information

MDPI

Review

Attacker Behaviour Forecasting Using Methods of Intelligent Data Analysis: A Comparative Review and Prospects

Elena Doynikova [1,*], Evgenia Novikova [1,2] and Igor Kotenko [1]

[1] St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, St. Petersburg 199178, Russia; novikova@comsec.spb.ru (E.N.); ivkote@comsec.spb.ru (I.K.)
[2] Saint Petersburg Electrotechnical University "LETI", Department of computer science and technology, St. Petersburg 197022, Russia
* Correspondence: doynikova@comsec.spb.ru

Received: 18 February 2020; Accepted: 16 March 2020; Published: 23 March 2020

Abstract: Early detection of the security incidents and correct forecasting of the attack development is the basis for the efficient and timely response to cyber threats. The development of the attack depends on future steps available to the attackers, their goals, and their motivation—that is, the attacker "profile" that defines the malefactor behaviour in the system. Usually, the "attacker profile" is a set of attacker's attributes—both inner such as motives and skills, and external such as existing financial support and tools used. The definition of the attacker's profile allows determining the type of the malefactor and the complexity of the countermeasures, and may significantly simplify the attacker attribution process when investigating security incidents. The goal of the paper is to analyze existing techniques of the attacker's behaviour, the attacker' profile specifications, and their application for the forecasting of the attack future steps. The implemented analysis allowed outlining the main advantages and limitations of the approaches to attack forecasting and attacker's profile constructing, existing challenges, and prospects in the area. The approach for attack forecasting implementation is suggested that specifies further research steps and is the basis for the development of an attacker behaviour forecasting technique.

Keywords: cyber attack; attacker; attacker profile; attacker behaviour; metrics; features; attributes; intelligent data analysis; attack forecasting; comparative review

1. Introduction

The attacker model plays an important role in the tasks of the attack modelling, forecasting, and risk analysis. Existing approaches consider different attacker's characteristics when modelling attacks. Some of them use high level goals of the malefactor [1]—hackers, spies, terrorists, corporate raiders, professional criminals, vandals, and voyeurs.

Others approaches analyze the location of the attacker—internal or external [2]—and the complexity of the vulnerabilities they exploit [3]—script kiddies, hackers, and botnet owners.

In [2], the classification of attackers based on several attributes is suggested. The analyzed parameters include the quantity of the malefactors, their motives, and their goals, which allows authors to define three types of attackers—individuals, organized groups, and intelligence agency.

Federal Service for Technical and Expert Control (FSTEC) of Russian Federation classifies attacker according to its skills and location in the system—internal attacker with low skills, internal attacker with medium skills, internal attacker with high skills, external attacker with low skills, external attacker with medium skills, and external attacker with high skills.

Various approaches are used to further clarify the type of attacker carrying out an ongoing attack. These approaches include the following:

1. Techniques based on attack graph analysis.
2. Techniques based on hidden Markov model.
3. Techniques based on fuzzy inference.
4. Techniques based on attributing cyber attacks using intelligent data mining techniques including neural networks, statistics, and so on.

However, it is possible to highlight several limitations of these techniques. One of them is the lack of a unified and validated approach to the attacker model description. According to these approaches, different attackers' attributes result in different attackers' profiles, and these approaches as a rule do not consider the latest paradigm shifts and novel attack vectors that appear owing to the development of the Internet of Things (IoT), cyber-physical systems, software defined networking (SDN), 5G mobile networks, and so on. Another significant problem in the attacker profiling process is the lack of consistent labeled datasets for model training.

There are currently a number of surveys in the area of attack forecasting and prediction. For example, in 2016, Gheyas and Abdallah [4] surveyed the detection and prediction of insider threats. In [5], the authors investigated the attack projection, prediction, and forecasting methods in cyber security. They distinguish between attack projection that relates to the next adversary steps [6]; attack intention recognition, which deals with detection of the final malefactor goal [7]; attack/intrusion prediction, which relates to the definition of which type of attack will take place, as well as when and where it will arise [8]; and, finally, network situation forecasting, which is connected with assessment of possible cyber security risks and their evolution. The authors outlined four different classes of the approaches based on the type of mathematical model used—discrete models (attack graphs, Bayesian networks, Markov Models), continuous models (time series analysis, Grey models), machine learning techniques and data mining techniques, and other approaches (similarity based, among others). They also focused on the problem of the source data used for predictions as different approaches operate on different levels of abstraction and require different types of data. They showed that the following types of input data can be used: (1) raw data, such as network traffic and system logs; and (2) abstract data, such as alerts from intrusion detection/protection systems and/or numerical representation of network security state. The authors discussed the advantages and limitations of each approach and showed the current status of each approach, that is, proof of concept or live tool. However, these surveys did not address the issues of the attacker profile definition or attacker attribution and its influence on the attack forecasting process.

Another interesting review of attacker models and profiles for cyber-physical systems (CPSs) is provided in [9]. The authors focused on the related work on the following: (1) attacks against CPS and ad-hoc attacker models, (2) profiling attackers for CPS, and (3) generic attacker models for CPS. They reviewed works that discuss attackers who target or leverage the physical layer in their attacks (mechanical, electrical interactions). The authors gave the main definitions concerning the attacker and attacker's profile. For example, they define an attacker as a person(s) aimed to achieve some malicious goal in the system, and an attacker profile as a template listing possible actions, motivations, or capabilities of the attacker. They note that an attacker model (together with compatible system models) should represent all possible interactions between the attacker and the system. Besides, they also include the constraints for the attacker model such as finite computational resources and no access to shared keys.

