Data Fidelity: Security's Soft Underbelly

Dr. Char Sample[1]
Char.sample@icf.com

Steve Hutchinson[1]
steve.hutchinson@icf.com

Prof. Tim Watson[2]
tw@warwick.ac.uk

Bil Hallaq[2]
bh@warwick.ac.uk

Prof. Carsten Maple[2]
cm@warwick.ac.uk

Dr. Jennifer Cowley[1]
jennifer.cowley@icf.com

[1]ICF contractor for US Army Research Laboratory, Adelphi, Maryland
[2]University of Warwick, Coventry, United Kingdom

***Abstract*—*The events of 2016 created a growing concern over the weaponization of information. Weaponized information is actually a symptom of a larger problem, namely, data fidelity. This group of researchers began considering the impact and issues that associate with data fidelity in cyber security.*

Presently, a fundamental universal assumption existing in cyber security solutions is that the entered data being secured is an accurate, faithful representation of the actual events that are occurring in the real world. This assumption of data fidelity is present in every major cyber security product. This work-in-progress paper acknowledges the data fidelity problem, by providing a model that couples the data object with the environment in an attempt to reduce the potential for weaponized information, thereby improving data fidelity.

Keywords—data fidelity; security assumption; data object; data environment, weaponized information.

I. Introduction

The 2016 US presidential election resulted in the emergence of new terms such as "fake news", "post-truth", "weaponized information", "disinformation", and, more recently, "alternative facts". All of these terms comprise the general problem, data fidelity. Although much attention is being directed toward the weaponized information (WI) [1], [2] in the political world [3], deployment of WI in the cyber security domain has not yet been considered; however, this can easily be considered a component of hybrid warfare [4]. The deployment of WI used against security solutions could be catastrophic [5]. Data publishers in the physical world have institutions such as Reuters, API, academia, and other institutions that can ultimately discern the truth of physical events. The virtual environment has no such equivalents. The veracity or fidelity of data created by cyber security systems is assumed to be correct. However, cyber security systems lack effective mechanisms to ensure data fidelity.

In the physical world, events can sometimes be verified through the use of the human's five senses (sight, sound, smell, touch, and taste). The virtual world, where computers are modeled on the human mind [6], has constructs for sight, sound, and in some cases touch, but not the remaining senses. Thus, the ability to examine data for fidelity in the virtual environment is impaired while the opportunity for successfully entering bad data remains. An implicit level of trust is placed in the machines, and although the machines themselves may have reliable security tools, these machines and their tools are presently incapable of inspecting the fidelity of the actual data that is entered. As the case of Stuxnet illustrated, the use of digital certificates gave the appearance that the malware was legitimate [7], even though data fidelity was compromised.

Security specialists can point to various security solutions designed to ensure that entered data was not altered through the use of data integrity solutions [8], [9], [10]; however, these solutions are all based on the assumption that the data put into the security solution faithfully represents reality. In order to understand the fidelity of data, we must first agreed upon several aspects of the data. These items include the definition of the data along with the environment or the situational awareness as these data relate to the data object (DO). The modification history or provenance associated with each of these items requires examination collectively and individually. The researchers believe (1) that coupling the DO with the data environment (DE) creates the context for the data element in which the data were created and (2) that the resultant value will provide the requisite accuracy and exactness necessary to assure a high level of data fidelity.

This research focuses on modeling a method for validating and tracking the fidelity of security data entered into a security system. The model proposed is a work in progress, and, as such, it does not present as a full solution. The authors welcome feedback and suggestions that can contribute to the robustness of this model.

