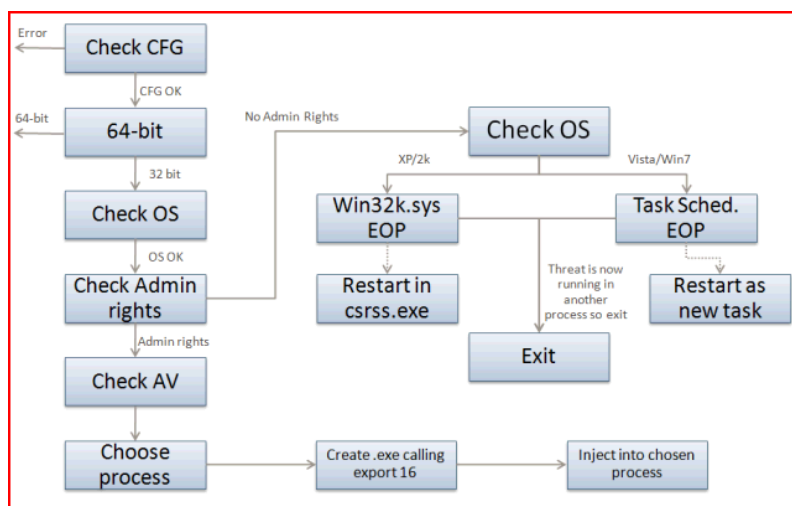


Figure 9. Control flow of export 15. Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. “W32.Stuxnet Dossier.” Symantec Security Response (February 2011). http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

When all the checks performed by export 15 are completed successfully, export 16 is called.

Export 16 is the main installer for Stuxnet. It checks the date and the version number of the compromised computer; decrypts, creates and installs the rootkit files and registry keys; injects itself into the services.exe process to infect removable drives; injects itself into the Step7 process to infect all Step 7 projects; sets up the global mutexes that are used to communicate between different components; and connects to the RPC server.¹¹⁷

¹¹⁶ Ibid., 17.

¹¹⁷ Ibid.

After checking configuration data, export 16 “checks the value ‘NTVDM TRACE’ in the system registry. If this value is equal to 19790509 the threat will exit.”¹¹⁸ Symantec describes this value as a “do not infect marker,”¹¹⁹ but also makes an interesting observation at this point in their analysis, shedding light on a possible motive of Stuxnet’s designers.

As a date, the value may be May 9, 1979. This date could be an arbitrary date, a birth date, or some other significant date. While on May 9, 1979 a variety of historical events occurred, “According to Wikipedia, Habib Elghanian was executed by a firing squad in Tehran sending shock waves through the closely knit Iranian Jewish community. He was the first Jew and one of the first civilians to be executed by the new Islamic government. This prompted the mass exodus of the once 100,000 member strong Jewish community of Iran which continues to this day.”¹²⁰

Symantec makes a statement cautioning “readers on drawing any attribution conclusions, (as) attackers would have the natural desire to implicate another party.”¹²¹ The inclusion of this inference is interesting nonetheless, as it points to Israel as a probable source of Stuxnet.

Figure 10 is a visual depiction of the actions taken by export 16.

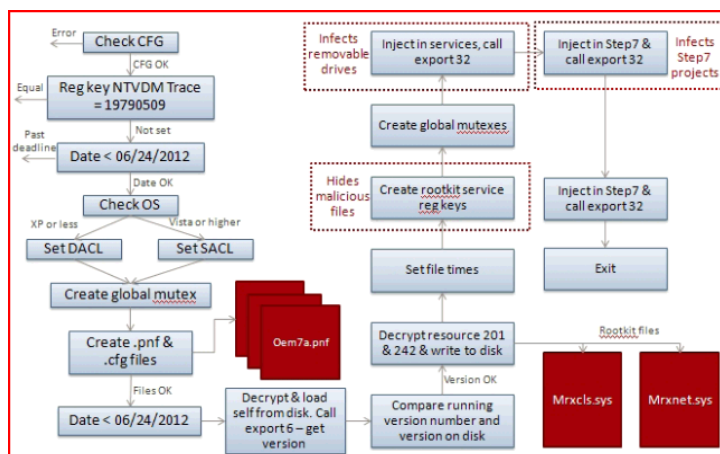


Figure 10. Infection routine flow. Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien.

“W32.Stuxnet Dossier.” Symantec Security Response (February 2011).

http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

¹¹⁸ Ibid.

¹¹⁹ Ibid., 18.

¹²⁰ Ibid.

¹²¹ Ibid.

After checking the registry, Stuxnet checks the date of the configuration file to determine that it is before June 24, 2012, then checks the OS to determine how it will modify access controls and “ensure no write actions are denied. It uses the SetSecurityDescriptorDacl function for computers running Windows XP and SetSecurityDescriptorSacl for computers running Windows Vista.”¹²² Stuxnet then sets up a global mutex (mutual exclusion object). This “allows multiple program threads to share the same resource, such as file access, but not simultaneously.”¹²³ This is how Stuxnet effectively “communicates between different components.”¹²⁴ After this,

Stuxnet creates 3 encrypted files. These files are read from the .stub section of Stuxnet; encrypted and written to disk, the files are:

1. The main Stuxnet payload .dll file is saved as Oem7a.pnf
2. A 90 byte data file copied to %SystemDrive%\inf\mdmeric3.PNF
3. The configuration data for Stuxnet is copied to %SystemDrive%\inf\mdmcpq3.PNF
4. A log file is copied to %SystemDrive%\inf\oem6C.PNF¹²⁵

After these files are created, encrypted and written to disk, the date is verified once again, and Stuxnet calls export 6 to “read the version number from its own configuration data and compare it with the version number from the file on disk.”¹²⁶ Assuming that the two versions match,

Stuxnet will extract, decode, and write two files from the resources section to disk. The files are read from resource 201 and 242 and are written to disk as “Mrxnet.sys” and “Mrxcls.sys” respectively. These are two driver files; one serves as the load point and the other is used to hide malicious files on the compromised computer and to replace the Stuxnet files on the disk if they are removed.¹²⁷

Mrxcls.sys (Resource 242) and Mrxnet.sys (Resource 201) are system driver files that were signed with a compromised digital certificate from Realtek, as described earlier in this chapter. “A different version of the driver was also found signed by a different compromised digital certificate from JMicron.”¹²⁸ As stated, Stuxnet is the first cyber-threat to compromise

¹²² Ibid.

¹²³ Quin St. Inc., “Webopedia,” Quin St. Inc., <http://www.webopedia.com/TERM/M/mutex.html> (accessed June 5, 2012)

¹²⁴ Falliere, O. Murchu, and Chien, “W32.Stuxnet Dossier,” 18.

¹²⁵ Ibid.

¹²⁶ Ibid.

¹²⁷ Ibid.

¹²⁸ Ibid., 20.

two digital certificates. “Mrxcsl.sys is a driver that allows Stuxnet to be executed every time an infected system boots and thus acts as the main load-point for the threat.”¹²⁹ Mrxnet.sys enables Stuxnet’s rootkit functionality, which was discussed earlier in this chapter. In examining this resource file, another inference was made by Symantec in regard to Stuxnet’s creators.

In the driver file, the project path b:\myrtus\src\objfre_w2k_x86\i386\guava.pdb was not removed. Guavas are plants in the myrtle (myrtus) family genus. The string could have no significant meaning; however, a variety of interpretations have been discussed. “According to Wikipedia, Esther was originally named Hadassah. Hadassah means ‘myrtle’ in Hebrew.” Esther learned of a plot to assassinate the king and “told the king of Haman’s plan to massacre all Jews in the Persian Empire...The Jews went on to kill only their would-be executioners.”¹³⁰

Of course, this observation was followed by the same statement of disclaimer as the first reference to Iranian Jews. Indeed it may be somewhat of a reach. Regardless, its inclusion in an otherwise technical analysis of a computer worm is entertaining, and claims of Israel’s motives for involvement are not without basis. After creating more global mutexes,

Stuxnet passes control to two other exports to continue the installation and infection routines. Firstly, it injects the payload .dll file into the services.exe process and calls export 32, which is responsible for infecting newly connected removable drives and for starting the RPC server. Secondly, Stuxnet injects the payload .dll file into the Step7 process S7tgotpx.exe. Export 2 is used to infect all Step7 project files.¹³¹

As stated earlier, one method of Stuxnet’s propagation is through the infection of WinCC/STEP 7 software, which controls the SCADA core system and configures PLCs. Stuxnet accomplishes this by way of export 2.

The main export, Export 16, calls Export 2, which is used to hook specific APIs that are used to open project files inside the s7tgotpx.exe process. This process is the WinCC Simatic manager, used to manage a WinCC/Step7 project.¹³²

Another important element of Stuxnet is its ability to “contact a command and control server on the Internet for instructions and updates.”¹³³

¹²⁹ Ibid.

¹³⁰ Ibid., 24.

¹³¹ Ibid., 18.

¹³² Ibid., 33.

¹³³ Byres, Ginter, and Langill, *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems*, 7.

Stuxnet contacts the command and control server on port 80 and sends some basic information about the compromised computer to the attacker via HTTP. Two command and control servers have been used in known samples: www.mypremierfutbol.com & www.todaysfutbol.com. The two URLs previously pointed to servers in Malaysia and Denmark. This feature gave Stuxnet backdoor functionality, as it had the possibility (before the *futbol* domains were blocked) to upload and run any code on an infected machine.¹³⁴

Stuxnet's command and control functionality is represented in Figure 11.