The authors reviewed 19 related works and came to the following conclusions:

1. Seven works explicitly use different attacker profiles, seventeen define dimensions, and the vast majority use actions to characterize the attacker. Just two works define a system model and perform risk analysis without explicitly considering an attacker model. This shows the trend of

defining an attacker model to perform security analysis of CPS and, at the same time, there exist various ways to model the attacker.

2. All the papers share the same actions or the same intuitions on the attackers, but they apply those actions to different definitions of attacker models.
3. Different works propose different attacker profiles. The boundaries between the different attacker profiles are not well defined, thus it is hard to classify a specific attacker as one specific profile. The authors outline the following six types of attackers based on related research: (1) a basic user [10,11] (also known as script kiddie, unstructured hacker, hobbyist, or cracker) uses already established and potentially automated techniques to attack a system, and has average access to hardware, software, and Internet connectivity; (2) an insider [11–14] (disgruntled employees or social engineering victims) can cause damage to the target depending on the employment position or the system privileges he/she owns (e.g., user, supervisor, administrator)—this type is of high importance for systems that are mainly protected through air-gaps between the system network and the outside world (often used in CPS); (3) a hacktivist [10–12] aims to promote a political agenda, often related to freedom of information (e.g., Anonymous); (4) a terrorist [11–13], also known as cyber-terrorist, is a politically motivated attacker who uses information technology to cause severe disruption or widespread fear [15,16]; (5) a cybercriminal [10–14] (sometimes called black hat hacker or structured hacker) is an attacker with extensive security knowledge and skills, he/she takes advantage of known vulnerabilities, and potentially has the knowledge and intention of finding new zero-day vulnerabilities, his/her goals can range from blackmailing to espionage (industrial, foreign) or sabotage; (6) a nation-state [10–13] is an attacker sponsored by a nation/state, and his/her targets are usually public infrastructure systems, mass transit, power or water systems, and general intelligence.
4. Finally, the authors outlined nine common parameters that are used to generate metrics. Examples of metrics are as follows:

    a. tools (resources) available, also known as attacklets, or actions in the abstract definition of the attacker model—these define which types of tools are available to the attacker;
    b. camouflage or preference to stay hidden—expresses the aim and/or the ability of the attacker to not be tracked down after or while performing an attack;
    c. distance to the CPS—an attacker can be located in another country, within WiFi range, or possibly have direct access to the system.

The authors also introduced the multilevel framework of metrics that is aimed to correlate low level events with high level events in order to determine the attacker profile. The limitation of the approach is that it does not establish techniques and methods linking low level events with high level events. For example, the financial support metric (which can take values of low, medium, or high) expresses what budget the attacker has in order to perform an attack. However, it is not clear how the budget can be calculated on the basis of the security events registered in the system.

To conclude, modern monitoring tools and data analysis systems give new possibilities in the area of the attacker's profile construction and prediction based on the traces that the attacker leaves in the system. We argue that an approach to attack forecasting that uses relations between features in the raw security related data, attacker attributes that represent his/her behaviour, and attack development is promising for timely and efficiently counteracting cyberattacks. In this paper we start with reviewing studies that take into account such relations as soon as it is not considered in detail in the aforementioned surveys. We analyze the latest research in this area, existing challenges, and possible solutions, and conclude with a general description of the approach that can be used for forecasting attacker's goals.

Thus, the main contribution of this paper is as follows:

- Comparative analysis and classification of existing techniques for attackers' behaviour forecasting and used characteristics of attackers.

- Existing challenges and solutions in the considered area.
- A common approach to attack forecasting task implementation that specifies further research steps and is the basis for the development of an attacker behaviour forecasting technique.

The paper is structured as follows. The comparative analysis of the existing approaches to the attacker's profile specification, the characteristics used to describe the attacker's profile, and the attack forecasting using it are given in Section 2. Section 3 outlines existing challenges and solutions in the considered area. Besides, a common approach to attack forecasting implementation that specifies further research steps is given in Section 3, and is the basis for the development of the attacker behaviour forecasting technique. The paper ends with the conclusion and future work prospects.

2. The Comparative Analysis of the Approaches to the Attacker's Profile Specification and Attack Forecasting

The review of the existing approaches to the attacker's profile definition and attack forecasting showed that it is possible to highlight two general approaches:

(1) the results of the attack prediction depend strongly on the attacker's model, and it is required to define the attacker's model explicitly;
(2) the attack forecasting is based on data analysis without explicit attacker's model specification, and the attacker's behaviour is constructed implicitly on the basis of the sequence of the security events.

The first group of approaches consists of the techniques based on attack graph analysis [17–28], hidden Markov model [29–34], and fuzzy logic [35–37].

The second group of approaches consists of techniques that implement attack attribution using machine learning techniques including neural networks, statistics, and some others [38–41].

In the subsections below, these approaches are given more in detail. The summarized information on these techniques, their advantages, and their limitations is given in Table 1.

It should be noted that different researchers use not only different techniques to specify the attacker's profile, but different concepts and terms to describe attacker's behaviour, for example, "threat model", "attacker's profile", and "attacker's behaviour".

Table 1. Techniques for attacker profile specification and its application for attack forecasting. HMM, hidden Markov model; RNN, recurrent neural network; CPS, cyber-physical system.

