IJBT (2018) 01: 1-11
RESEARCH ARTICLE

From prediction to anticipation of cyber attacks

Michael Weiss[1]

Abstract With the rising volume and variety of cyber attacks, it has become increasingly harder for businesses and organizations to defend against attacks. The paper makes the case that to respond to this challenge, we need to anticipate new threats, not merely react to known threats. It reviews reactive approaches to cyber attacks where current actions are based on past behavior, and proactive approaches guided by predictions about the future.

Keywords: Cybersecurity · Prediction · Anticipation · Threats · Cyber attacks

1 Introduction

In this paper, we discuss that to address the mounting threat of cyber attacks we need to anticipate new threats, not merely react to known threats. With the rising volume and variety of cyber attacks, it has become increasingly harder for businesses and organizations to defend against attacks. The paper makes the case that to respond to this challenge, we need to switch from a reactive to a more proactive approach to cybersecurity.

The existing approach to cybersecurity has been mostly reactive. For example, traditional mechanisms to defend against malware are based on matching attacks against known signatures. As new strains of malware are discovered, signatures are added to the list of known attacks. This approach works only as long as the volume and variety of attacks is low. With the increase in the number of attacks, however, by the time a new attack has been identified, significant damage may already have been done [1].

The literature on the cognitive basis of prediction [2–4] provides an overarching perspective for this paper. As suggested by [2], prediction comprises two types of activities: on one hand, forecasting or prediction in the narrow sense, and anticipation on the other. The key distinction between both is that in the former, current actions are based on past behavior and that in the latter, predictions about the future guide current actions.

In the remainder of the paper, after describing the method used to conduct the review of existing approaches, we first describe approaches to predicting cyber attacks from past behavior. We then argue that to address the challenges imposed by the

✉ Michael Weiss:
michael_weiss@carleton.ca

[1] Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada

rapidly changing cybersecurity environment, approaches inspired by the study of anticipatory thinking are required.

2 Method

The review of techniques reported in this paper was conducted by first identifying the candidate papers through databases like Google Scholar. In a second step, by following references, for applying topic modeling to extract the underlying, latent themes from those papers. Topic modeling provides an alternative to manual clustering of articles and allows us to identify non-obvious connections between the ideas expressed in the selected papers.

2.1 Selection of papers on prediction and anticipation

For this paper, we collected a set of 108 documents related to reactive and proactive approaches to prediction. The papers included both highly cited articles matching either the keywords “machine learning” or “anticipation” and “cybersecurity”. Also included are key papers cited by those papers, e.g., papers on the cognitive foundation of anticipatory thinking. To allow for emerging topics to be represented in the collection, we also handpicked recent conference papers and theses with a lower number of citations. A more thorough systematic review of the literature is the subject of future work.

2.2 Creating the topic model

Topic modeling is a probabilistic technique for extracting latent topics from a set of documents [5, 6]. It does not require a human to label the documents, and, thus, belongs to the class of unsupervised learning algorithms. Topic modeling has been applied to areas such as analyzing emerging trends of security vulnerabilities [7] and the evolution of scientific fields [8].

A common topic modeling technique is Latent Dirichlet Allocation (LDA) [5]. In LDA, documents are represented as a bag of words, i.e., the order of the words does not matter. Given a sample of documents and the number of topics, LDA produces a distribution $P(z \mid d)$ that a document d is about a given topic z and a distribution $P(w \mid z)$ that a topic z is associated with a word w.

To construct the topic model, we used only paper abstracts as documents. There is a balance between the length of the documents provided to a topic model, the number of potential topics each document contains, and the ease with which we can understand the created topic model. By focusing on the abstracts, we emphasize highlights of the articles as summarized by its authors. Abstracts can also be scanned more quickly than the full articles when examining the articles associated with a given topic.

Table 1 The major latent topics in prediction and anticipation research

ID	Weight	Keywords	Name
9	0.583	security, information, detection, systems, network, system, attacks	Intrusion detection
8	0.243	cybersecurity, game, failures, theory, domain, afd, failure	Adversarial thinking
6	0.238	data, prediction, learning, science, processing, model, mining	Predictive analytics
4	0.230	software, vulnerabilities, vulnerability, metrics, code, security, development	Vulnerability prediction
5	0.225	attack, risk, engineering, scenarios, cyber, set, graphs	Scenario modeling
3	0.158	attacks, anticipation, hijacking, types, emails, phishing, mechanisms	Complex attack detection
7	0.103	identity, theft, botnet, breach, news, company, software	Text and network mining
1	0.098	anomalies, events, performance, feature, ability, hros, extrapolation	Anticipation of failure
2	0.097	malware, patterns, threat, behavior, classification, hotspot, file	Behavioral analysis

We then created a topic model, and iterated the model with different numbers of topics, until a set of mostly independent clusters of documents emerged. The literature also suggests that 10-12 topics are a good heuristic value for the number of topics [9]. Table 1 shows the output of this step. For each topic, the keywords associated with the topics produced by the topic model, a name assigned by the researcher, and the topic weight are listed.