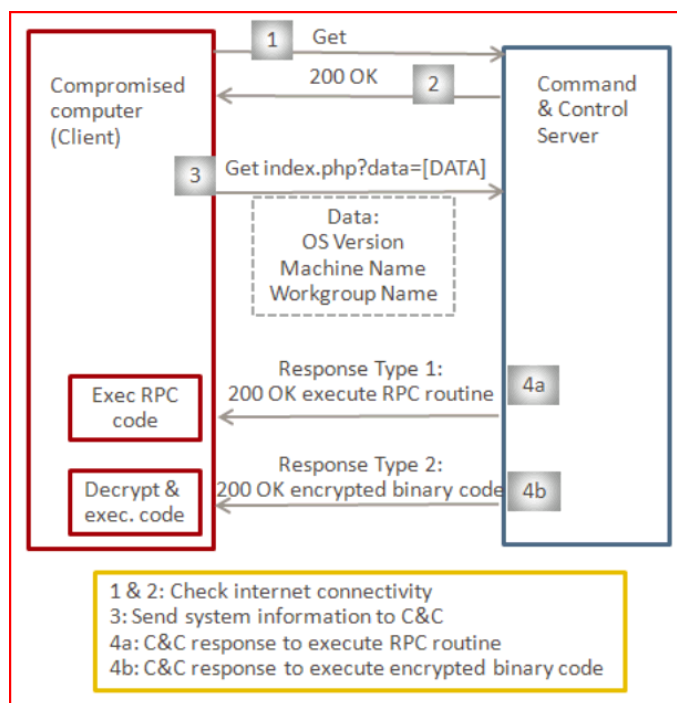


Figure 11. Command and control. Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. "W32.Stuxnet Dossier." Symantec Security Response (February 2011). http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).

At this point, it is appropriate to reiterate the purpose of the massive coding effort embodied by Stuxnet. As stated earlier, Stuxnet was approximately 1 megabyte in size, making it one of the most complex computer worms to date. For perspective, SQL Slammer, the smallest computer worm, was only 376 bytes long. Stuxnet's code was designed to

¹³⁴ Falliere, O. Murchu, and Chien, "W32.Stuxnet Dossier," 22.

effectively manipulate the PLCs of the Siemens' PCS 7 SCADA system while simultaneously concealing its presence and propagation. After examining the internal workings of Stuxnet itself, it becomes relevant to address the threat in terms of how it physically affects the industrial control systems it infects.

When first installed on a computer with any STEP 7 software installed, Stuxnet attempts to locate Siemens STEP 7 programming stations and infect these. If it succeeds, it replaces the STEP 7 routines on the programming stations, so that any person viewing a PLC's logic would not see any changes Stuxnet later makes to the PLC. These actions occur on all computers with STEP 7 software installed, irrespective of whether the compromised computers are connected to PLCs. Stuxnet then looks for specific models of Siemens PLCs. If it is able to connect to one of these two models, it "fingerprints" the PLC by checking for the existence of certain process configurations and strings in the PLC. If Stuxnet finds what it is looking for in the PLC, it starts one of three sequences to inject different STEP 7 code "payloads" into the PLC. The PLC's PROFIBUS driver is replaced and the main PLC program block and the primary watchdog block are significantly modified. Two of Stuxnet's injected payloads are designed to change the output frequencies of specific Variable Frequency Drives (VFDs) and thus the speed of the motors connected to them, essentially sabotaging an industrial process. A third payload appears to be designed to control the overall safety system. This payload takes the inputs coming from the PLC's I/O modules and modifies them so that the PLC safety logic uses incorrect information. The Stuxnet logic then tells the PLC's outputs to do what it wants. This is possibly to prevent a safety system from alarming on or overriding the changes the worm is making to the VFD operations.

It was obvious that Stuxnet was built for purposes of industrial sabotage, but uncovering its target required further analysis. Certain facts were known. Analysts knew that Stuxnet was a directed attack, as it actively searched for PLCs with specific configurations. They knew that Stuxnet required "the industrial control system to have frequency converter drives from at least one of two specific vendors, one headquartered in Finland and the other in Tehran, Iran"¹³⁵ that operate at extremely high speeds, "between 807 Hz and 1210 Hz."¹³⁶ It was also apparent that Stuxnet changed these output frequencies at a very slow rate over intervals of months. Analysts also knew that the majority of Stuxnet infections occurred within Iran. Keeping these facts in mind, German Security Specialist Ralph Langner was able to determine Stuxnet's target was the Natanz uranium enrichment facility. Langner describes this process in a 2011 lecture:

¹³⁵ Eric Chien, "Stuxnet: A Breakthrough," Symantec Official Blog, entry posted November 16, 2010, <http://www.symantec.com/connect/blogs/stuxnet-breakthrough> (accessed June 7, 2012).

¹³⁶ Ibid.

We extracted and decompiled the attack code, and we discovered that it's structured in two digital bombs -- a smaller one and a bigger one. And we also saw that they are very professionally engineered by people who obviously had all insider information. We were looking for timers and data structures and trying to relate them to the real world -- to potential real world targets. In order to get target theories, we remember that it's definitely hardcore sabotage, it must be a high-value target and it is most likely located in Iran. Now you don't find several thousand targets in that area. It basically boils down to the Bushehr nuclear power plant and to the Natanz fuel enrichment plant. So I told my assistant, "Get me a list of all centrifuge and power plant experts from our client base." And I phoned them up and picked their brain in an effort to match their expertise with what we found in code and data. We were able to associate the small digital warhead with the rotor control. And if you manipulate the speed of this rotor, you are actually able to crack the rotor and eventually even have the centrifuge explode. What we also saw is that the goal of the attack was really to do it slowly and creepy -- obviously in an effort to drive maintenance engineers crazy, that they would not be able to figure this out quickly. The big digital warhead -- I started to research scientific literature on how these centrifuges are actually built in Natanz and found they are structured in what is called a cascade, and each cascade holds 164 centrifuges. These centrifuges in Iran are subdivided into 15, what is called, stages. And guess what we found in the attack code? An almost identical structure. We figured out that both digital warheads were actually aiming at one and the same target, but from different angles. The small warhead is taking one cascade, and spinning up the rotors and slowing them down, and the big warhead is talking to six cascades and manipulating valves. So in all, we are very confident that we have actually determined what the target is. It is Natanz, and it is only Natanz.¹³⁷

Clearly, the Natanz facility was a high security site. For obvious reasons, the network architecture of the actual facility is not publicly available. However, it is fair to assume that at minimum, the Natanz facility made use of Siemens' recommended protocols (as described in the previous chapter). Assuming these protocols were implemented, it is probable that Stuxnet's course of infection began with the introduction of an infected USB stick into the site's Corporate Network. The infected USB stick could potentially have originated from a number of sources. It could have been acquired from an employee of an off-site contractor, or possibly from an industry trade show. It is however, important to keep in mind that an infected USB drive is not the only possible cause for Stuxnet's initial introduction to the system. Regardless, after the USB stick's introduction, the worm could have traveled through the Perimeter Network via the corporate WAN, into the Process Control and Control System Networks, (bypassing multiple firewalled zones), and eventually made its way to the end-point of the ICS's PLCs. Figure 12 is a visual representation created by analysts at

¹³⁷ Ralph Langner, "Cracking Stuxnet, a 21st-Century Cyber Weapon," TEDTalks. (March 2011), <http://dotsub.com/view/919a6aa7-b5f0-4583-aba7-12e082a39b1c/viewTranscript/eng> (accessed June 8, 2012).

Tofino Security that depicts the pathway Stuxnet would likely have taken through a high security site such as the Iran's uranium enrichment facility at Natanz.

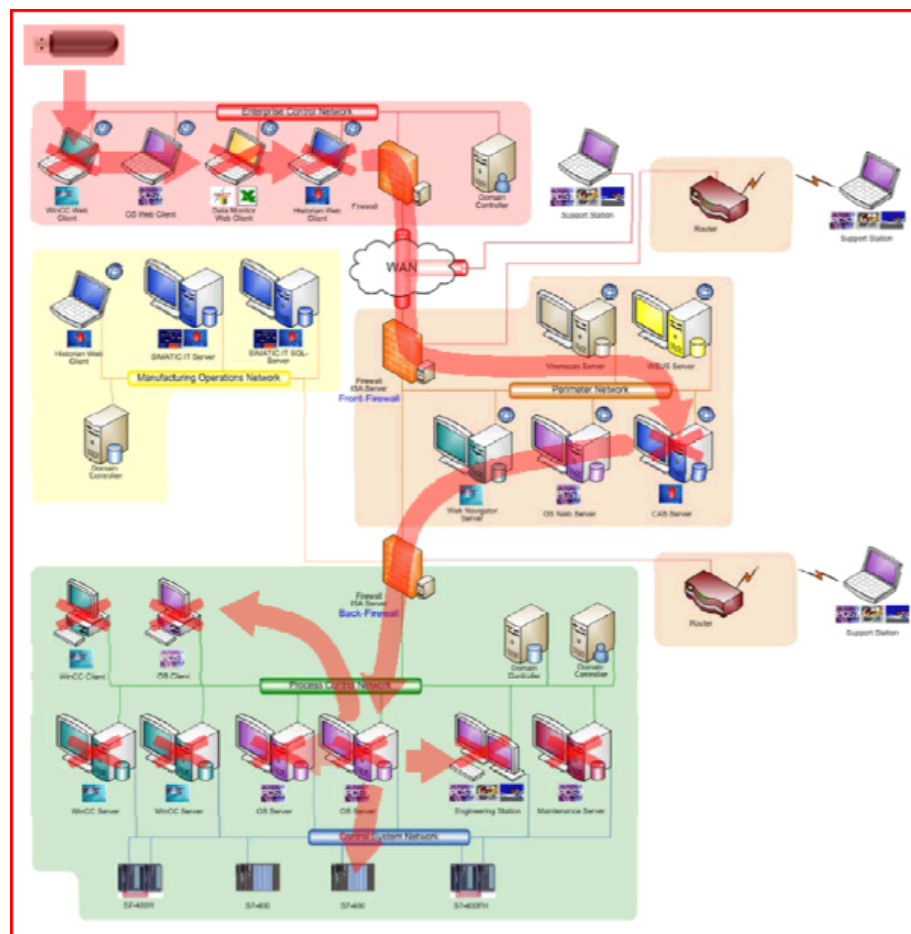


Figure 12. Compromising the site's network. Source: Byres, Eric P., Andrew Ginter, and Joel Langill. *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems*. Alberta: Tofino Security/Abterra Technologies, 2011. <http://abterra.ca/papers/How-Stuxnet-Spreads.pdf> (accessed June 3, 2012).