Method	Related Research	Datasets and Features	Attacker Classification	Characteristics	Description and Advantages	Limitations
Attack graph analysis based on the analysis of network topology, software and hardware configuration, relationships between users and services, and vulnerabilities	Schneier, B., 1999 [17]; Ingols, K. et al., 2009 [19]; Kheir, N. et al., 2010 [20]; Kotenko, I. and Stepashkin, M., 2006 [21]; GhasemiGol, M. et al., 2016 [25]; Wang, L. et al., 2008 [26]; Kotenko, I. and Doynikova, E., 2018 [27]	–	• internal attacker with low skills • internal attacker with medium skills • internal attacker with high skills • external attacker with low skills • external attacker with medium skills • external attacker with high skills	• attacker skills • location	• uses a list of vulnerabilities that could be exploited by the given attacker • shows every path that an attacker can use to gain privileges • the path to be selected is determined by the attacker's skills, as well as goals and motivation	• expert knowledge to define probabilities of the next attack action selection, attacker skills and location • used attacker's model utilizes in major cases only two dimensions—skills that could be defined explicitly or implicitly, and his/her location • definition of the probabilities is a complicated process and requires great expertise of the security administrator
HMM-based approach	Rashid, T. et al., 2016 [29]; Bar, A. et al., 2016 [30]; Deshmukh, S. et al., 2019 [31]; Jhawar, R. et al., 2016 [33]	events generated by honeypots	-	–	• allows modeling normal behaviour • allows detecting insider threat • links different types of events in one model that is able to reveal trends in attack implementation and is able to detect abnormal attack sequences	• the result strongly depends on the input dataset and the distribution of the events. • does not explicitly use the attacker's model • the skills of the attackers as well as motivation, available tools, and financial support are not considered
	Katipally, R. et al., 2011 [34]	network traffic with emulated attacks	• criminal groups • insiders • terrorists • hackers • phishers • nations • spyware/malware authors • bot-net operators	• goals • intension • level of expertise	• allows modeling normal and abnormal behaviour	• the attackers' profiles are used to generate different attacks in training set, therefore, it is not used for attack prediction • the result strongly depends on the input dataset and the distribution of the events

Table 1. *Cont.*

Method	Related Research	Datasets and Features	Attacker Classification	Characteristics	Description and Advantages	Limitations
Fuzzy inference-based approach	Çayirci, E., Rong, C., 2009 [42]	• NSL-KDD CUP1999 [42,43] • CAIDA DDoS Attack 2007 Dataset [44] • qualitative attributes of the events	-	-	• deals with uncertainty existing in intrusion detection domain • allows constructing fuzzy profiles of the user behaviour or anomalous activity • fuzzification process effectively smoothens the abrupt break of normal activity and intrusion • combination of machine learning and fuzzy logic	• focus on the intrusion detection • average accuracy is 96.54% [45], classical machine learning techniques outperform approaches based on fuzzy inference • the combination with different advanced machine learning techniques such as clustering and association rule mining allows one to enhance the accuracy
	Pricop, E. and Mihalache, S.F., 2015 [35]. Mallikarjunan, K.N. et al., 2018 [36]	high level abstract variables	6–9 profiles, e.g., • script kiddie • hacker • disgruntled employee • terrorists • industrial spy • cyber warrior	No unified set of attributes to define attacker's profile. example: • resource • skills • motivation	• the main goal is the risk analysis • there is an attempt to link malicious activities to the attacker's profiles • allows describing such fuzzy parameters as motivation or knowledge to determine the attacker's profile	• there is no link with low level events generated by security sensors • proof of concept • no unified approach to define characteristics describing attacker's profile • no unified set of attacker's profiles
Fuzzy inference and attack graphs	Orojloo and Abdollahi Azgomi, 2016 [46]	high level abstract data and use case		• skills • knowledge • access (location) • interaction	• the attacker's profile is not used explicitly; however, the approach considers the characteristics of the attacker • the basis is the attack graph • the probability of the transition between vertexes is calculated with consideration of the uncertainty in data	• strongly depends on the expertise of the cyber security specialist • proof of concept
Fuzzy inference based on statistical user profiles	Kudłacik, P. et al., 2016 [37]	qualitative attributes of the log events such as keyboard keys' sequences, characteristic data sequences retrieved from pointing device, chosen options, and so on.	-	-	• allows constructing a fuzzy user profile on the basis of the statistical profiles • low computational complexity • link low level events to users' behaviour profile	• limited with detection of abnormal user's behaviour

Table 1. *Cont.*

Method	Related Research	Datasets and Features	Attacker Classification	Characteristics	Description and Advantages	Limitations
Attack attributing	Rid, T. and Buchanan, B., 2015 [38]	behavioural indicators, including atomic indicators (IP addresses, email addresses, domain names, and small pieces of text) and computed indicators ('hash')	-	-	behavioural indicators allow pointing a specific adversary who has employed similar behaviours in the past	• will be useful only in the case of correct synthesis of information flows from the technical to the operational and strategic layers
Attribution of honeypot data	Fraunholz, D. et al., 2017 [40]	• source IP address • operating system • user–agent (protocol) • cookies	• guest • external employee • internal employee • activists • state-sponsored • ethical hacker • criminals • cracker • hobby hacker	• skill • resources • motivation • intention	attempt to link raw data and high-level metrics	• though the method is introduced for IT-Security in Industry 4.0, the specific features of CPS are not considered • techniques for calculation of specific metrics require further development
RNN	Perry, I. et al., 2018 [41]	• destination port • alert signature • alert category • alert severity • proto • source port • host	–	–	• allows predicting cyber attack behaviour • accuracy of 55% for teams classification and 80% for the next alerts prediction	• depends on data sets • specific classes of attackers are not considered

2.1. Attacker Behaviour Prediction Based on Attack Graphs

The construction and application of attack graphs for attack modeling and prediction is one of the most widely used approaches. First proposed in [17], this concept was developed in many other research papers [18–28]. In the general case, an attack graph is a set of linked nodes that represents the attacker's aims and actions. The construction of the attack graph is usually based on analysis of the network topology, vulnerability analysis, and software and hardware configuration analysis, and as the result, it shows dependencies between vulnerabilities and the overall security state of the target network.