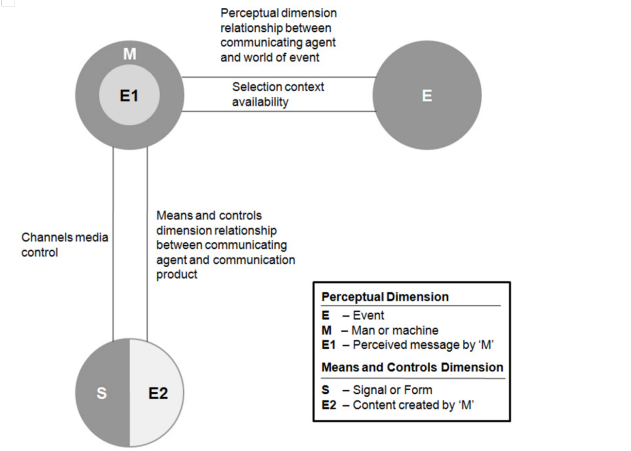
II. Literature Review Weaponized Information

In the physical world, two-person or multi-person control of critical systems that handle critical operations is an example of an early trust model [11]. Because the process now is automated, two-person control on security data is not practical. However, the model requiring the validation and verification by two entities on the same object is relevant.

A case in which perception and reality differed in the virtual environment was witnessed with Stuxnet [12], [13]. Although, Stuxnet is now considered an old attack, one of the key features of this malware was the deceptive capabilities [12], [13] that convinced the operator through normal alert

978-1-5090-5476-3/17/$31.00 ©2017 IEE

processing that the system was operating as expected when in reality the centrifuges were out of control [12]. In other words, the display data was not a faithful or accurate representation of actual events happening in the systems. This was a clear case of disinformation in a security system where perception management was successfully deployed [13].



[a] Taken from: http://communicationtheory.org.

Fig. 1. Gerbner's Model of Communication (reproduced from [14], [15] with permission).

In order for two entities to effectively communicate, both entities must perceive the same message [14], [15]. This perception relies on (1) context or (2) an environment intertwined with the actual object. Fig. 1 illustrates the relationship between object and environment for sending and receiving entities. Gerbner's model defines the relationship between the observer's perception and reporting the perception between the reported data. The perception by the final recipient is not addressed. This important area is recognized as one of the emergent research areas resulting from this work.

A. *Defintions*

Weaponized information in the political sense occurs when malevolent actors subvert or abuse the content or the release time of data to achieve their goals [16] of sowing confusion, creating doubt, paralyzing decision making, demoralizing readers, or blackmailing targets [16]. The application of these attributes of deception in cyber security environments can have similar effects of sowing confusion, creating doubt and paralysis, or deceiving operators to change the security posture based on misinformation.

Szfranski [1] characterized WI as a message content-based attack on knowledge or beliefs. Cybenko et al. [17] refers to these attacks as cognitive hacks (CH). CHs decompose into overt CHs (OCHs) and covert CHs (CCHs). Traditional techniques to defend against OCHs include authentication and data integrity checking technologies such as encryption and digital signatures. Defending against CCHs is more difficult because the goal is to validate the data, not the entity that enters the data [17].

Data has fidelity when it contains a true, accurate, and re-creatable set of components that comprise an information element. The information element minimally contains both the data object and the environment in which the data were created. The resulting objects that result from creating data fidelity are a hashed value of the data object and a hashed value that contains the data object coupled with the environment in which the object was created.

B. *Goals*

Pomerantsev and Weiss [16] enumerated the following goals for weaponized information: confuse, blackmail, demoralize, subvert, and paralyze. These goals serve to blur the line between truth and fiction. Thus, the goal for this data fidelity model is to offer independent modeling to determine the fidelity of input data in highly critical environments. When data appear to fail the fidelity test, a secondary goal will be to provide clarity and offer potential explanations to what has changed about the data.

Before we discuss the importance of data fidelity for technical security solutions, we need to be clear about the ways that data fidelity can be undermined along with the existing data security controls.

Historically speaking, the three fundamental constructs of data security are confidentiality, integrity, and availability [18]. Confidentiality ensures that only the intended viewers are able to view the message content [18], and this is most commonly achieved through encryption and authentication. Data integrity, oftentimes achieved using encryption or digital signature technologies, ensures that the message content remains unchanged once entered [18]. Availability, usually achieved through redundancy, assures the user that the data are ready for use when needed [18]. Non-repudiation, where senders cannot deny sending a message and similarly recipients cannot deny message receipt [18], is oftentimes added as a fourth data security pillar, and data provenance can be used to support non-repudiation.