In reality, there are many alternative pathways through which infection may have occurred. An employee could have used a laptop on an infected outside network and subsequently infected the Natanz system, or a contractor may have visited the site, bringing a compromised external drive or laptop onto the site's network, or an employee at another facility could have used a file share at Natanz over the WAN and infected the network in this way. As prior cyber-attacks have demonstrated, often times security patching is overlooked.

If the Perimeter Network was not adequately patched, infection could have occurred via access by VPN.

Any VPN connection from a compromised host on the Enterprise Control Network to a host on the Perimeter Network using common Windows RPC communications is at risk. Specifically any host on the Perimeter Network with no patch for the 2008 MS08-067 vulnerability would allow the worm to compromise the Perimeter Network.¹³⁸

“The worm could have also been sent to the organization through a targeted email that contained a special dropper program designed to install Stuxnet.”¹³⁹ For example, an attached PDF file could easily have contained such a dropper.

The initial infection of a computer on the target company network could also occur by the contractor supplying PLC project files that are infected. Due to the nature of contractor/client relationships and the need for continuous collaboration, a variety of project files are freely exchanged between team members. These files not only include the PCS 7 project files that the Stuxnet worm could piggy back on, but also other potentially vulnerable file formats including drawing, spreadsheet, database and PDF files that future worms could exploit. It is unlikely that the transfer of these files can be completely prevented, since many are essential to the engineering design process.¹⁴⁰

Stuxnet’s successful penetration of the Natanz site’s networks was wholly due to its versatility in terms of infection paths. Its creators planned for multiple contingencies, and clearly had insider knowledge of the system’s design. See Appendix B for a partial attack graph displaying the multiple ingresses Stuxnet may have used to spread through the security zones of the Natanz facility’s ICS.

The sheer complexity of Stuxnet’s design evidences the fact that its creators could not simply have been overzealous graduate students, or a loosely knit hacking collective. It remains clear that Stuxnet was a massive, expertly organized coding effort conducted by professionals at the top of the industry, armed with insider information largely unavailable to the public. According to Ralph Langner, Stuxnet’s creators “knew all the bits and bytes that they had to attack. They probably even know the shoe size of the operator. So they know everything.”¹⁴¹ Langner was correct in this assertion, as Stuxnet’s creators were supported by

¹³⁸ Byres, Ginter, and Langill, *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems*, 19.

¹³⁹ Ibid.

¹⁴⁰ Ibid.

¹⁴¹ Langner, “Cracking Stuxnet, a 21st-Century Cyber Weapon.”

those whose business is to “know everything,” the United States Central Intelligence Agency, who had been actively seeking to sabotage Iran’s uranium enrichment program for years. “The C.I.A. had introduced faulty parts and designs into Iran’s systems, even tinkering with imported power supplies so that they would blow up, but the sabotage had relatively little effect.”¹⁴² The CIA’s involvement with Stuxnet was, of course, highly classified until a recent leak in the Department of Justice provided David E. Sanger, of the *New York Times* with details regarding the entire operation, which was initiated by former President Bush, and continued by President Obama under the code name “Olympic Games.”

From his first months in office, President Obama secretly ordered increasingly sophisticated attacks on the computer systems that run Iran’s main nuclear enrichment facilities, significantly expanding America’s first sustained use of cyberweapons, according to participants in the program. Mr. Obama decided to accelerate the attacks even after an element of the program accidentally became public in the summer of 2010 because of a programming error that allowed it to escape Iran’s Natanz plant and sent it around the world on the Internet. Computer security experts who began studying the worm, which had been developed by the United States and Israel, gave it a name: Stuxnet.¹⁴³

According to Sanger’s article “Obama Order Sped Up Wave of Cyberattacks Against Iran,” the President wrestled with the decision to allow the attacks to continue despite Stuxnet’s release into cyberspace. The article describes the deployment of Stuxnet’s three variants which occurred in June 2009, and March and April of 2010. Visual representations of the clusters of infection of these three variants are provided in Appendix C. The concentrations and distribution of the variants are represented in Appendix D.

In the following weeks, the Natanz plant was hit by a newer version of the computer worm, and then another after that. The last of that series of attacks, a few weeks after Stuxnet was detected around the world, temporarily took out nearly 1,000 of the 5,000 centrifuges Iran had spinning at the time to purify uranium. This account of the American and Israeli effort to undermine the Iranian nuclear program is based on interviews over the past 18 months with current and former American, European and Israeli officials involved in the program, as well as a range of outside experts¹⁴⁴

¹⁴² Sanger, David E. “Obama Order Sped Up Wave of Cyberattacks Against Iran,” *New York Times*, June 1, 2012. http://www.nytimes.com/2012/06/01/world/middleeast/obama-ordered-wave-of-cyberattacks-against-iran.html?_r=4&pagewanted=1 (accessed June 15, 2012)

¹⁴³ Ibid.

¹⁴⁴ Ibid., 2.

An important element of President Obama's reasoning process was the fact that Israel was reaching a point of critical mass in terms of a potential military intervention in Iran. Obama feared that allowing this to occur would start "a conflict that could spread throughout the region."¹⁴⁵ In order to deter conventional Israeli strikes, the National Security Agency (NSA) worked in tandem with Israeli intelligence.

The unusually tight collaboration with Israel was driven by two imperatives. Israel's Unit 8200, a part of its military, had technical expertise that rivaled the N.S.A.'s, and the Israelis had deep intelligence about operations at Natanz that would be vital to making the cyber-attack a success. But American officials had another interest, to dissuade the Israelis from carrying out their own pre-emptive strike against the Iranian nuclear facilities. To do that, the Israelis would have to be convinced that the new line of attack was working. The only way to convince them, several officials said in interviews, was to have them deeply involved in every aspect of the program.¹⁴⁶

The first step was to develop a "beacon" that was sent into the Natanz facility to conduct reconnaissance. Eventually the beacon program reported back to the NSA with "maps of the electronic directories of the controllers and what amounted to blueprints of how they were connected to the centrifuges."¹⁴⁷ Armed with this information, Stuxnet's creators were able to develop the worm's code. After a prototype was created, a massive testing effort was initiated. This involved building functional models of Iran's P-1 centrifuges, which were a somewhat antiquated design purchased on the black market from notorious Pakistani nuclear profiteer, AQ Khan. As luck would have it, the US government had appropriated a number of these centrifuges when Libyan dictator Muammar Qaddafi abandoned his own nuclear aspirations.¹⁴⁸

When Colonel Qaddafi gave up his nuclear weapons program in 2003, he turned over the centrifuges he had bought from the Pakistani nuclear ring, and they were placed in storage at a weapons laboratory in Tennessee. The military and intelligence officials overseeing Olympic Games borrowed some for what they termed "destructive testing," essentially building a virtual replica of Natanz, but spreading the test over several of the Energy Department's national laboratories to keep even the most trusted nuclear workers from figuring out what was afoot.¹⁴⁹

¹⁴⁵ Ibid.

¹⁴⁶ Ibid.

¹⁴⁷ Ibid.

¹⁴⁸ Ibid., 3.

¹⁴⁹ Ibid.

After months of testing, the worm was declared ready for deployment. According to Sanger, Stuxnet's initial entry point to Natanz was, in fact, via an infected USB stick. To accomplish this, Stuxnet's architects relied on "engineers, maintenance workers and others, both spies and unwitting accomplices, with physical access to the plant."¹⁵⁰ Once the initial infection had occurred, Stuxnet's ability to self-replicate and automatically update via RPC aided in delivering new variants of the worm. Eventually, Stuxnet's effects manifested, as centrifuges began spinning out of control in Natanz. As per Stuxnet's creators' intent, the Iranians had no idea what the cause of the faulty centrifuges was.

The code would lurk inside the plant for weeks, recording normal operations; when it attacked, it sent signals to the Natanz control room indicating that everything downstairs was operating normally. "This may have been the most brilliant part of the code," one American official said. Later, word circulated through the IAEA that the Iranians had grown so distrustful of their own instruments that they had assigned people to sit in the plant and radio back what they saw. When a few centrifuges failed, the Iranians would close down whole "stands" that linked 164 machines, looking for signs of sabotage in all of them. "They overreacted," one official said. "We soon discovered they fired people."¹⁵¹

The first two variants of Stuxnet were responsible for intermittent failures, but it wasn't until 2010, after a third variant of the worm was sent into Natanz, that the wholesale destruction of 1,000 centrifuges occurred. For visual representations of Stuxnet's variants and their spread see appendices B and C. Despite the apparent success of the Olympic Games, Stuxnet's deployment brought with it serious implications. Eventually, intelligence surfaced that Stuxnet had escaped the Natanz facility by way of an engineer's laptop.