In major cases, the attacker's model is defined via two important characteristics—his/her skills and location. For example, in the literature [19,21,27], these attributes are used to implement attack reachability analysis depending on the location (internal or external) and skills of the attacker (low, medium, or high). In fact, the level of the attacker's skills defines a list of vulnerabilities that could be exploited by the given attacker. In [27], the attacker's skills are correlated with meanings of "attacker skills" or "knowledge required" parameters of the attack patterns defined in Common Attack Pattern Enumeration and Classification (https://capec.mitre.org/) database and weaknesses from Common Weakness Enumeration (https://cwe.mitre.org/) database. This allows authors to link existing vulnerabilities to high-level malefactor activity such as "host discovery", "active operating system (OS) fingerprinting", and so on.

Wang et al. 2008 [26] assigned to each malefactor action a score that reflected the probability of its implementation. This score implicitly defines the attacker's skills, and in the approach, it was determined on the basis of the expert's knowledge regarding the vulnerability being exploited. Kheir et al. [20] enhanced the attack graph model by adding the service-dependency graph, which presents a network model for the relationships between users and services, showing how they perform their activities using the available services in order to increase the efficiency of the attack modeling.

In [25], the authors introduced the concept of the uncertainty-aware attack graph, which is used to handle the uncertainty of attack probability. This uncertainty appears owing to the measuring probability of vulnerability exploitation. In fact, it is difficult to find the precise probabilities for all attack graph nodes, and the authors suggest assigning the node probability in the form of interval values or constraints. However, both probability intervals and constraints are set by the experts. For example, the constrain may be described as follows [25]: "The probability of attack on workstation is greater than the probability of attack on webserver plus 0.05".

The experiments showed that the introduction of the uncertainty to the attack graph modeling and forecasting, on one hand, adds extra flexibility to the security administrator and may significantly reduce the attack graph, resulting in its better comprehensiveness. On the other hand, the definition of the probabilities and constraints is a complicated process and requires great expertise of the security administrator.

A set of European research projects devoted to the attacker's behaviour prediction as well as risk assessment utilized the approach based on analysis of the attack graphs, including TREsPASS (https://cordis.europa.eu/project/id/318003) (Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security) and MASSIF (MAnagement of Security information and events in Service InFrastructures) [47].

The TREsPASS project is interesting in that, when constructing an attack graph, the authors consider not only software exploits and configuration weaknesses, but also physical entities that could be used to gain access to the information resources. As the result, they developed the special attack navigator map tool, which allows uniting computer network entities and physical objects of the critical infrastructure, highlighting the fact that the attack may be implemented on both the networking level and the level of the physical objects. The forecasting of the malefactor actions considers the attacker's profiles presented in [48]. These profiles, known as threat agents, are based on eight attributes: intent, access, outcome, limits, resource, skill level, objective, and visibility.

To summarize, it is possible to say that attack graphs show every possible path that an attacker can use to gain further privileges—the path to be selected is determined by the attacker's skills as well as goals and motivation. In the general case, the attack graph complexity is $O(scn^2)$, for n machines in the attack graph, where s is the average number of exploits per machine and c is the average number of security conditions per machine. The survey of the graph-based techniques showed that the used attacker's model utilizes, in major cases, only two dimensions of the attacker's model—skills that could be defined explicitly or implicitly, and his/her location. Obviously, understanding the attacker's motivation and goal could significantly reduce the complexity of the attack graph and, as a result, increase the efficiency of the attack forecasting.

2.2. Attacker Behaviour Prediction Based on Hidden Markov Model

The Markov-based methods are very close to the attack tree models. In general, they are constructed on the basis of system states, and transitions between them, caused by events. Each transition is characterized by a probability that is independent of the past, and depends only on the two states involved—the behaviour of a process at a given point in time depends only on the state of the process at a previous point in time. The hidden Markov models (HMMs) for modeling normal behaviour to detect cyber attacks were first proposed in [29]. The authors used them to describe normal behaviour of the users as a sequence of the events and then applied them to detect insider threat. Since then, a significant amount of research has been done to enhance the HMM and its learning algorithm for detecting and predicting cyber attacks [30,31,33,34]. They vary in structure of HMM, used datasets, and particular tasks solved.

For example, in [30], the authors used HMMs to model and predict attack propagation based on data from different types of honeypots. In the research, they used data from the following families of honeypots:

- Glastopf (https://github.com/mushorg/glastopf)—a honeypot that emulates vulnerabilities that are relevant to web applications;
- Kippo (https://github.com/desaster/kippo)—a medium-interaction SSH honeypot;
- Honeytrap (https://www.honeynet.org/projects/active/honeytrap/)—a low-interaction honeypot that aims at collecting malware in an automated way;
- Dinoaea (https://www.div0.sg/single-post/dionaea-malware-honeypot)—malware capturing a honeypot that emulates several well-known protocols.

Thus, the authors managed to link different types of events in one model that is able to reveal trends in attack implementation and is able to detect abnormal attack sequences.

In [31], the authors applied a set of HMMs named as the fusion hidden Markov model. They construct k HMMs on k different low-correlated partitions of data and make a prediction using a nonlinear weight function. The latter is implemented by a neural network that is trained on the predictions of HMMs to the next state output. The application of k HMMs defines rather strict requirements to the HMMs; they have to be diverse and low correlated. To fulfill this requirement, the authors use a dissimilarity function to divide data into k different subsets, such that each subset contains a particular temporal pattern of the data. The input data are the real attack logs collected by the Cowrie honeypot [32], which is a medium-interaction SSH and telnet honeypot. The authors divided them into 19 groups corresponding to different activities, and these groups were modeled as states of the HMMs.