As a clarifying example, consider a Network Intrusion Detection System (NIDS). Briefly, a NIDS monitors one or more network segments, inspecting frames/packets and matching their contents against a set of rules that specify patterns/signatures of interest and their corresponding alerts, which are sent to network security responders [19]. The NIDS runs on one (or more) computer while interacting with the computer's operating system to receive network data access, the data ruleset, process the rules, and send alerts [19].

An attacker wishing to undermine the NIDS data fidelity has several opportunities, including the following: (1) sending overlapping fragments of data packets that the NIDS reconstructs in a different way from the destination computer, thus presenting harmless data to the NIDS but malicious data to the target computer; or (2) compromising the NIDS host operating system to hide network data from the NIDS or to alter its rulesets, processing, or alerts. The network security responders will in turn find that the data they are reliant on to

Dr. Char Sample, Steve Hutchinson, and Dr. Jennifer Cowley are with ICF at The US Army Research Laboratory 2800 Powder Mill Road, Adelphi, Maryland 20783 (char.sample@icf.com, steve.hutchinson@icf.com and Jennifer.cowley@icf.com. Dr. Sample is also a visiting scholar at the University of Warwick in Coventry, UK.
Professor Tim Watson and Professor Carsten Maple along with Bil Hallaq are with the Cyber Security Centre, University of Warwick, Coventry, UK, (e-

judge the security of the network becomes the equivalent of fake news, thereby influencing them in ways that are advantageous to the attacker.

If we widen our scope to include other security solutions, we can see that firewalls are very similar in function to NIDS, processing incoming network data from its network interfaces and forwarding, dropping, or modifying them according to its rules and internal state [20]. Firewalls share identical issues with data fidelity as NIDS.

Many technical security controls rely on forms of encryption. Although arguably not strictly data, all of these protections rely on a good source of random binary sequences when generating ciphertext, digital signatures, and blocks within a Blockchain or cryptographic hashes. If the randomness is compromised (for example, through various types of entropy attack on an operating system or hypervisor), then the encryption can be catastrophically weakened [21].

The self-protection systems typically included in technical security solutions protect against integrity attacks through the use of remote logging, hashing, and signing of data and encryption and the hardening of processes and filesystems [22]. However, the data they rely on arrives from a supply chain of upstream components (i.e., in the case of NIDS: host operating system, network interface controller, upstream network segments, switches, and routers), any of which might be deleting, inserting, or altering the data in some way.

Network situational awareness has been considered as a method to focus attention on potential problems on the network [23]. These Netflow-based monitoring systems traditionally only examine the header field of the packet, leaving the data field alone. Although these solutions may detect some changes in the size of the payload and the rate of data flow, the best that they offer is an indicator of potential fidelity problems, not evidence or proof.

Application input validation offers some protections such as bounds checking and prevention of buffer overflows, and both techniques are data-centric solutions [24]. However, due to the variety of data ingested, application input validation checks are also vulnerable to data fidelity problems. These solutions are unaware of specific details concerning the data; instead, these solutions provide general data checking that is applicable in a variety of environments [24].

Authenticated sources that enter bad data are described in the Byzantine General's problem [25]. The Byzantine General's problem is believed to be solvable through fault tolerance. Cybenko et al. [17] also invoked the Byzantine General's problem and suggested multiple sources or a fault-tolerant approach to deal with CH. However, another approach mentioned by Cybenko et al. [17] suggested evaluating data within context. Due to the amplification of bad data, this effort favors the use of data within context.

Although the technical solutions discussed thus far provide necessary aspects of data security, they fail to assure data fidelity upon input. Furthermore, the systems are natively incapable of providing the validation and verification of the fidelity of data being input, thus resulting in a large and exploitable gap that will grow as deception techniques continue to gain in popularity [21].