As apparent from Table 1, the topic "intrusion detection" is the topic with the highest weight and the topic "behavioral analysis" is the topic with the lowest weight.[2] The clusters obtained by topic modeling provide the basis for our review of prediction and anticipation techniques. The application of topic modeling enabled the discovery of subtle connections between articles via common topics that may have easily been missed in a manual expert review.

3 Prediction

All techniques reviewed in this section assume that to predict the future you are restricted to examining the past. This premise covers the most current machine learning and predictive analytics techniques. The techniques correspond to six of nine topics in Table 1 and comprise the majority of the articles found. They include predictive analytics, intrusion detection, behavioral analysis, text and network mining, complex attack detection, and vulnerability prediction.

[2] A high topic weight indicates an area that much of the existing research has focused on, whereas a low weight often suggests an emerging research area.

3.1 Predictive analytics and machine learning

Predictive analytics is the art of building and using models to make predictions. It uses machine learning to build those models [10]. There is no fixed set of methods, yet, for applying machine learning to cybersecurity. Some peculiarities of cybersecurity also make it more challenging to apply machine learning: the evolution of attacks that requires learning to be incremental; a high data volume; a high cost of errors; the need to label training data which requires substantial effort preparing the data; and a lack of data sets [11].

A survey of machine learning techniques for cybersecurity is provided in [12]. Machine learning techniques can be grouped into supervised, unsupervised, and hybrid techniques. Supervised methods require data instances to be assigned categories or labels (e.g., “spam” or “ham”). These include methods like decision trees, Naive Bayes and Support Vector Machines. Unsupervised methods do not require labels. They include k-means clustering and association mining. Hybrid methods can be used in supervised or unsupervised modes, and include neural networks, genetic algorithms, and Bayesian networks.

One particular challenge in the cybersecurity context is that machine learning models can themselves be attacked [13]. Through carefully crafted attacks, attackers can gain an understanding of the internal state of a machine learning model, which allows them to attack more effectively in the future. Attacks against machine learning models fall into two categories: integrity attacks (an attacker tries to get the algorithm to accept a harmful attack as benign) and availability attacks (attackers train the model to classify benign instances as harmful) [13]. Recent research [14] calls for algorithms that can unlearn what they had incorrectly “learned” from attacks against them.

3.2 Intrusion detection

Intrusion detection is the process of monitoring a network or system for intrusions or attacks [15]. Common types of attacks include scanning attacks which are used to gather information about a network or system, penetration attacks in which an attacker tries to gain unauthorized access to a system, and Denial of Service (DoS) attacks that aim to exhaust the resources of a network. Systems built to detect intrusion attempts can be grouped into misuse and anomaly detection [15]. Misuse detection can detect known attacks (i.e., abnormal behavior) with predefined characteristics. Conversely, anomaly detection considers intrusions to be deviations from normal behavior.

Manual analysis of intrusions is limited to mitigating known attacks. As attacks evolve, human-created rules used to detect them become ineffective [1, 16]. Due to the volume, variety, intensity, and velocity of data that they have to process analysts are also missing many malicious security events [16]. Machine learning helps automate the analysis of attacks. Machine learning methods for misuse detection are typically based on a classification of attacks against signatures [12]. Clustering algorithms can support anomaly detection. They can also be used to extract new signatures for misuse detection [12].

In recent work, hybrid analyst-in-the-loop approaches have been proposed that combine the best of human analysts and machine learning algorithms [17, 18]. They respond to the problems of how to present analysts with the right information and the lack of current datasets for training machine learning classifiers. Analysts can become

overwhelmed with monitoring real-time security events given their limited time "budget" to investigate alerts. One solution is to combine automation (rule-based detection of potential threats) and exploration (visualization of the associated information) [17]. Exploration allows analysts to discover new anomalies that are not yet covered by rules, select features useful for detection, and validate existing rules.