In [33], the continuous time Markov chain is used to make a prediction of the attack propagation.

It is clearly seen that this group of approaches does not use the attacker's model explicitly. The result of the prediction by the HMM strongly depends on the input dataset and the distribution of the events. The prediction of the attack goal is done on the basis of the most probable transition for the current system state, that is, the most frequently met sequence of the events. The skills of the attackers as well as motivation, available tools, and financial support are not considered.

In [34], the authors specify the attacker behaviour based on their goals, intention, and level of expertise, and outlined eight profiles of the attackers such as criminal groups, insiders, terrorists, hackers, phishers, nations, spyware/malware authors, and bot-net operators. However, the definition of the HMM presented in their approach did not consider the attacker's profile. The HMM is described as follows:

$$\lambda = (A, B, \pi, N),$$

where N corresponds to five different types of malicious behaviour (scanning, enumeration, access attempt, malware attempt, exploitation by denial of service), where

π is the state probabilities,

A is the transition probabilities, and

B is the observation probabilities.

Interestingly, the authors used the attacker's profiles for generating different training sets containing five types of malicious behaviour.

2.3. Attacker Behaviour Pattern Discovery Using Fuzzy Inference

The benefits of the fuzzy logic approaches consist in their ability to operate with uncertainty. We consider several works devoted to the intrusion detection based on fuzzy logic [45,49–51]. In major cases, fuzzy logic is applied to produce some averaged description of the parameters used to describe either normal or malicious activities. For example, in [50], the fuzzification process is applied to the metrics describing TCP service channel between two IP end-points—count, uniqueness, and variance. The authors defined five fuzzy sets for each metric: LOW, MEDIUM-LOW, MEDIUM, MEDIUM-HIGH, and HIGH, and defined the fuzzy set distributions using historical data. The authors applied fuzzy rules constructed as a combination of these parameters to determine the type of malicious activity, such as port scanning. In [51], fuzzy rules are constructed based on the results obtained by association rule mining. In [45], the authors applied leader-based k-means clustering to preprocess data before application of the fuzzification process. Thus, the existing approaches differ in preprocessing steps and data attributes to construct fuzzy rules for classifying the types of the malicious activities.

In [37], the authors solve the problem of constructing profiles of the normal user behaviour based on the analysis of the log events such as keyboard keys' sequences, characteristic data sequences retrieved from pointing device, chosen options, requested network resources, and so on. They apply fuzzy logic to the qualitative attributes of these events to describe a set of fuzzy profiles and identify masqueraded attacks.

In [46], an approach to combining attack graphs and fuzzy logic to predict attacker's behaviour was suggested. The attack graph is constructed in a traditional manner as a sequence of possible malefactor steps. Four parameters characterizing the attacker are assigned to each step: "the required knowledge for performing the attack action; (ii) the required access for conducting the attack action (the attack step may need physical access or it can be performed remotely); (iii) the required user interaction level for successful preformation of the attack (such as social engineering attacks against employees or the attacks targeting human-machine interface operators); and (iv) the required skill for conducting the attack" [46]. These parameters take the following values: low importance, moderate importance, importance, high importance, and very high importance. The fuzzy sets are described by triangular function. The complexity of the attack step depends on the values of these four variables. Apart of the assessment of the complexity of each attack step, the authors rate the alternatives existing for each attack step. This rating reflects the attractiveness of each step for the attacker and is evaluated on the basis of the expert's assessments. It is also a fuzzy variable that takes the following values: very low, moderate, high, and very high. To make a prediction of the attack deployment, the authors apply the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) approach, which is a multi-criteria decision making method suggested for fuzzy environment [52]. It allows the analyst to compare alternatives described by fuzzy variables. The general scheme of the approach is given in Figure 1 [46].

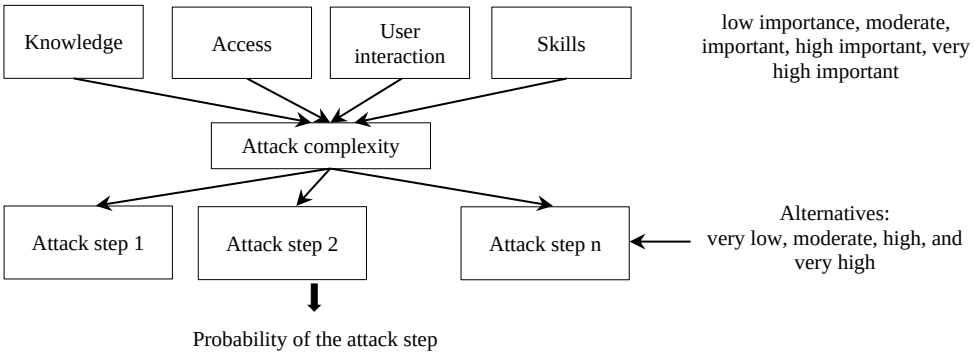

Figure 1. The overall scheme of the approach based on the combination of the attack graph and fuzzy logic.

Pricop and Mihalache [35] apply a fuzzy approach to model the impact of cyber attacks. Like in [46], the authors describe the attacker's profile as a combination of the following three parameters: knowledge, technical resources, and motivation—that is, a function of three inputs and one output. They define six types of attackers, as follows: script kiddie, hacker, disgruntled employee, terrorists, industrial spy, and cyber warrior. The script kiddie is an inexperienced and unskilled attacker that uses known exploits, and whose motivation is usually to get reputation, while the cyber warrior has the highest levels of knowledge, resources, and motivation. The cyber warrior is the most dangerous type of attacker, targeting the critical infrastructure.