An error in the code, they said, had led it to spread to an engineer's computer when it was hooked up to the centrifuges. When the engineer left Natanz and connected the computer to the Internet, the American and Israeli-made bug failed to recognize that its environment had changed. It began replicating itself all around the world. Suddenly, the code was exposed.¹⁵²

The exposure of the Stuxnet worm into cyberspace was clearly unintentional. It was due to this mistake that it was eventually detected by VirusBlokAda and anti-virus signatures and security patches were developed to mitigate future infections. If not for this error, Stuxnet might still be silently lurking in Natanz, sabotaging centrifuges to this day.

¹⁵⁰ Ibid., 4.

¹⁵¹ Ibid.

¹⁵² Ibid., 5.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

In truth, the implications of Stuxnet's release into cyberspace remain to be seen. Since Stuxnet's discovery, security firms have developed detection signatures for most commercially available anti-virus products to protect against future infections. Microsoft has also created patches for the vulnerabilities that Stuxnet exploits, and Verisign has revoked the worm's compromised digital certificate signatures. These facts, as well as the directed nature of the worm stand to reason that, at present, the world has little to fear from Stuxnet's current incarnation. However, the availability of the source code for such a sophisticated cyber-weapon may potentially serve to inspire future computer-based attacks. While decentralized groups such as hacktivist collectives and would-be cyber-terrorists currently lack the resources to develop and implement such an attack, it is foreseeable that nation states such as Russia, China, and possibly even Iran might benefit from examining Stuxnet in regard to developing their own IW capacities. Recently, US intelligence officials argued that Iran used its developing IW arsenal to attack Saudi Arabia's state owned oil company, Aramco, erasing important data from over half of the company's computers in an act of retaliation towards the US. The likelihood of similar attacks will undoubtedly increase in years to come, as there are clear advantages to the offensive use of cyber-weapons over conventional military intervention.

Indeed, computer security analysts have recently reported that Stuxnet's creators are likely conducting reconnaissance for future attacks. This assertion is evidenced by the recent surfacing of the W32.Duqu worm. Duqu's emergence was first reported in 2011 by Hungarian Internet security firm CrySyS. It was given its name because it creates files with the prefix "DQ." It is nearly identical to Stuxnet structurally, which implies that it was created by the same team of programmers, or at very least, someone with access to Stuxnet's source code (prior to its escaping Natanz in 2010). To date, Duqu has infected six organizations across twelve countries including Iran, India, Sudan, Vietnam, France,

Switzerland, Netherlands, Austria, Hungary, Indonesia, and the Ukraine. The geographic distribution of Duqu is visually represented in Figure 13.



Figure 13. Geographic Distribution of W32.Duqu. Source: Falliere, Nicolas, Liam O. Murchu, and Eric Chien. “W32.Duqu The Precursor to the Next Stuxnet.” Symantec Security Response (November 23, 2011). http://scadahacker.com/files/duqu/w32_duqu-the-next-precursor-to_the_next_stuxnet_v1.4.pdf (accessed July 1, 2012).

Initial infection with Duqu typically occurs via email. “The attackers used a specifically targeted email with a Microsoft Word document. The Word document contained a currently undisclosed 0-day kernel exploit that was able to install Duqu.”¹⁵³ Like Stuxnet, Duqu’s driver is also signed with a compromised digital signature, this time from C-Media rather than Realtek. Other similarities include Duqu’s ability to inject itself into trusted processes, its identical RPC component, and its use of P2P and C&C. For a detailed examination of Duqu’s method of installation see Appendix E. While its similarities with Stuxnet are striking, Duqu’s main difference is that its motive is not sabotage but rather espionage. As such, the payload Duqu delivers is an infostealer, which is capable of collecting a wide array of data from its target including keystrokes, machine information (OS

¹⁵³ Nicolas Falliere, Liam O. Murchu, and Eric Chien, “W32.Duqu: The Precursor to the Next Stuxnet,” Symantec Security Response (2011), 2. http://scadahacker.com/files/duqu/w32_duqu-the-next-precursor-to_the_next_stuxnet_v1.4.pdf (accessed July 1, 2012).

version, patches, machine name, users, etc) process lists, network information, lists of shared folders, lists of machines on the same network, and screen shots.¹⁵⁴ Symantec describes Duqu as a Remote Access Trojan (RAT), whose main purpose is to report data back to the attacker “for use in a future attack.”¹⁵⁵ Another difference with Stuxnet is Duqu’s inability to self-replicate, although it can be instructed to copy itself to remote machines via network shares. Duqu also has a limited attack window, removing itself from infected machines after 36 days. Table 4 is a comparison of the functions of Stuxnet and Duqu.

Table 4. Comparison of Stuxnet and Duqu

Feature	Duqu	Stuxnet
Composed of multiple modules	Yes	Yes
Rootkit to hide its activities	Yes	Yes
System driver is digitally signed	Yes (C-Media)	Yes (Realtek, JMicron)
System driver decrypts secondary modules in PNF files	Yes	Yes
Decrypted DLLs are directly injected into system processes instead of dropped to disk	Yes	Yes
Date sensitive: functionality is controlled via complex, encrypted configuration file	Yes (36 days)	Yes
Use XOR based encryption for strings	Yes (key: 0xAE1979DD)	Yes (key: 0xAE1979DD)
Referencing 05.09.1979 in configuration file (http://en.wikipedia.org/wiki/Habib_Elghanian)	Yes (0xAE790509)	Yes (0xAE790509)
New update modules via C&C	Yes (keylogger)	Yes
Known Module to control PLC/SCADA systems	No	Yes

Source: Szor, Peter. “Duqu: Threat Research and Analysis.” McAfee Labs (2011). <http://scadahacker.com/files/duqu/duqu-threat-analysis.pdf> (accessed July 1, 2012).

Due to the geographic distribution of infections, as well as the likelihood that Duqu is the work of Stuxnet’s creators, Americans should take solace in the notion that its purpose is likely to support US information warfare operations abroad. However, the proliferation of industrial control systems is widespread in the United States, making domestic control systems cyber-security an important priority for national defense.

¹⁵⁴ Peter Szor, “Duqu: Threat Research and Analysis,” McAfee Labs (2011), 16. <http://scadahacker.com/files/duqu/duqu-threat-analysis.pdf> (accessed July 1, 2012).

¹⁵⁵ Falliere, O. Murchu, and Chien, “W32.Duqu: The Precursor to the Next Stuxnet,” 2.

The primary goals of the field of computer security are to support the concepts of confidentiality, integrity, and availability. Confidentiality refers to the preservation of “authorized restrictions on information access and disclosure, including means for protecting privacy and proprietary information.”¹⁵⁶ Integrity refers to “guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity.”¹⁵⁷ Availability refers to “ensuring timely and reliable access to and use of information.”¹⁵⁸ In support of these concepts, there are a variety of best practice standards that are recommended for securing industrial control systems.

One of the most important standards is the prompt and secure application of vendor supplied security patches. The history of cyber-attacks outlined in the introductory chapter of this work bears testament to the potentially dire consequences of overlooking this simple step. There are however, key considerations that must be taken into account when applying patches to an ICS.

Patches should not be downloaded directly to the control system. Instead, use a staging system that is not connected to the control system network but is connected to the enterprise network, modem, or other connection. All downloads used on the control system network need to be validated by the vendor and checked with utilities such as Message Digest Algorithm (MD5) or Secure Hash Algorithm (SHA). Hashing functions verify that messages and data have not been changed or tampered with in transit.¹⁵⁹

Another standard is the use of double firewalled de-militarized zones to separate business networks from process control networks. This practice was discussed in Chapter 3, during this study’s examination of Siemens’ recommended best practices for PCS 7 systems. Regardless, it is relevant to reiterate the nature and importance of Firewalls at this point.

The firewall is considered the first layer of network defense. Its most obvious role is to act as a castle wall and protect the inside from the harsh outside. Another, and often forgotten, role is to prevent traffic from escaping to the outside. This traffic could be corporate data or viruses and other malware. In both cases, the firewall is the enforcer of

¹⁵⁶ Donald L Evans, Phillip J. Bond, and Arden L. Bement, “Standards for Security Categorization of Federal Information and Information Systems,” National Institute of Standards and Technology (2004), 2. <http://csrc.nist.gov/publications/fips/fips199/FIPS-PUB-199-final.pdf> (accessed July 2, 2012).

¹⁵⁷ Ibid.

¹⁵⁸ Ibid.

¹⁵⁹ US-CERT, “Cyber-Security for Control Systems Engineers & Operators,” US Department of Homeland Security (2010), <https://www.vte.cert.org/VTEWEB/go/csspcbt.aspx> (accessed July 2, 2012).

network security policy. The firewall also provides a very good log of the traffic between the two networks (inside/outside). The logs are usually the first sign that someone has been doing something he or she isn't supposed to. The firewall is often seen as the panacea for cyber security. However, unless due care is taken with deployments, key errors in configuration can lead to critical security issues. One approach to firewall deployment is "white listing," where only specific services are allowed. By setting the default configuration to "all deny," administrators can permit access based on specific needs. The opposite approach is opening up the entire firewall for access and then turning off what is to be blocked. This task of tuning the firewall to accommodate only that which is explicitly allowed can be difficult in large deployments, especially in industrial domains that are dynamic in terms of data communication requirements.¹⁶⁰

Figure 14 is a visual representation of the implementation of double firewalled DMZs.