III. Scenarios

There are two versions of this processing model for consideration: a full feature version and a lite version. The full feature version allows for deeper analysis with a full history using a modified technology based on the Blockchain ledger, hereafter referred to as the Blockchain ledger. The lite version relies only on the hashed values of the data and the environment. Also of note, the out-of-band (OOB) path for any object is considered the protected path.

The following abbreviations will assist the reader in the scenario and processing discussion: DO, DE, good (g), bad (b), out-of-band path (o), security information and event management (SIEM) path(s).

When the data are initially entered, the environment is inspected to determine whether it supports the event being recorded. For example, if the data object shows an alert indicating that resources are full, environmental values such as memory usage or process resource usage can easily be checked to determine whether the data are correct. Once verified, the DO is signed and coupled (DO, DE).

The environment includes the source from whence the data came, variables with which the data object interacts (i.e., process size, memory usage for the process, connection information, etc.), and a capture of connection identifying information and other relevant data. The DE, although signed, is not fully trusted until the (DOs, DEs) pair is compared against (DOo, DEo) pair.

There are four general scenarios that we will address for this exercise. We have purposely not included cases where the SIEM system detects bad object or environmental values, because by the time the SIEM detects such problems, too much damage has already occurred. The use of the OOB monitoring system should create scenarios in which the shadow system outputs will not match the SIEM outputs. These mismatched scenarios are the scenarios of interest for this effort. For this discussion, the SIEM system includes all of the components of a SIEM system.

A. Scenario 1: Good data object, good SIEM environment, and OOB match.

In this case, the DO and the DE values are the same when SIEM and OOB compare results. This scenario is used as a control case.

$$(DOs, DEs) = (DOo, DEo)$$

In the new discussion, an example of this would be the report of a building fire, where the camera crews zoom in on the site while firefighters are seen fighting the fire. The story (data object) being reported matches what is sensed in the environment. However, in the virtual environment, when an alert is generated, this signal indicates that a specific source IP address attempted access on a specific targeted port. In addition to the alert being generated, the same source address was accessible, and the command history shows the source host attempting to reach the target address on the specified port. In this particular case, ground truth data are complete.

B. *Scenario 2: Good data object, bad environment, SIEM and OOB mismatch on environment.*

$$((DOs, DEs) <> (DOg, DEo)) \text{ AND } ((DOs = DOo) \text{ AND } (DEs <> DEo))$$

This scenario suggests that data are being used out of context, and data must be sent for further interpretation. In many cases, this environmental perturbation may be in the normal range; therefore, the lite version can be configured to default pass with logging, whereas the full version will associate with more actions.

Re-visiting the example of the fire and the burning building, the scenario depicts fire fighters fighting the blaze; however, in this instance, the background has changed, that is, night replaces day. If we looked strictly at the object, even digitally signed, the object would be considered valid.

In a related security example, this is one of the events that occurred with Stuxnet [12], [13]. Although the alert messages came through normal channels, the environment was most certainly not normal [12], [13]. Some mild environmental perturbations are normal for objects, and, in the full version, these ranges will be in the archive; however, in the lite version, this process will require a manual response.

C. *Scenario 3: Bad data object, good environment, SIEM and OOB mismatched on data object.*

$$((DOs, DEs) <> (DOg, DEo) \text{ AND } (DOs <> DOo) \text{ AND } (DEsg = DEog))$$

This scenario involves a change to the object that does not perturb the environment. In the example using the news story and the building fire, the color of the building may change from red to brown. In that case, the environment would appear undisturbed, but the object's characteristics would be different. In a virtual environment, a false alert could be injected into a good status to divert resources.

D. *Scenario 4: Bad data, bad environment.*

$$((DOs, DEs) <> (DOg, DEo)) \text{ AND } ((DOs <> DOo) \text{ AND } (DEs <> DEo))$$

In this scenario, both the object and the environment do not match. As a result of this discrepancy, an immediate error condition will be generated.