The variables describing the attacker profile are linguistic variables that take values from fuzzy sets—very small, small, medium, big, and very big—which are presented by triangular curves. The highest score is assigned to the industrial spy; the cyber warrior, terrorist, and disgruntled employee have a medium score; the hacker's score is small; and the script kiddie has a very small score.

The attacker's profile, that is, the score [35], is used then to estimate the attack success rate. The impact of the attack is also a fuzzy function of four linguistic variables: the attacker profile (score), protection level, vulnerabilities, and restore cost. In the approach, these variables are described by a membership function of triangular form, defined for three fuzzy values—small, medium, and big. The attack success rate allows the analyst to understand how these parameters influence the overall security state of the information system.

In [36], the authors try to link attack steps to produce the attacker's profile. They developed a fuzzy inference system that takes as input the following linguistic variables: scanning/reconnaissance, enumeration, exploit by access attempt, exploit by denial of service, exploit by malware attempt, and output the attacker category. The possible attacker's categories are as follows: criminals, insiders, terrorists, hackers, phishers, nations, spyware/malware authors, bot-net operators, and amateur/script kids.

The linguistic variables used to determine the attacker's category may take the following fuzzy values: none, low, and high, which are described by a triangular form.

Thus, it is possible to conclude that there are two broad groups of approaches based on fuzzy logic to predict the attacker's behaviour.

The first group of techniques uses fuzzy inference to detect the type of the malicious activity, and the fuzzy rules describe generalized (fuzzy) dependencies between security event attributes. It is worth noticing that, in major cases, the authors apply fuzzy inference to detect attacks that have rather specific characteristics, such DoS attacks and port scanning. It could be explained that the most widely used data sets are NSL-KDD CUP 1999 and CAIDA UCSD "DDoS Attack 2007". These datasets do not contain complicated long-term attacks. They also do not consider attacks targeting IoT-based infrastructures, cyber-physical systems, "smart" homes, and so on.

The second group of techniques mostly focuses on risk assessment and uses the attacker's profile explicitly as an input variable that defines the success rate of the attack. The advantage of the application

of fuzzy logic is the ability to describe such fuzzy parameters as the motivation or knowledge of the malefactor. However, the major limitation of this group is the inability to link low level events to the attributes used to characterize the malefactor profile. The possible solution is to implement consequent mapping of low level events to middle level activities, and then determine the high level attributes of the attacker such as skills, resources, and motivation.

2.4. Attributing Cyber Attacks

In [38], the concept of attack attributing is used, that is, the determination of attack author, based on behavioural indicators. Behavioural indicators are combinations of actions and other indicators of malicious activity. These indicators can be atomic and computed. Atomic indicators are discrete pieces of data that cannot be broken down into their components without losing their forensic value. Atomic indicators include IP addresses, email addresses, domain names, and small pieces of text. Computed indicators are similarly discrete pieces of data, but they involve some element of computation. An example is a 'hash', a unique signature derived from input data, for instance, a password or a program. Hashes of programs running on their network's computers may match hashes of programs known to be malicious.

In some cases, behavioural indicators point to a specific adversary who has employed similar behaviours in the past. It might be repeated social engineering attempts of a specific style via email against low-level employees to gain a foothold in the network, followed by unauthorized remote desktop connections to other computers on the network delivering specific malware.

The authors outline that, though details are critical for attacker attributing, they will be useful only in the case of correct synthesis of information flows from the technical to the operational and strategic layers.

In [39], the authors build a cyber attacker model profile (CAMP) that can be used to characterize and predict cyber attacks. The authors define two types of variables used—dependable and independent. They denote the frequency and distribution of attacks as well as money earned from cybercrime as dependable variables (DVs), while unemployment rate, level of education, and corruption are independent variables. The authors constructed the attack prediction model linking both types of variables and showed how much variation in the DVs they can explain for given values of independent variables.

In [40], the attribution of honeypot data is considered. The authors define an attacker via a unique tuple (source IP address, operating system, user–agent (protocol), {cookies}). They assumed that the knowledge of the operating system, user agent, and set of cookies allows more accurate classification than the source IP address only. Honeypot data (HD) are used to calculate skill, resources, motivation, and intention. Further, they integrate skill (S) and resources (R) into the capability rating, and motivation (M) and intention (I) into the threat rating. Their combination is used to calculate the total threat score. S, R, M, and I are determined by weighted accumulation of all affecting features f_i:

$$V = \sum_{i=1}^{n} a_i f_i,$$

where n is the total number of features f_i;

a_i is the weight for the i-th feature, $\sum_{i=1}^{n} a_i = 1$.

The features f_i are derived from the considered observed HD features v_i and get values of $\{0, \ldots, \gamma\} \in \mathbb{Q}$. The maximal value of S, R, M, and I is γ. The dimension and boundaries for v_i vary between the parameter and sensor resolution. The part of sample feature set provided by the authors is represented in Table 2.

Table 2. Part of sample feature set for attackers' classification suggested in [40].

Origin									**Temporal**	
Port		IP address		User agent	URL	Domain	E-mail	User ID	OS	Inter-Arrival Time	
Protocol	Service	Autonomous system	Country							Standard deviation	Average

For example, to calculate R, the assumption can be made that fast inter-arrival times are related to a higher degree of automation (higher attackers' resources). The motivation attribute can be estimated by the time and effort an attacker invests into a particular attack. Quantifying an attacker's intention is the most complex task. The authors define intention as the degree or potential of attacker's maliciousness.