The lack of labeled data makes it difficult to use supervised machine learning models. The system proposed in [18] combines unsupervised and supervised learners. It uses outlier detection to identify suspicious activities. Suspicious activities are presented to the analyst who labels activities as actual attacks or normal behavior. These labels can then be used to train a supervised learner. The analyst helps the system identify new and evolving attacks, while machine learning can predict known attacks without input from the analyst.

3.3 Behavioral analysis

The traditional approach to detecting malware is to define signatures and match incoming malware samples against them. Manually defining signatures is a laborious process and inadequate to keep up with changing strains of malware. There are over 6 million new strains of malware each year [19]. It is also easy for an attacker to circumvent signature-based approaches. Malware can be modified without altering its behavior but changing its signature.

More recent approaches to malware detection are based on behavioral analysis. Behavioral analysis profiles malware by creating a trace of the instructions it calls and uses a combination of clustering and classification techniques [20]. It does not require a manually labeled set of malware samples. Instead, malware samples are initially clustered by the similarity of their behavioural profiles. These clusters can then be used as labels to train a classifier. Behavioral analysis is also robust against the evolution of malware. Even when changes are made to the malware, its behavior will be similar.

3.4 Text and network mining

Text and network mining algorithms have shown much promise for predicting cyber attacks. Methods discussed elsewhere in this paper heavily rely on text analysis. For example, behavioral profiles used for malware analysis (section 3.3) contain sequences of instructions and their arguments that the malware invokes on the host system. These can be translated into features by extracting n-grams from those sequences and creating a bag-of-words representation on which the classification and clustering algorithms can operate [20].

Similarly, methods for vulnerability prediction (section 3.6) often rely on documents (e.g., CVE reports) or sources that can be interpreted as documents (e.g., source code). In [7], topic modeling is used to examine trends in CVE (Common Vulnerability and Exposure) reports. The authors first identify latent topics in the reports to categorize vulnerabilities. They then compute the weight of each topic across different years to understand vulnerability trends. In [21], source code files are treated as documents and symbols in the code as words. A bag-of-words

representation of the code provides input to a classifier that predicts the likelihood of a software system containing vulnerabilities.

Network mining applies techniques from social network analysis to help identify, e.g., the central nodes in a network, clusters of nodes, or the roles nodes play in a network. Nodes represent actors or systems and the ties that connect them stand for their relationships. Network mining has been applied to analyze botnets [22]. The authors create a network model of the communication patterns between the hosts of a botnet using NetFlow data and analyze the network using a PageRank algorithm to identify the central hosts.

3.5 Complex attack detection

Increasingly, attacks are executed in multiple steps, making them harder to detect. Such complex attacks require that defenders recognize the separate stages of an attack, possibly carried out over a longer period, as belonging to the same attack. Complex attacks can be divided into exploration and exploitation phases [23]. Exploration involves identifying vulnerabilities and scanning and testing a system. It is how an attacker gathers information about the system. Exploitation involves gaining and maintaining access. At this stage, the attacker applies the know-how gathered during the exploration stage.

An example of a complex attack that combines exploration and exploitation is a sequence of a phishing attack, followed by an exfiltration attack. First, attackers will attempt to collect information on the organization they intend to attack, e.g., names of key employees. Then, they will craft a targeted phishing attack. The phishing attack allows the attackers to gain access to the user's system and install malware. The purpose of the malware could be to extract files from the user's machine or to use the user's machine as an attack vector to attack other machines in the organization's network.

A phishing attack is usually carried out by sending an email purporting to come from a trusted source and tricking its receiver to click on a URL that results in installing malware on the user's system. This malware then creates a backdoor into the user's system for staging a more complex attack. Phishing attacks can be recognized both by the types of keywords used in the email (as with a spam email), as well as by the characteristics of URLs included in the message [24]. Features that have been used successfully to detect phishing attacks include URLs that include IP addresses, the age of a linked-to domain, and a mismatch between anchor and text of a link.

3.6 Vulnerability prediction

Vulnerabilities are weaknesses, flaws or deficiencies that can be exploited by threats to cause harm to an asset. This section focuses on software vulnerabilities. Given that not all vulnerabilities are of equal impact and that resources are limited, authors of software need to prioritize on which patches to create and system administrators on which of these patches to deploy. Vulnerability prediction can assist in this task by

predicting the kinds of vulnerabilities that exist in a system and the risk of them being exploited.