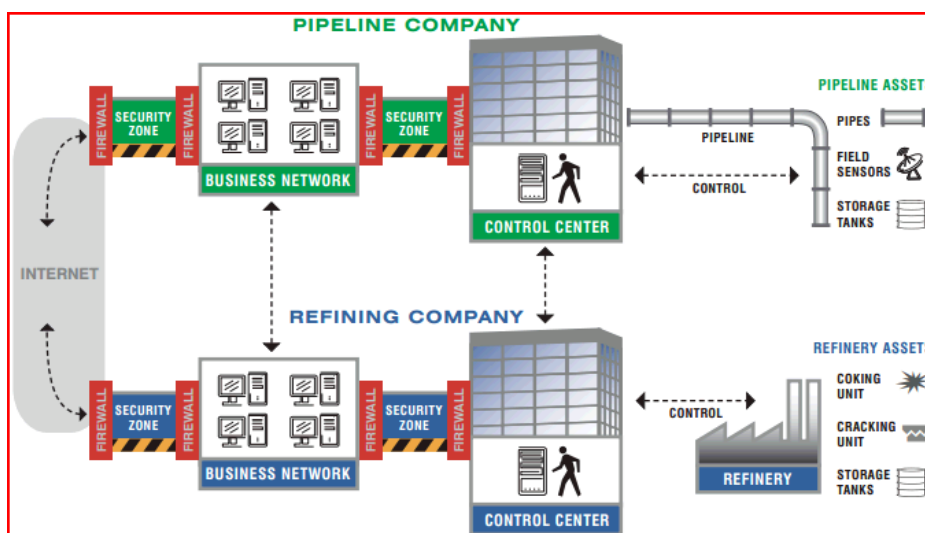


Figure 14. Double Firewalled DMZ. Source: Wybourne, Martin N., Martha F. Austin, and Charles C. Palmer. "National Cyber Security Research and Development Challenges Related to Economics, Physical Infrastructure and Human Behavior." I3P (2009).

<http://www.cyber.st.dhs.gov/docs/i3pnationalcybersecurity.pdf> (accessed July 2, 2012).

Other best practice standards relate to the use of virtual private networks, as sometimes employees are allowed to remotely access the control system from the business network. Providing a dedicated computer for VPN access that utilizes encryption with two-factor authentication can mitigate risk to a degree. Two-factor authentication generally involves confirming your identity by combining something you know, typically a password,

¹⁶⁰ Ibid.

with something you possess, a USB token or even a physical feature, such as a biometric scan of your thumb or iris. Another important heuristic for VPNs is the use of full tunnels.

When allowing VPN from an un-trusted source, such as a home office or vendor site, use only full tunnels. This means that when users are remotely connected into the corporate site, they cannot also interact with systems at their originating site. Full tunnels prevent users' systems from acting as a gateway between the two networks, thereby reducing the risk of malicious traffic such as a worm or virus crossing from their network to yours. Of course, this will not prevent the external system making the VPN connection from infecting your network if it is already infected, as it is now a trusted node on your network.¹⁶¹

Finally, the proper use of Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) are essential to industrial control systems security. These systems are typically configured to detect threats by two methods, signature-based and anomaly-based. Signature-based threat detection involves “comparing the signatures of known threats against observed events to identify possible incidents.”¹⁶² Anomaly-based detection involves “comparing definitions of what activity is considered normal against observed events to identify significant deviations.”¹⁶³ There are also two types of these systems, network-based and host based.

A network-based system monitors network traffic for particular network segments or devices and analyzes the network and application protocol activity to identify suspicious activity. It can identify many different types of events of interest. It is most commonly deployed at a boundary between networks, such as in proximity to border firewalls or routers, VPN servers, remote access servers, and wireless networks... A host based system monitors the characteristics of a single host and the events occurring within that host for suspicious activity. Examples of the types of characteristics a host-based system might monitor are network traffic (only for that host), system logs, running processes, application activity, file access and modification, and system and application configuration changes. Host-based systems are most commonly deployed on critical hosts such as publicly accessible servers and servers containing sensitive information.¹⁶⁴

There are a number of inherent challenges that warrant future research in regard to securing industrial control systems. One important issue is the fact that the majority of control systems operate in real-time, 24 hours a day, 7 days a week, 365 days a year. As

¹⁶¹ Ibid.

¹⁶² Karen Scarfone and Peter Mell, “Guide to Intrusion Detection and Prevention Systems,” National Institute of Standards and Technology (2007), 2-7. <http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf> (accessed July 2, 2012).

¹⁶³ Ibid.

¹⁶⁴ Ibid.

such, responses to inputs are immediate, and must be correct, as “some of the processes they control cannot be restarted or reversed...The situation is further complicated by the difficulty of patching or reconfiguring an uninterruptible system.”¹⁶⁵ Research into fast, efficient, wide-scale methods of authentication for the various components of modern industrial control systems is also essential to overcoming the challenge of enforcing security in a real-time environment.

As the components of a process control system grow in number, type, ownership, capability, and interconnection, the trustworthiness of the entire system depends on the ability of the components to quickly authenticate themselves at system startup or following a local or system-wide disruptive event. Further, both the hardware and software systems should be able to be authenticated, and should be resistant to tampering attacks.¹⁶⁶

Another recommendation for future research is the development of tools for progressively securing ICS software. While tools “have emerged for enterprise systems, few tools have appeared for process control system software development. The dual requirements of integrity and real-time availability bring new challenges in (creating) such tools.”¹⁶⁷

Finally, an important avenue that warrants exploration is the practice of red-teaming, in which an organization purposefully tasks a team of “white-hat” hackers with compromising their own networks in an effort to identify vulnerabilities and deter future attacks. This practice has been employed by corporations as well as the US government. An early example of red-teaming was a 1997 exercise known as Eligible Receiver, in which white-hat hackers under NSA direction compromised government systems in an effort to assess vulnerabilities.

Eligible Receiver is the code name of a 1997 internal exercise initiated by the Department of Defense. A red team of hackers from the National Security Agency (NSA) was organized to infiltrate the Pentagon systems. The red team was only allowed to use publicly available computer equipment and hacking software. Although many details about Eligible Receiver are still classified, it is known that the red team was able to infiltrate and take control of the Pacific command center computers, as well as power

¹⁶⁵ Martin N. Wybourne, Martha F. Austin, and Charles C. Palmer, “National Cyber Security Research and Development Challenges Related to Economics, Physical Infrastructure and Human Behavior,” I3P (2009), 17. <http://www.cyber.st.dhs.gov/docs/i3pnationalcybersecurity.pdf> (accessed July 2, 2012).

¹⁶⁶ Ibid.

¹⁶⁷ Ibid.

grids and 911 systems in nine major U.S. cities.¹⁶⁸

Another government simulation known as Black Ice was conducted in 2001, largely in preparation for the 2002 Olympics in Salt Lake City, Utah. This was not a red-team exercise. However, it is arguably necessary to plan ahead for the potential real-world consequences of a successful cyber-attack on critical infrastructure within US borders. In doing so, organizational and bureaucratic issues can be assessed and improved upon. The simulation entailed a disaster response scenario in which a hypothetical ice storm destroyed power and bulk transmission lines and a subsequent cyber-attack on the SCADA system controlling the power grid was added as a force multiplier. Paula Scalingi, the Director of the Department of Energy's Critical Infrastructure Protection Office at the time, described the importance of the simulation in terms of crisis response.

What was discovered is that if you have a prolonged power outage that goes on for several hours, your infrastructure starts to degrade. Power backup only lasts so long. And it's not just telecommunications. Water systems rely on electric power, as does the natural gas industry and the natural gas-powered electric utilities in the region. Emergency responders struggle through the chaos that results from Internet outages, cell phone overload and telephone failures. The ice storm could easily have been replaced with scenarios of multiple bombs, hijackings or other physical catastrophes.¹⁶⁹

These types of exercises should serve as prototypes for future simulations, as technology is rapidly evolving, and the potential threats we face as nation are continually growing.

Currently, there is no grand theory for the use of cyber-weapons. While the international community has formed diplomatic agreements such as the Nuclear Non-Proliferation Treaty, and the Chemical and Biological Weapons Conventions in regard to the regulation of WMDs, no accords exist governing the use of cyber-weapons. To compound matters, on an asymmetric battlefield, the lines between warfare and covert intelligence activities are often indistinct. As the future will undoubtedly usher in the development of increasingly sophisticated cyber-weapons, perhaps only time will tell how evolving cyber-arsenals will shape the nature of warfare in the information age. Some argue that

¹⁶⁸ *Frontline PBS*, "Cyber War," April 24, 2003
<http://www.pbs.org/wgbh/pages/frontline/shows/cyberwar/warnings/> (accessed June 8, 2012).

¹⁶⁹ Dan Verton, "Utah's Black Ice Scenario," CNN (October 21, 2001),
<http://archives.cnn.com/2001/TECH/ptech/10/21/black.ice.idg/index.html> (accessed July 1, 2012).

international co-operation and regulation of the Internet on a global level are vital to the security of all nations, and therefore, alliances and intelligence sharing are essential.

The best weapon against the online thieves, spies and vandals who threaten global business and security would be international regulation of cyberspace. This is not happening at the moment. People have to realize the Internet is an integral part of every country, politically, socially and business-wise. Not to focus on cyber-security is playing with fire.¹⁷⁰

Creating a framework for such co-operation would be difficult, given the lack of effective domestic legislation and the sensitive nature of classified information. In “International Pathways to Cybersecurity,” Franz Stefan Gady, from the East-West Institute, describes their recent consortium on cyber-security at the European Parliament in February 2010 during the Seventh World-Wide Security Conference. Gady highlights the challenges present in an effort towards international cooperation in cyberspace.