The authors use the following classes of attackers [40]: guest (G), external employee (E), internal employee (I), activists (A), state-sponsored (N), ethical hacker (W), criminals (O), cracker (C), and hobby hacker (H).

The values for different classes are calculated using $V \in$ {S,R,I,M}, which are ordered as $V_{ci} < V_{cj} \ldots < V_{cn}$, $\forall\, c \in C$, and then transformed to $\{0, \ldots, \gamma\} \in \mathbb{Q}$, by assigning 1 to the first class and iterating over all classes while incrementing the value by 1 for each less-than operator. Then, all values are normalized with $\gamma = 10$. In Figure 2, the heat map proposed by the authors to represent attackers' classes is provided (the capitalized abbreviation marks the appropriate class).

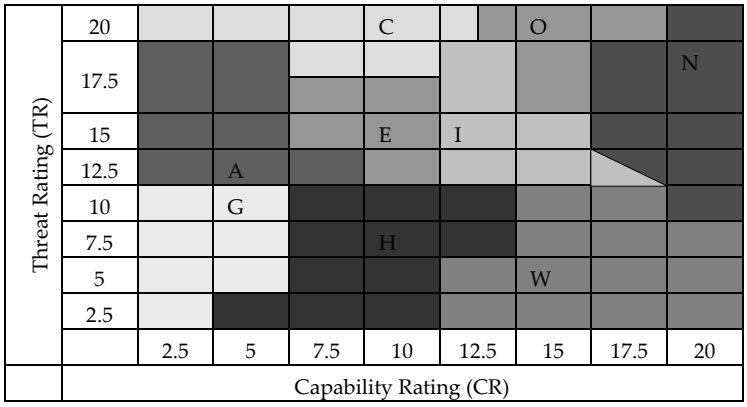

Figure 2. Attackers' classes [40].

Though the method is introduced for IT-Security in Industry 4.0, nonetheless, the specific features of CPS are not considered.

In [41], the authors propose the method for predicting the behaviour of cyberattacks using recurrent neural networks (RNNs). They use the dataset obtained from the 2017 Collegiate Penetration Testing Competition (CPTC) to obtain long-short-term-memory (LSTM) models. The dataset includes Suricata alerts obtained, while ten student teams attempted to penetrate the virtualized network and exploit vulnerabilities. The authors trained two sets of models: the first set determines the team that caused the alert, and the second predicts the second alert. The used features are as follows: destination port, alert signature, alert category, alert severity, proto, source port, and host. The authors achieved accuracy of 55% for teams classification and 80% for the next alerts prediction.

Finally, while the last works on the attacker behaviour forecasting using machine learning make attempts to overcome the challenge of linking raw data with valuable attacker metrics, the feature set is still not specified, the set of metrics that forms the attacker profile is not unified, the techniques of metrics calculation on the basis of the extracted features should be enhanced, and the training dataset problem still exists.

3. Challenges, Possible Solutions, and Common Approaches

The analysis conducted allowed us to conclude on the main challenges existing in attack goal forecasting to this moment:

1. Lack of uniformity in the classification of attackers, distinguished metrics, and attributes, as well as the definition of the same classes and metrics.
2. The gap between the raw data (such as network traffic and events logs), attacker profile, and forecasting of the attacker behaviour, as well as the methods for the determination of relationships between them.
3. Lack of datasets suitable for research of the relationships between attacker steps and his/her goals.
4. Absence of the research that demonstrates if there is an influence of the attacker profiling and attributing on the attack forecasting.

The lack of uniformity indicates insufficient elaboration of the problem under research. Besides, it prevents efficient countermeasure selection for different classes of attackers, understanding of current research state, comparative quantitative analysis of the various developed techniques, and elaboration of the existing results. An attempt to overcome this challenge was made in [9]. However, the authors do not describe how to link low level events with high level events, that is, they did not proposed a solution to the second problem.

Considering the second challenge, an attempt to link low level events with security metrics was made in the Structured Threat Information Expression (STIX) project [53]. Structured Threat Information Expression (STIX) is a structured language for specification of various threats and automated analysis. The idea behind the development of this language is to link low level events with high level concepts. The following components of the language are specified [54]: observables; indicators (observation patterns and their meanings); incidents (attack actions instances); adversary tactics, techniques, and procedures (methods that are used by an attacker, including attack patterns, malware, exploits, and so on); exploit targets (e.g., vulnerabilities, weaknesses, and configurations); courses of action (response actions to prevent an attack); campaigns (sets of incidents and/or TTPs with a single goal); threat actors (attacker identification); and reports.

For each component, the set of properties is specified. For example, for threat actors, the following properties are used: name, description, aliases, roles, goals, sophistication, resource_level, primary_motivation, secondary_motivations, and personal_motivations.

All properties are of a nominal type (i.e., values are selected from a list). Thus, for threat actor labels, the possible values are as follows: activist, competitor, crime-syndicate, criminal, hacker, insider-accidental, insider-disgruntled, nation-state, sensationalist, spy, and terrorist. While for the threat actor sophistication (captures the skill level of a threat actor; ranges from "none", which describes a complete novice, to "strategic", which describes an attacker who is able to influence supply chains to introduce vulnerabilities), the values are as follows: none, minimal, intermediate, advanced, expert, innovator, and strategic. In this project, however, how to determine the values of these properties automatically from the raw data is not also described. It should be actively used by the security companies in order to reveal and then automate the process of linking low events and high level attack concepts; however, there is not much activity in this field.

The second challenge is connected with the third challenge, that is, the absence of datasets for analysis aimed at revealing existing interrelations and features characterizing attackers and their goals. The following approaches are used to overcome it:

- Use existing datasets with specific attacks' data.
- Use honeypots to generate real data.
- Use normal data and add data on attacks intentionally (use attack generators).