There are two ways to predict vulnerabilities: based on metadata (i.e., information about the vulnerabilities) or from inherent properties of a system. A natural question to ask is whether we can predict the timing and impact of an exploit from the information in a CVE report. A classifier for this task is described in [25]. The features it uses include text fields of the report (e.g., description), timestamps (e.g., the time between the first time the vulnerability was reported and the time of its exploit), and cross-references to other reports. As new information about vulnerabilities that have been exploited becomes available, the classifier can be retrained to incorporate this information.

A machine learning model to predict the likelihood that a software system contains vulnerabilities from the system's source code (an inherent property) has been described in [21]. This approach trains a classifier directly on the source code rather than on quality metrics derived from the code. It was even found to be capable of predicting vulnerabilities in future releases of the same software. It has been demonstrated that architectural flaws (another inherent property) are also good indicators of security issues [26]. Changes to the architecture (including patches to fix security issues) can result in new vulnerabilities. The authors calculate structural (dependencies between files) and evolutionary metrics (co-changes among files) from source code and its revision history. Certain patterns in those metrics (e.g., frequent changes) indicate architectural flaws known to correlate with security issues.

4 Anticipation

The techniques in this section allow us to select actions based on their anticipated consequences. They furthermore enable us to operate in a continually evolving environment. This ability sets anticipatory techniques apart from predictive techniques. Techniques in this section include adversarial thinking, scenario modeling, and anticipation of failure. They are embodied in some of the more recent approaches to dealing with cyber attacks.

4.1 Adversarial thinking

Game theory studies the strategic interaction between players. Training in game theory has been shown to help sensitize students to the role of human adversaries in cybersecurity [27]. It can also be used to model multiple levels of reasoning like level-k reasoning (e.g., a level-2 strategy would be for operators to expect attackers to try to anticipate their moves and to act accordingly). Game theoretic models have also been implemented in algorithms to protect critical infrastructures and for mechanism design [28].

Coherence networks provide a way of representing competing hypotheses and the evidence they explain [29]. It has been used to anticipate, understand and respond to actions of an opponent in adversarial problem-solving situations (e.g., military decisions) [29]. In this approach, a hypothesis and its evidence is considered 'the more coherent, the less supporting evidence' the hypothesis requires. Evidence and

hypotheses have associated activation levels and these activation levels are propagated through the links among them. As evidence is observed, it propagates through the coherence network and updates the activation level of associated hypotheses and evidence. This technique can capture the dynamics of the evolution of hypotheses and evidence.

4.2 Scenario modeling

Scenario modeling was first developed as a management approach to support strategic decision-making [30]. It is a method to examine possible alternative futures given a projection of trends. The goal of scenario modeling is not to predict the future, but to prepare for an uncertain, unfolding future. Often only external scenarios are modeled. However, modeling internal scenarios (resources and capabilities) provides insights into an organization's capability to execute. In the context of cybersecurity, external refers to attacks and attackers and internal to defenders and their capabilities.

A scenario can be thought of as a sequence of observable indicators or signals. In the context of scenarios, we often focus on "weak signals" as signals that we need to pay close attention to because of their far-reaching impact [30]. Multiple scenarios can be combined into a tree in which internal nodes indicate indicators and branches represent possible alternatives [31]. This construct provides the basis for scenario generation and failure analysis.

Applications of scenario modeling to cybersecurity include: modeling attacks, generating attack scenarios, and assessing the impact of attacks. Different approaches have been proposed to generate potential attack scenarios: merging existing attacks [32] and applying attack patterns [33]. An AI planning approach for generating attack scenarios has been described in [35]. Recent work on intrusion prevention systems also suggests that we can use scenarios to assess the impact of ongoing complex attacks [34].

For instance, common attack patterns can be extracted from a public collection of attack types (CAPEC [36]) and codified in the form of patterns [33].

These patterns capture knowledge about attacks from the perspective of an attacker: each captures an attacker's goal and the steps to carry out the attack. The authors then show how this collection of attack patterns can be used to generate possible attack scenarios given an attacker's goals. This approach was able to replicate an expert vulnerability analysis.

4.3 Anticipation of failure

Mindfulness is the capability to discover and manage unexpected behavior [37]. It combines the concept of anticipation with that of resilience. Anticipation comprises preoccupation with failure, reluctance to simplify, and sensitivity to operation [37]. When we apply those processes to cybersecurity, they imply that we must pay close attention to signs of abnormal behavior in our networks and systems, question what we take for granted (i.e., expect attacks to evolve), and always search for a coherent explanation of our observations (i.e., maintain multiple competing hypotheses about the state of the world).