- There is a clear lack of a commonly agreed definition of what cyber-security means. All states treat cyber-security as a domestic issue and hence definitions and legal frameworks vary across nations, which makes international cooperation difficult.
- Breakthrough solutions will require the effective integration of technical, business, legal, defense and international policy competencies on a level that has not happened so far.
- Current diplomatic assets assigned to the problem are inadequate to the task and reflect a lack of political commitment at high levels.
- The commercial drivers for building security into network equipment, networks and services are not adequate. This is the result of a lack of consumer awareness of the risk exposure they face and a lack of leadership and commitment from those in control.
- States have the right to organize offensive and defensive assets for information operations of a strategic character to affect the strategic intentions of other states but international law does not adequately regulate these assets. There needs to be a clear definition what “cyber-peace” means.
- There are three levels of information warfare that need to be regulated: political, military strategic and military tactical. The last (meaning electronic warfare assets oriented to single enemy targets or groupings in localized vicinity) is often overlooked.¹⁷¹

After stating the problem sets, Gady defines nine areas of blockage in international cyber-security discourse. Both education and terminology were identified as major bottlenecks in raising cyber-security awareness.

¹⁷⁰ Franz Stefan Gady, “International Pathways to Cyber Security,” East West Institute (March 30, 2010), <http://www.ewi.info/international-pathways-cybersecurity-0> (Accessed July 2, 2012).

¹⁷¹ Ibid.

- Education and Awareness. Awareness needs to reach “critical mass” in public perception in order for it to become a pragmatic item of private and public sector agendas.
- Terminology. Defining and understanding various descriptions of the issues at hand, whether seen as Cyber-security (U.S.), Information Security (Russia), or Internet Security (China).
- Creation of a sense and system of responsibility. Responsibility needs to be imbedded at three levels (a) individual and corporate end users; (b) creators of technology and media; (c) government.
- Understanding the end user as well as growth of new media and technology.
- Constant battle between security, privacy and freedom. Such matters will not have a one-off solution. Decision makers will need to understand that in order to reach solutions. Some compromises need to be made and balances struck among these three important factors.
- Lack of a legal framework. Lack of domestic legal frameworks will impede international legal cooperation.
- Challenging human nature. By nature we have consistently reacted to threats once they triggered specific actions. The decision-making and reaction mentality needs to change. We must pro-actively address vulnerabilities before they are exercised by threats.
- Dismantle the perception of domestic boundaries. Many treat cyber-security as a domestic issue, failing to understand that cyber-security is a challenge that transcends all borders and requires strong international dialogue, trust and cooperation.
- Economics. While the above aspects are considered, it is important to take into account the economics behind achieving cyber-security cooperation. Who will pay for security? Can incentives be created for corporations and individuals?¹⁷²

Cooperation with our allies is presumably feasible considering the post 9/11 information sharing policies of the United Kingdom and USA. However, until a common terminology is agreed upon, and domestic legal frameworks are established, it will be difficult to implement international legislation.

Some suggested the setting up of a legal framework that will be more comprehensive than current international legislation such as the Council of Europe Convention on Cyber-crime. There is, however, a need to find a consensus agreement on the definition of threats, before switching to global frameworks. There is also a need for education awareness, capacity and trust building.¹⁷³

Domestically, the Comprehensive National Cyber-security Initiative is an important piece of legislation that may serve as a framework for future cyber-security reforms. Introduced by the Bush Administration in 2008, the initiative’s intent is to secure government

¹⁷² Ibid.

¹⁷³ Ibid.

networks from international and domestic attackers and mitigate future threats. While the initiative was a step in the right direction, many argue that it is lacking in content. The Cyber Intelligence Sharing and Protection Act (CISPA) is the latest incarnation of domestic cyber-security legislation. It was recently passed by the House of Representatives in April of 2012. CISPA's stated aim is to "establish procedures to allow intelligence community elements to share cyber threat intelligence with private-sector entities and utilities, and encourage the sharing of such intelligence."¹⁷⁴ Clearly there are advantages to such intelligence sharing in respect to securing industrial control systems, as many essential utilities are owned by private-sector entities. However, opponents of the act argue that it effectively empowers the government to violate personal privacy on a grand scale.

What sparked significant privacy worries is the section of CISPA that says "notwithstanding any other provision of law," companies may share information "with any other entity, including the federal government." By including the word "notwithstanding," House Intelligence Committee Chairman Mike Rogers and ranking member Dutch Ruppersberger intended to make CISPA trump all existing federal and state civil and criminal laws. "Notwithstanding" would trump wiretap laws, Web companies' privacy policies, gun laws, educational record laws, census data, medical records, and other statutes that protect information, warns the ACLU's Richardson: "For cyber-security purposes, all of those entities can turn over that information to the federal government."¹⁷⁵

Creating effective cyber-security legislation is not a simple task. It requires balancing security issues with privacy concerns, and reconciling the gap between the world of policymakers and the IT community. There are many obstacles to surmount in this process. While the US feels its way along this slippery slope, the international community will most certainly be paying close attention. As the tools of international conflict between nations continue to evolve, revolutions within nations are likewise being facilitated by way of information technology. The global proliferation of technologies supported by the Internet has irreversibly changed the diplomatic landscape, in some cases providing a voice for those struggling under oppressive regimes. For example, social networking through sites like

¹⁷⁴ 112th Congress, "H.R.3523. Bill Summary & Status," Congressional Reporting Service (2012), <http://thomas.loc.gov/cgi-bin/bdquery/z?d112:HR03523:@@D&summ2=m&> (accessed July 2, 2012).

¹⁷⁵ Declan McCullagh, "How CISPA Would Affect You," CNET (2012), http://news.cnet.com/8301-31921_3-57422693-281/how-cispa-would-affect-you-faq/ (accessed July 2, 2012).

Facebook and Twitter played a central role in organizing last year's Arab Spring uprisings, as well as the recent "Occupy Wall Street" inspired protests. Additionally, the increased availability of handheld devices has given media outlets and emergency responders real-time accounts of conditions on the ground in areas stricken by conflict or natural disasters. Whether these technological developments will ultimately support peace through democratic ideals and the obsolescence of conventional attrition-based warfare, or rather herald a dystopian future of censorship and digital doom is a question that will likely be answered in decades to come.

REFERENCES

- Andres, Steven. "Cyber Timeline." <http://homelandsecurity.sdsu.edu/690/timeline/> (accessed June 2, 2012).
- Bacevich, Andrew. *American Empire*. Cambridge: Harvard University Press, 2004.
- Barr, Dale. *Technical Information Bulletin 04-1 Supervisory Control and Data Acquisition Systems*. Arlington: National Communications System, 2004.
http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf (accessed June 3, 2012).
- Borland, John. "A Four-Day Dive Into Stuxnet's Heart." *Wired* (2010).
<http://www.wired.com/threatlevel/2010/12/a-four-day-dive-into-stuxnets-heart/> (accessed June 5, 2012).
- Boutin, Paul. "Slammed! An inside view of the worm that crashed the Internet in 15 minutes." *Wired* (2003). <http://www.wired.com/wired/archive/11.07/slammer.html> (accessed May 10, 2012).
- Bright, Peter. "Anonymous speaks: The Inside Story of the HBGary Hack." *Ars Technica* (2011). <http://arstechnica.com/tech-policy/news/2011/02/anonymous-speaks-the-inside-story-of-the-hbgary-hack.ars/3> (accessed May 15, 2012).
- Brito, Jerry and Tate Watkins. *Loving the Cyberbomb?* Fairfax: Mercatus, 2011.
http://mercatus.org/sites/default/files/publication/WP1124_Loving_cyber_bomb.pdf (accessed May 14, 2012).
- Byres, Eric P., Andrew Ginter, and Joel Langill. *How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems*. Alberta: Tofino Security/Abterra Technologies, 2011.
<http://abterra.ca/papers/How-Stuxnet-Spreads.pdf> (accessed June 3, 2012).
- Calderbank, Robert, and Neil J. A. Sloane. "Claude Shannon 1916-2001." *Nature* 410 (2001). <http://www2.research.att.com/~njas/doc/ces5.html> (accessed May 12, 2012).
- Cerf, Vinton G., and Robert E. Kahn. "A Protocol for Packet Network Intercommunication." *IEEE Transaction on Communications* 22 (1974).
<http://ece.ut.ac.ir/Classpages/F84/PrincipleofNetworkDesign/Papers/CK74.pdf> (accessed May 3, 2012).
- Dolak, John C. "Security Essentials: The Code Red Worm." SANS Institute (2001).
http://www.sans.org/reading_room/whitepapers/malicious/code-red-worm_85 (accessed May 10, 2012).
- Evans, Donald L., Phillip J. Bond, and Arden L. Bement. "Standards for Security Categorization of Federal Information and Information Systems." National Institute of Standards and Technology (2004).
<http://csrc.nist.gov/publications/fips/fips199/FIPS-PUB-199-final.pdf> (accessed July 2, 2012)