TABLE I. GERBNER'S MODEL PHYSICAL-CYBER EQUIVALENCES

	Model Example	**Cyber Example**
E	Fire, in house, fully involved, no nearby objects.	Anomalous event data on a network, or in a system process
E1	Perception of E	Sensing, sampling of some attributes associated with the event E
M	Observer of the event E	Sensed data match to rule, generating alert
S	Statement message conveyed through an available channel to M2	Log message sent to security event monitor/collector (SIEM)
E2	The perception of E, generated by M1, filtered through channel to M2	The alert message content, timestamp, where the context of E1 are assumed to be the same as the context of E2
M2	Uses the materials available from S|E2, using assumptions about the context of E	Uses the alert (available materials) to enter into a security workflow, to inform analyst(s) about the event (E) described by S|E2, assuming equivalent contexts

IV. PROCESSING

We propose a model solution that is applicable for each of the given scenarios. This proposed solution relies on the use of a modified version of Gebner's model to accommodate recipient perception management, digital signatures, and digital ledger technology. The researchers recognize that key management is a critical component of this solution; however, for this paper, key management is out of scope. Regardless of which version is used, lite or full, some background information is required.

Furthermore, the researchers have determined that the out-of band process that runs as the check against the well-known security solutions should run in a stealthy manner. At minimum, the kernel should be modified [26], allowing the processes to be hidden from view in the system process table. The connections established should also be secure and not visible. The logging capabilities, although able to use the system logging mechanism (a configurable option), will still have a parallel logging process that provides the interface directly to the item being monitored. Thus, the untampered item has been entered through machine interactions and not by human to machine input.

For each scenario, two sets of processing actions will be discussed: the first is for lite processing, and the second is the full version, which uses an archive system that allows for deeper analysis. The archive system is present for the full system and uses Blockchain technology in the archive to create and manage the digital data element ledger. Fig. 2 illustrates the processing associated with the archive.

Each DO is recorded and associated with the estimate of the current observation context. If, at a later time, an observation context (environment) is found to be "bad", then all DOs associated with that context are likely to be bad as well.

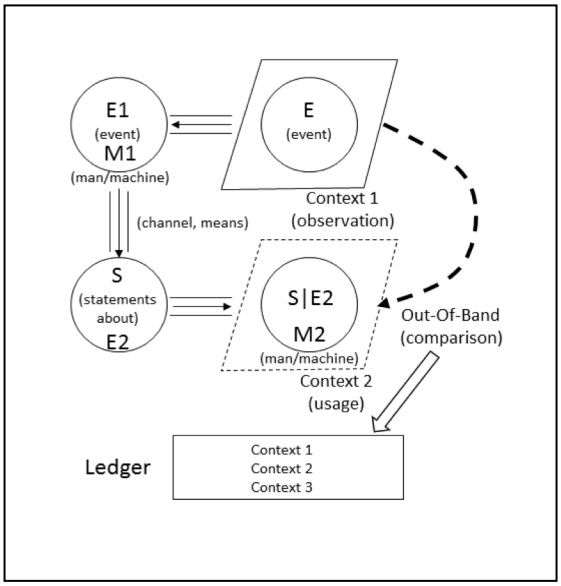


Fig. 2. Model with out-of-band context records sent to a ledger.

TABLE II. DATA LEDGER ENTRIES RECORDING CONTEXT CHANGE HISTORY

Estimates of environment (context instances at time)	Data objects with links to context
Context 1 (t1+)	DO1(t >= t1)
	DO2(t > t1)
Context 2 (t2+)	DO3(t >= t2)
Context 3 (t3+)	DO4(t >= t3)
Context 4 (t4)	DO_{future}(t >= t4)

A. Scenario 1

Upon element creation, two hashes are created: one for the element containing the (DO, DE) pair and another for the DO. A comparison between the checksums for (DO, DE) pairs is made, and when the coupled pairs match, the lite version discards the new information. The full version can be configured to create a permanent version that can be archived for provenance purposes on the archive system.