Anticipatory Failure Determination (AFD) [31] is an approach for envisioning failure scenarios. Its focus is not on learning from what failures have occurred in the past, but on discovering what failures may occur and how they can be brought about. AFD has recently been applied to model failure scenarios in cybersecurity [38]. The goal of the approach is to build an inventory of resources (indicators, tools, people, vulnerabilities, and information) that have enabled failures in the past. Failure scenarios start from failure indicators and work their way back through a causally linked chain of resources.

5 Discussion

Approaches to cybersecurity based on prediction (in the narrow sense, in which we have been using it in this paper) are limited in the extent to which they can cope with the evolution of cyber attacks. With the authors of [4], we made a distinction between prediction – basing actions on the past – and anticipation – basing current actions on future consequences of those actions.

Anticipation is a "future-oriented action, decision, or behavior based on a (implicit or explicit) prediction" [4]. Anticipatory systems include a forward model [4] that allows them to form hypotheses about the next response from the environment. Predictions from the forward model enable a system to compare predicted and observed responses and adjust its behavior [3]. This agrees with psychological experiments that show that current behavior is a function of both the context and the expected consequences of the behavior [3].

The forward model of biological anticipatory systems has its equivalent in the representation of possible futures in anticipatory techniques. In the case of adversarial thinking, possible futures are explored by the level-k reasoning of game theory or the competing hypotheses of coherence networks. In scenario modeling, the forward model consists of the generation and subsequent monitoring of attack scenarios. In anticipation of failure, mindful practices and reasoning backward from failures provide the anticipatory capability.

6 Conclusion

From this paper, we understand the need for a shift in the mindset on how we deal with cyber attacks, from a reactive to a more proactive approach founded in the emerging techniques of anticipatory thinking. This shift is required to manage an environment characterized by a significant increase in the volume and variety of cyber attacks. The paper also calls for more research on the proactive approach to cybersecurity.

References

1. Chen, HM, Kazman, R, Monarch, I, Wang, P: Can cybersecurity be proactive? A big data approach and challenges. IEEE Hawaii International Conference on System Sciences, 5978–5987 (2017)
2. Bubic A, Von Cramon DY, Schubotz RI: Prediction, cognition and the brain. Frontiers in Human Neuroscience, 4, 25:1-25:15 (2010)
3. Butz, MV, and Pezzulo, G: Benefits of anticipations in cognitive agents. The Challenge of Anticipation, 45–62, Springer (2008)