- Falliere, Nicolas, Liam O. Murchu, and Eric Chien. "W32.Stuxnet Dossier." Symantec Security Response (February, 2011).
http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (accessed June 13, 2012).
- Falliere, Nicolas, Liam O. Murchu, and Eric Chien. "W32.Duqu The Precursor to the Next Stuxnet." Symantec Security Response (November, 2011).
http://scadahacker.com/files/duqu/w32_duqu-the-next-precursor-to_the_next_stuxnet_v1.4.pdf (accessed July 1, 2012).
- Gady, Franz Stefan. "International Pathways to Cyber Security." East West Institute (2010).
<http://www.ewi.info/international-pathways-cybersecurity-0> (accessed July 2, 2012).
- Gimon, Charles A. "Heroes of Cyberspace: Claude Shannon." InfoNation.
<http://www.skypoint.com/members/gimonca/shannon.html> (accessed May 2, 2012).
- Granger, Sarah. "Social Engineering Fundamentals." Symantec Security Response, (November, 2010). <http://www.symantec.com/connect/articles/social-engineering-fundamentals-part-i-hacker-tactics> (accessed May 16, 2012).
- Howes, Eric L. "The Anatomy of a Drive-by-Download." (2004).
<http://www.spywarewarrior.com/uiuc/dbd-anatomy.htm> (accessed May 16, 2012).
- Katzenstein, Peter. *The Culture of National Security: Norms and Identity in World Politics*. New York: Columbia University Press, 1996.
- Kehoe, Brendan P. *Zen and the Art of the Internet*. New Jersey: Prentice Hall, 1992.
- Langner, Ralph. "Cracking Stuxnet, a 21st-Century Cyber Weapon." TEDTalks (March 29 2011). <http://dotsub.com/view/919a6aa7-b5f0-4583-aba7-12e082a39b1c/viewTranscript/eng> (accessed June 8, 2012).
- Lenin, Vladimir. *Imperialism: The Highest Stage of Capitalism*. London: Pluto Press, 1996.
- Lewis, James A. "Thresholds for Cyberwar." Washington DC: Center for Strategic and International Studies (2010). http://csis.org/files/publication/101001_ieee_insert.pdf (accessed May 11, 2012).
- Leyden, John. "Blaster Worm Spreading Rapidly." The Register (August 12, 2003).
http://www.theregister.co.uk/2003/08/12/blaster_worm_spreading_rapidly (accessed May 10, 2012).
- Loy, Jim "The Cuckoo's Egg - by Clifford Stoll." Review of The Cuckoo's Egg, by Clifford Stoll, Jim Loy Books, 1997. <http://www.jimloy.com/books/cuckoo.htm> (accessed May 5, 2012).
- Magnuson, Stew. "Cyber Experts Have Proof That China Has Hijacked U.S.-Based Internet Traffic." National Defense Magazine (November 12, 2010).
<http://www.nationaldefensemagazine.org/blog/Lists/Posts/Post.aspx?ID=249> (accessed May 16, 2012).
- Matrosov, Aleksandr, Eugene Rodionov, David Harley, and Juraj Malcho. "Stuxnet Under the Microscope." ESET (2010). http://go.eset.com/us/resources/whitepapers/Stuxnet_Under_the_Microscope.pdf (accessed June 13, 2012).

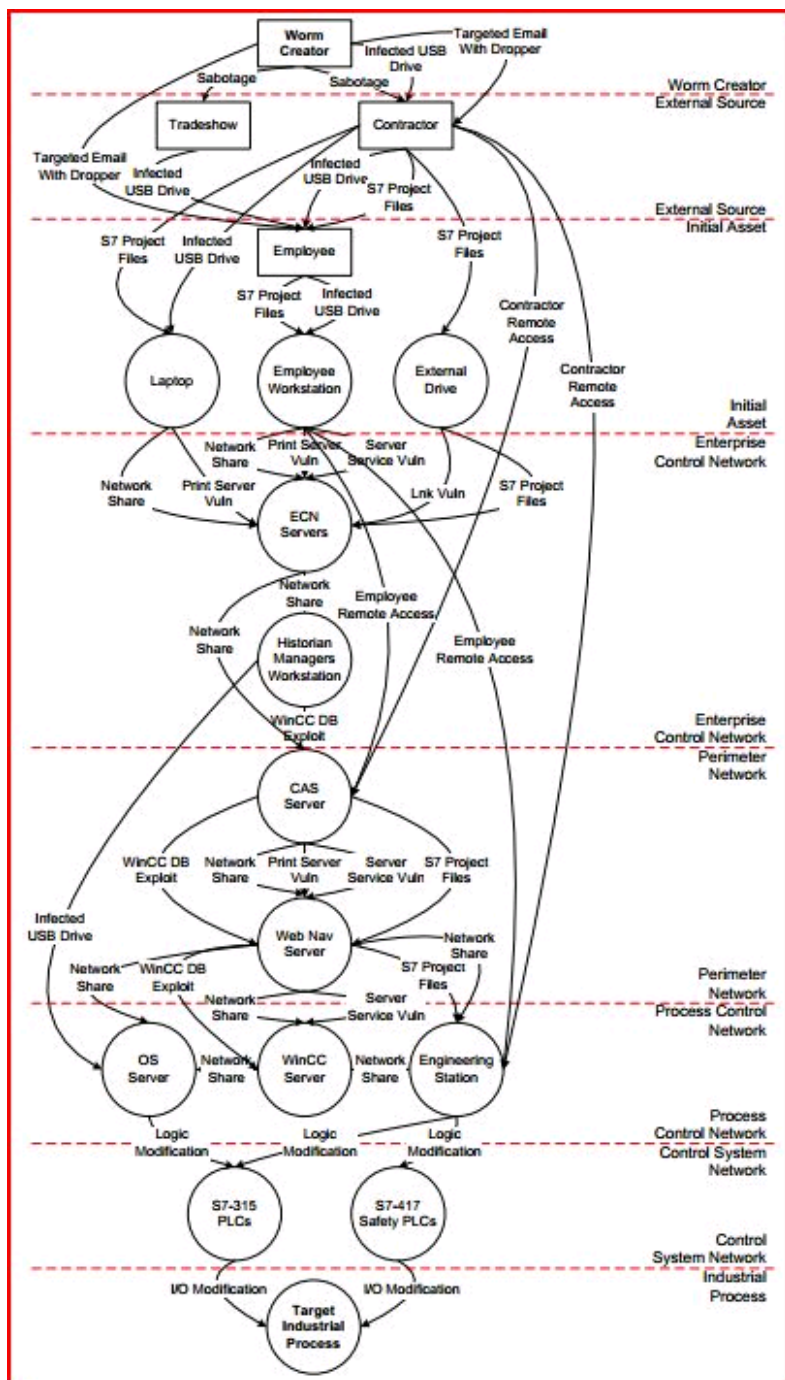
- McCullagh, Declan. "How CISPA Would Affect You." CNET (April 27, 2012).
http://news.cnet.com/8301-31921_3-57422693-281/how-cispa-would-affect-you-faq/
(accessed July 2, 2012).
- Mearsheimer, John J. *The Tragedy of Great Power Politics*. New York: W.W. Norton, 2001.
- Microsoft. "How to Recognize Phishing Email Messages or Links." Microsoft (2011).
<http://www.microsoft.com/security/online-privacy/phishing-symptoms.aspx> (accessed
May 15, 2012).
- Obama, Barack. "Remarks by the President on Securing Our Nation's Cyber Infrastructure."
The White House (March 29, 2009).
http://www.whitehouse.gov/the_press_office/Remarks-by-the-President-on-Securing-Our-Nations-Cyber-Infrastructure/ (accessed May 11, 2012).
- Ogorkiewicz, Maciej, and Piotr Frej. "Analysis of Buffer Overflow Attacks." Windows
Security (2008).
[http://www.windowsecurity.com/articles/Analysis_of_Buffer_Overflow_Attacks.htm](http://www.windowsecurity.com/articles/Analysis_of_Buffer_Overflow_Attacks.html)
l (accessed May 16, 2012).
- PC Mag Encyclopedia. PC Magazine. The Computer Language Company Inc. (2012).
http://www.pcmag.com/encyclopedia_term/0,2542,t%3Dworm&i%3D54874,00.asp
(accessed May 19, 2012).
- Poulson, Kevin. "Slammer Worm Crashed Ohio Nuke Plant Network." Security Focus
(2003). <http://www.securityfocus.com/news/6767> (accessed May 10, 2012).
- Quin St. Inc. "Webopedia." Quin St. Inc. (2012).
<http://www.webopedia.com/TERM/M/mutex.html> (accessed June 5, 2012).
- Russel, Deborah, and G.T. Gangemi, *Computer Security Basics*. Sebastopol: O'Reilly &
Associates, 1994.
- Scarfone, Karen, and Peter Mell. "Guide to Intrusion Detection and Prevention Systems."
National Institute of Standards and Technology (February 2007).
<http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf> (accessed July 2,
2012).
- Schwartau, Winn. *Information Warfare, Chaos on the Electronic Superhighway*. New York:
Thunder's Mouth Press, 1994.
- Sommer, Peter and Ian Brown. *Future Global Shocks: Reducing Systemic Cybersecurity*.
Paris: Organization for Economic Co-operation and Development (January 14, 2011).
<http://www.oecd.org/dataoecd/57/44/46889922.pdf> (accessed May 11, 2012).
- Szor, Peter. "Duqu: Threat Research and Analysis." McAfee Labs (2011).
<http://scadahacker.com/files/duqu/duqu-threat-analysis.pdf> (accessed July 1, 2012).
- "The Hazard of USB Drives." Biz IT Newsletters (October 14, 2010).
<http://www.biznuzz.com/newsletters/?action=article&article=71> (accessed May 19,
2012).