B. Scenario 2

Upon element creation, two hashes are created: one for the element containing the (DO, DE) pair and another for the DO. A comparison is made, and when the checksums do not match, the data object checksums are compared. If the DOs and the DOo are equal, it implies that the objects are the same but the environment has changed. The lite version would result in the operator manually examining the environment and determining whether the discrepancy is benign or not. The full version would use the DO as the index into the archive. The archive contains DE entries that pair with the DO, and each status indicates whether the event is benign or not. In addition, the Blockchain would be entered into the archive for later processing. This secondary processing would allow for examination of the changes to determine the point where the data became less reliable.

C. Scenario 3

Upon element creation, two hashes are created: one for the element containing the (DO, DE) pair and another for the DO. A comparison is made, and when the checksums do not match, the data object checksums are compared. If the DOs and the DOo are not equal, it implies that the object has changed. The lite version would result in the object being automatically invalidated. The full version would begin by extracting the DE values for comparison. If DEs and DEo are equal, then the problem is with the object not the environment. The DOo will act as the index into the archive. This scenario processing would result in the Blockchain for the object being entered into the archive for later processing. Secondary processing would allow for examination of the changes to determine the point where the data became less reliable.

D. Scenario 4

Upon element creation, two hashes are created: one for the element containing the (DO, DE) pair and another for the DO. A comparison between the checksums for (DO, DE) pairs is made, and when the coupled pairs do not match, the DOs and DOo hashed values are compared when they do not match the environments are extracted and compared when they do not match, the lite version discards the new information. The full version can be configured to create a permanent archive that can be inspected and compared for provenance purposes on the archive system.

V. CONCLUSIONS

This proposed model relies on combining digital signatures and aspects of Blockchain technologies on security data that are tightly bound to the defined environment, creating a coupling for evaluation when environmental or object states change. Cybenko et al. [17] recommended a similar solution in his work on CH, where he also stated the importance of accurate baseline data. This solution also relies on accurate baseline data that are stored in the out-of-band comparison system.

Cybenko et al. [17] also considered solving CH through modeling using the Byzantine General's problem [25]. The algorithmic solution for the Byzantine General's problem becomes resource intensive and may still fail to solve the data fidelity problem in the current data-rich environments that comprise security systems today. Furthermore, weaponized information is amplified, and the majority of bad data when modeled using the Byzantine General's solution will be accepted as good, faithful data.

This proposed model presents a non-traditional use of digital signatures and Blockchain technologies [27], [28] as a method of providing details that can be forensically used to reconstruct, with greater accuracy, the point in time where events crossed from benign to problematic. While certainly not a comprehensive solution, this work represents a potential first step in an ongoing process to begin the process of solving a security challenge that has recently gained prominence in

another environment, and it will likely present with greater frequency in the virtual environment.

REFERENCES

[1] R. Szfranski. "A theory of information warfare: Preparing for 2020", Air University Maxwell Airforce Base, 1997. Available at: http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA328193.

[2] I. Munro. "Information Warfare in Business: Strategies of Resistance and Control in the Network Society". London: Routledge, 2005.

[3] M. Nance, "The Plot to Hack America: How Putin's Cyberspies and WikiLeaks Tried to Steal the 2016 Election", New York: Skyhorse Publishing Inc., 2016.

[4] G. Commin and E. Filiol, "Unrestricted Warfare versus WesternTraditional Warfare: A Comparative Study", *Leading Issues in Cyber Warfare and Security Volume II*, Academic Conferences and Publishing International Limited; Reading, UK, 2015.

[5] J. Kallberg and B. Thuraisingham, From cyber terrorism to state actors' covert cyber operations. "Strategic Intelligence Management—National Security Imperatives and Information and Communications Technologies", Kidlington, Oxford: Butterworth-Heinemann, pp. 229–233, 2013.

[6] M. Gazzaniga, R. B. Ivry, and G. R. Mangun. "*Cognitive Neuroscience: The Biology of the Mind*", New York: W.W. Norton & Company Inc., 2014.

[7] T. M. Chen and S. Abu-Nimeh, "Lessons from Stuxnet," Computer, vol. 44, no. 4, pp. 91–93, 2011.