4. Pezzulo, G, Butz, MV, and Castelfranchi, C: The anticipatory approach: definitions and taxonomies. The Challenge of Anticipation, 23–43, Springer (2008)
5. Blei, DM, Ng, AY, and Jordan, MI: Latent Dirichlet Allocation. Journal of Machine Learning Research, 3, 993–1022 (2003)
6. Blei, DM: Probabilistic topic models. Communications of the ACM, 55(4), 77–84 (2012)
7. Neuhaus, S, and Zimmermann, T: Security trend analysis with CVE topic models. IEEE International Symposium on Software Reliability Engineering, 111–120 (2010)
8. Hall, D, Jurafsky, D, and Manning, CD: Studying the history of ideas using topic models. Conference on Empirical Methods in Natural Language Processing, 363–371, Association for Computational Linguistics (2008)
9. Mathew, G, Agarwal, A, and Menzies, T: Trends in topics at SE conferences (1993-2013). arXiv preprint arXiv:1608.08100, 1–19 (2017)
10. Kelleher, JD, Mac Namee, B, and D'Arcy, A: Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies. MIT Press (2015)
11. Sommer, R, and Paxson, V, Outside the closed world: On using machine learning for network intrusion detection. IEEE Symposium on Security and Privacy (SP), 305–316 (2010)
12. Buczak, AL, and Guven, E: A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, **18**(2), 1153–1176 (2016)
13. Barreno, M, Nelson, B, Joseph, AD, and Tygar, JD: The security of machine learning. Machine Learning, **81**(2), 121–148 (2010)
14. Bailetti, T, Gad, M, and Shah, A: Intrusion Learning: An Overview of an Emergent Discipline. Technology Innovation Management Review, **6**(2), 15–20 (2016)
15. Deka, RK, Kalita, KP, Bhattacharya, DK, and Kalita, JK: Network defense: Approaches, methods and techniques. Journal of Network and Computer Applications, **57**(C), 71-84 (2015)
16. Ferguson, AJ, and Harris, NE: Moving big-data analysis from a forensic sport to a contact sport using machine learning and thought diversity. Journal of Information Warfare, **14**(2), 53–70 (2015)
17. Shah, A, Abualhaol, I, Gad, M, and Weiss, M: Combining exploratory analysis and automated analysis for anomaly detection in real-time data streams. Technology Innovation Management Review, **7**(4), 25–31 (2017)
18. Veeramachaneni, K, Arnaldo, I, Korrapati, V, Bassias, C, and Li, K: AI2: training a big data machine to defend. IEEE International Conference on Big Data Security on Cloud (Big Data Security), International Conference on High Performance and Smart Computing (HPSC), and International Conference on Intelligent Data and Security (IDS), 49–54 (2016)
19. Benzmu¨ller, R: Malware trends 2017. G DATA Security Blog, https://www.gdatasoftware.com/blog/2017/04/29666-malware-trends-2017 (2017)
20. Rieck, K, Trinius, P, Willems, C, and Holz, T: Automatic analysis of malware behavior using machine learning. Journal of Computer Security, **19**(4), 639–668 (2011)
21. Scandariato, R, Walden, J, Hovsepyan, A, and Joosen, W: Predicting vulnerable software components via text mining. IEEE Transactions on Software Engineering, **40**(10), 993–1006 (2014)
22. Francois, J, Wang, S, and Engel, T: BotTrack: tracking botnets using NetFlow and PageRank. International Conference on Research in Networking, 1–14, Springer (2011)
23. Esteves, J, Ramalho, E, and De Haro, G: To improve cybersecurity, think like a hacker. MIT Sloan Management Review, **58**(3), 71–77 (2017)
24. Fette, I, Sadeh, N, and Tomasic, A: Learning to detect phishing emails. ACM International Conference on World Wide Web, 649–656 (2007)
25. Bozorgi, M, Saul, LK, Savage, S, and Voelker, GM: Beyond heuristics: learning to classify vulnerabilities and predict exploits. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 105–114 (2010)
26. Feng, Q, Kazman, R, Cai, Y, Mo, R, and Xiao, L: Towards an architecture-centric approach to security analysis. Working IEEE/IFIP Conference on Software Architecture (WICSA), 221-9230 (2015)
27. Hamman, ST, Hopkinson, KM, Markham, RL, Chaplik, AM, and Metzler, GE, Teaching game theory to improve adversarial thinking in cybersecurity students, IEEE Transactions on Education, 2017 (in press)
28. Alpcan, T: Game theory for security. Encyclopedia of Systems and Control, 495–499 (2015)
29. Thagard, P: Adversarial problem solving: Modeling an opponent using explanatory coherence. Cognitive Science, **16**(1), 123–149 (1992)

30. Schoemaker, PJ: Scenario planning: a tool for strategic thinking. Sloan Management Review, **36**(2), 25–40
31. Kaplan, S, Haimes, YY, and Garrick, BJ: Fitting hierarchical holographic modeling into the theory of scenario structuring and a resulting refinement to the quantitative definition of risk. Risk Analysis, **21**(5), 807–807 (2001)
32. Ahn, W, Chung, M, Min, BG, and Seo, J: Development of cyber-attack scenarios for nuclear power plants using scenario graphs. International Journal of Distributed Sensor Networks, **11**(9), 836258 (2015)
33. Li, T, Paja, E, Mylopoulos, J, Horkoff, J, and Beckers, K: Security attack analysis using attack patterns. IEEE International Conference on Research Challenges in Information Science (RCIS), 1–13 (2016)
34. Albanese, M, and Jajodia, S: A Graphical Model to Assess the Impact of Multi-Step Attacks. The Journal of Defense Modeling and Simulation, 1548512917706043, 1–15 (2017)
35. Pasquale, L, Hanvey, S, Mcgloin, M, and Nuseibeh, B: Adaptive evidence collection in the cloud using attack scenarios. Computers & Security, 59, 236–254 (2016)
36. MITRE, Common Attack Pattern Enumeration and Classification. https://capec.mitre.org/about/index.html, last accessed in Nov. 2017
37. Weick, KE, Sutcliffe, KM, and Obstfeld, D: Organizing for high reliability: Processes of collective mindfulness. Crisis management, **3**(1), 81–123 (2008)
38. Badalkhani, P: Using publicly available information to predict cyber failures. Master's Thesis, TIM Program, Carleton University (2016).