- Thornburgh, Nathan. "Inside the Chinese Hack Attack." *Time* (August 25, 2005). <http://www.time.com/time/nation/article/0,8599,1098371,00.html> (accessed May 18, 2012).
- US Army Cyber Operations and Cyber Terrorism Handbook 1.02*. Army Training and Doctrine Command. August 15 2005. <http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA439217> (accessed May 11, 2012).
- US-CERT. "Cyber-Security for Control Systems Engineers & Operators." US Department of Homeland Security (March 7, 2010). <https://www.vte.cert.org/VTEWEB/go/csspcbt.aspx> (accessed July 2, 2012).
- Vardarajan, Latha. "National Security Policy" (lecture, San Diego State University 2009).
- Verton, Dan. "Utah's Black Ice Scenario." *CNN* (October 21, 2001). <http://archives.cnn.com/2001/TECH/ptech/10/21/black.ice.idg/index.html> (accessed July 1, 2012).
- Washington Post. "Timeline: The U.S. Government and Cybersecurity" *Washington Post* (May 16, 2003). <http://www.washingtonpost.com/wp-dyn/articles/A50606-2002Jun26.html> (accessed May 5, 2012).
- Wilson, Clay. "Botnets, Cybercrime, and Cyberterrorism." Congressional Research Service (January 29, 2008). <http://www.fas.org/sgp/crs/terror/RL32114.pdf> (accessed May 14, 2012).
- Wybourne, Martin N., Martha F. Austin, and Charles C. Palmer. "National Cyber Security Research and Development Challenges Related to Economics, Physical Infrastructure and Human Behavior." I3P (2009). <http://www.cyber.st.dhs.gov/docs/i3pnationalcybersecurity.pdf> (accessed July 2, 2012).
- Zetter, Kim "Revealed: The Internet's Biggest Security Hole" *Wired* (2008). <http://www.wired.com/threatlevel/2008/08/revealed-the-in/> (accessed May 16, 2012).
- 112th Congress. "H.R.3523. Bill Summary & Status." Congressional Reporting Service (April 26, 2012). <http://thomas.loc.gov/cgi-bin/bdquery/z?d112:HR03523:@@D&summ2=m&> (accessed July 2, 2012).

APPENDIX A
STUXNET TIMELINE

W32.Stuxnet Timeline	
Date	Event
November 20, 2008	Trojan.Zlob variant found to be using the LNK vulnerability only later identified in Stuxnet.
April, 2009	Security magazine Hakin9 releases details of a remote code execution vulnerability in the Printer Spooler service. Later identified as MS10-061 .
June, 2009	Earliest Stuxnet sample seen. Does not exploit MS10-046 . Does not have signed driver files.
January 25, 2010	Stuxnet driver signed with a valid certificate belonging to Realtek Semiconductor Corps.
March, 2010	First Stuxnet variant to exploit MS10-046 .
June 17, 2010	Virusblokada reports W32.Stuxnet (named RootkitTmphider). Reports that it's using a vulnerability in the processing of shortcuts/.lnk files in order to propagate (later identified as MS10-046).
July 13, 2010	Symantec adds detection as W32.Temphid (previously detected as Trojan Horse).
July 16, 2010	Microsoft issues Security Advisory for "Vulnerability in Windows Shell Could Allow Remote Code Execution (2286198)" that covers the vulnerability in processing shortcuts/.lnk files. Verisign revokes Realtek Semiconductor Corps certificate.
July 17, 2010	Eset identifies a new Stuxnet driver, this time signed with a certificate from JMicon Technology Corp.
July 19, 2010	Siemens report that they are investigating reports of malware infecting Siemens WinCC SCADA systems. Symantec renames detection to W32.Stuxnet.
July 20, 2010	Symantec monitors the Stuxnet Command and Control traffic.
July 22, 2010	Verisign revokes the JMicon Technology Corps certificate.
August 2, 2010	Microsoft issues MS10-046 , which patches the Windows Shell shortcut vulnerability.
August 6, 2010	Symantec reports how Stuxnet can inject and hide code on a PLC affecting industrial control systems.
September 14, 2010	Microsoft releases MS10-061 to patch the Printer Spooler Vulnerability identified by Symantec in August. Microsoft report two other privilege escalation vulnerabilities identified by Symantec in August.
September 30, 2010	Symantec presents at Virus Bulletin and releases comprehensive analysis of Stuxnet.

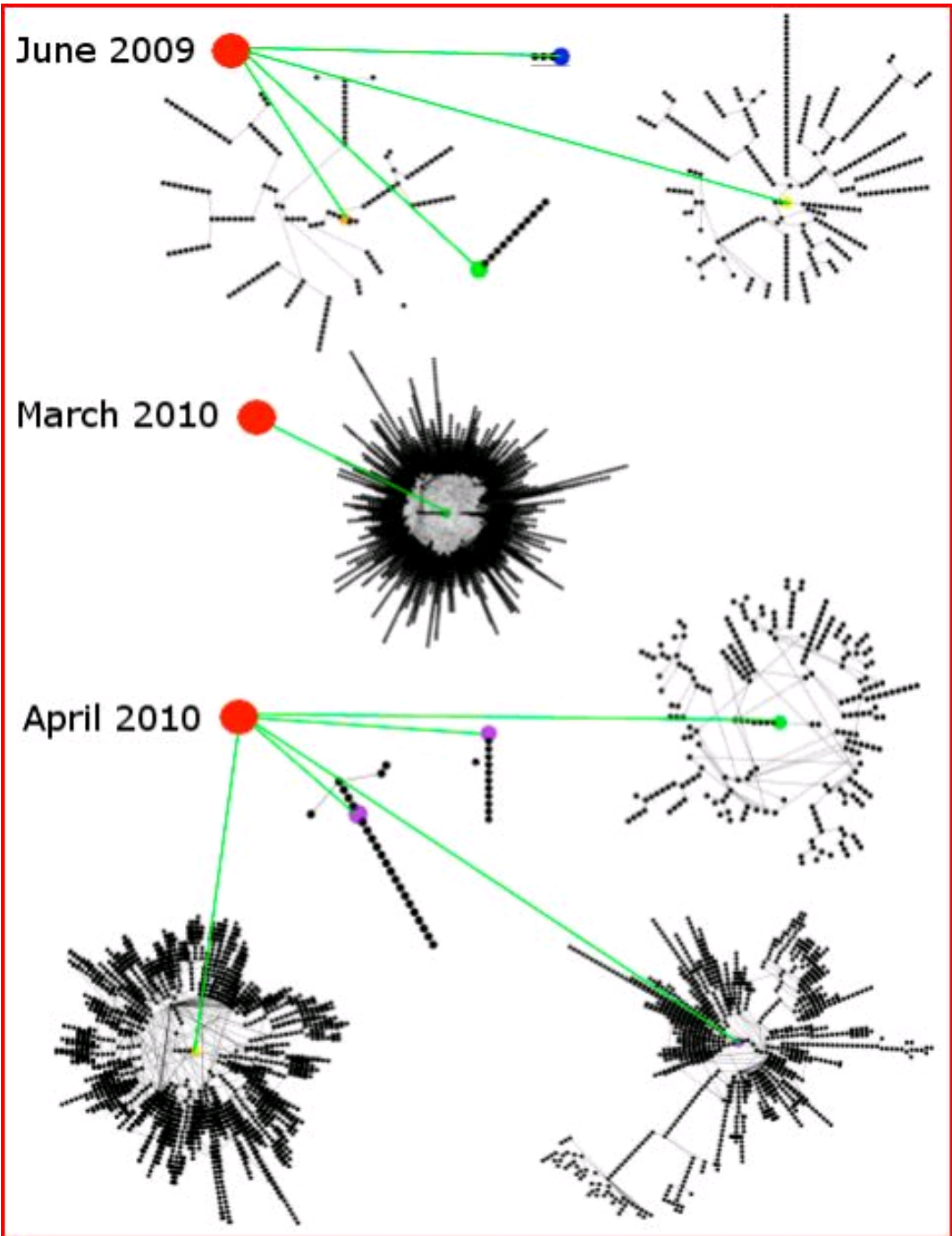
APPENDIX B
PARTIAL ATTACK GRAPH



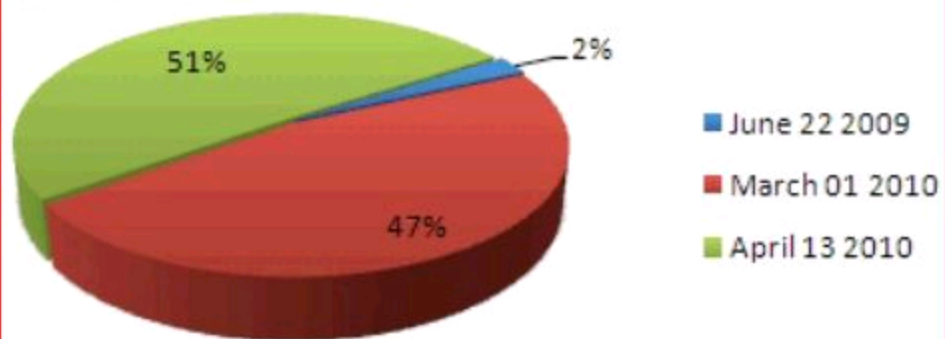
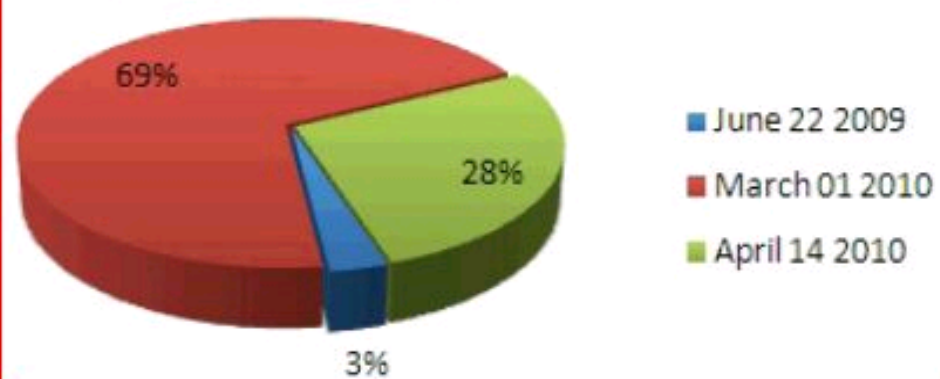
APPENDIX C

CLUSTERS OF INFECTION BASED ON INITIAL

INFECTION



APPENDIX D
STUXNET VARIANTS

Stuxnet Variants**Variant Infection Distribution**

APPENDIX E
DUQU INSTALLATION PROCESS