[8] FIPS Publication 46-3. "Data encryption standard (DES)", National Institute of Standards and Technology, Available at: csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf.

[9] V. Rimen and J. Daeman. "Advanced encryption standard", Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology, pp. 19–22, 2001.

[10] D. Naccache, D. M'Ralhi, S. Vaudenay, and D. Raphaeli. "Can DSA be improved? Complexity trade-offs with the digitial signature standard", Workshop on the Theroy and Application of Cryptographic Techniques, Springer Berlin: Heidelberg, pp. 77–85, 1994.

[11] L. T. Hosmer. "Trust: The connecting link between organizational theory and philosophy", Acad. Manage. Rev., vol. 20, no. 2, pp. 379–403, 1995.

[12] S. Kamouskos, "Stuxnet worm impact on industrial cyber-physical system security." IECON 2011 – 37th Annual Conference on Industrial Electronics Society, IEEE, November 7, 2011.

[13] P. Shakarian, "Stuxnet: cyberwar revolution in military affairs". Small Wars Journal, April 2011.

[14] G. Gerbner, "Toward a general model of communication," Educ. Technol. Res. Dev., vol. 4, no. 3, pp. 171–199, 1956.

[15] Communication theory website. Available at: http://communicationtheory.org/wp-content/uploads/2012/08/gerbner-general-model-of-communication.jpg.

[16] P. Pomerantsev and M. Weiss, "The meanance of unreality: how the Kremlin weaponizes information, culture and money", The Interpreter, 2014.

[17] G. Cybenko, A. Giani, and P. Thompson. "Cognitive Hacking a Battle for the Mind", Computer, vol. 35, no. 8, pp. 50–56, 2002.

[18] G. Stoneburner. "800-33, Underlying Technical Models for Information Technology Security." National Institute for Technology 2001. Available at: http://csrc.nist.gov/publications/nistpubs/800-33/sp800-33.pdf.

[19] M. Roesch, "Snort—Lightweight intrusion detection for networks", Proceedings of LISA '99: 13th Systems Administration Conference, Seattle, WA, pp. 229–238, November 7–12, 1999.

[20] S. Kamara, S. Fahmy, E. Schultz, F. Kerschbaum, and M. Frantzen, "Analysis of vulnerabilities in internet firewalls". Computers & Security, vol. 22, no. 3, pp. 214–232, 2003.

[21] D. Eastlake 3rd, S. Crocker, and J. Schiller, "Randomness recommendations for security" (no. RFC 1750), 1994, Available at: https://www.ietf.org/rfc/rfc1750.txt.

[22] P. De Boer and M. Pels, "Host-based intrusion detection systems", Amsterdam University: Amsterdam, 2005.

[23] C. Gates, M. P. Collins, M. Duggan, A. Kompanek, and M. Thomas. "More netflow tools for performance and security." Proceedings of Large Installation Systems Administration Conference (LISA) 2004, Atlanta, GA, vol. 4, pp. 121–132, November 2004.

[24] J. H. Lee, M. S. Brown, and R. D. Roesler, Mdsi Software Set, "System and method for creating validation rules used to confirm input data", US Patent 6,535,883.

[25] L. Lamport, R. Shostak, and M. Pease. "The Byzantine Generals Problem." ACM Transactions on Programming Languages and Systems (TOPLAS), vol. 4, no. 3, pp. 382–401, 1982.

[26] Y. M. Wang, D. Beck, B. Vo, R. Roussev, and C. Verbowski. "Detecting stealth software with strider ghostbuster." Proceedings of the 2005 International Conference on Dependable Systems and Networks (DSN '05), IEEE, pp. 368–377, 2005.

[27] S. H. Ammous, "Blockchain technology: what is it good for?" Available at: http://capitalism.columbia.edu/files/ccs/workingpage/2016/ammous_blockchain_technology_.pdf.

[28] M. Staples. "Blockchain is useful for a lot more than bitcoin", The Conversation, 2016. Available at: http://theconversation.com/blockchain-is-useful-for-a-lot-more-than-just-bitcoin-58921.