The first approach is used for the detection of specific types of attacks based on training using the datasets. However, the most used datasets are deprecated and do not represent the last trends in attacks or paradigm of the modern information systems.

According to the second approach, in [30], the authors used the honeypot technology. The detailed description of attack features logged and dataset description, when using the honeypot technology, is provided in [55]. The analysis is based on the following assumption: the data are grouped by session ID for considering that the attacker attempts to implement some malicious scenario in one session, that is, different session IDs are independent of individual attacker characteristics. This allows the authors to group event sequences to create a training sample by sessions. However, this approach does not consider an opportunity to use several sessions to implement a complex multistep attack by a single attacker.

The researchers usually create their own datasets and use them [56]. Unfortunately, however, in most of these approaches, all these data are not annotated by attackers, that is, their skills, knowledge, and other characteristics that form their profiles. In fact, datasets contain only attacks of different types, and there are no labeled datasets characterizing attackers' skills. This is explained by the fact that the techniques used to detect attacks analyze the event sequences, their frequencies, and attributes. Until there is no research proving that the application of the attacker attribution may enhance the efficiency of the attack detection, there will be no datasets linking raw security events with attacker's profile concepts such as attacker motivation, goal, and so on. However, having such a dataset maybe extremely useful in detecting targeted and distributed in time attacks. Unfortunately, the absence of datasets is a common problem that can be solved with their targeted generation.

Thus, we argue that there is a need in the research that demonstrates if there is an influence of the attacker profiling and attributing on the attack forecasting. Thus, the fourth challenge is one of our future research directions. However, it is necessary to overcome the first three challenges first. In particular, we are planning start with the generation of the specific dataset. We consider that the approach presented in [57] is the promising one to generate datasets for attack attribution. It is based on mixed traffic generation, including attacks and normal traffic.

To conclude, we propose the following approach to the attacker behaviour forecasting:

1. First of all, we suppose to outline possible raw data sources. There are two types of sources: structured data and unspecified data. In [58], we outlined the following open sources of structured data considering objects of information security assessments: vulnerability databases, attack patterns databases, weaknesses databases, software and hardware databases, and so on. For accurate attack forecasting in real time, it is required to add another type of source data, network traffic, and event logs (which is unspecified). From the analyzed events datasets, the most interesting is the one provided in [56]. The dataset should contain data on various attacks with different goals implemented by attackers of different classes. From our point of view, the most complete classification from those reviewed was proposed in [40]. It incorporates the following classes: guest, external employee, internal employee, activists, state-sponsored, ethical hacker, criminals, cracker, and hobby hacker.
2. Extract features from the events dataset that can characterize different classes of attackers with different goals. While there are rather detailed sets of features from the network traffic (such as source IP address, operating system, user–agent (protocol), and {cookies} in [40]), the events features should be researched in more detail. In [41], the following set is proposed: destination port, alert signature, alert category, alert severity, proto, source port, and host. We can use this as the basis for future research.
3. Then, we suppose to outline and classify high level metrics that form the attacker profile, on the basis of the following metrics, proposed in [59]: attacker skill level, attacker knowledge, tools complexity, attack steps complexity, steps success rate, trace coverage rate, and so on.
4. Then, we propose to find out structural and semantic relations between data sources, objects of the attacker behaviour forecasting subject area, and metrics (from features extracted from the raw

data to high level metrics of attackers and attacks). To implement these, we plan to extend an ontology provided in [59] and determine transitional metrics.

5. Then, we propose to use the outlined characteristics and relationships to do the following:

    a. develop algorithms for metrics calculation;
    b. train a neuro-fuzzy network for attackers' behaviour forecasting.

We state that steps 1–4 are the necessary basis for step 5, while overcoming challenges 1–4 is the basis for the successful implementation of our research task.

Thus, at this stage, we developed the common approach to forecasting attacker goals and considered the future work scope on the basis of comparative analysis of the related research and existing challenges in the area.

4. Conclusions

In the paper, we reviewed the research in the area of attacker behaviour forecasting. Compared with the close survey described in [4], our research is focused on issues of the attacker profile definition or attacker attribution and its influence on the attack forecasting process. In [9], an interesting study that highlights main challenges in the area of attacker behaviour forecasting is provided and the multilevel system of metrics is introduced. Our goal in this research, however, is to determine how to link low level events with high level events. Besides, compared with the aforementioned papers, the main goal of the research outlined in the paper is the novel approach development. Though the main goal of the research outlined in the paper is not devoted only to the state-of-the-art, it is necessary for novel approach development. In the scope of our research, we considered four classes of approaches to the attacker behaviour forecasting, including attack graph based approach, HMM, fuzzy inference, and approaches based on intelligent data processing. The analysis shows that there is a lack of formalization and systematic representation of the attacker profile and of the definition of his/her characteristics that can be used for his/her specification. From our point of view, the most promising are approaches based on intelligent data analysis, as soon as they allow linking raw data and metrics describing an attacker.

The conducted analysis allowed us to outline the key challenges in the area. On the basis of these challenges and our task, we have selected the approach to the task implementation. The proposed approach specifies our further research steps and is the basis for the technique of attacker behaviour and goals forecasting under development.

The approach incorporates the following steps: (1) outline raw data sources, both structured and unspecified; (2) extract features from the events dataset that characterize different classes of attackers with different goals; (3) outline and classify high level metrics that form attacker profile; (4) find out structural and semantic relations between data sources, objects of the attacker behaviour forecasting subject area, and metrics (from features extracted from the raw data to high level metrics of attackers and attacks); and (5) use the outlined characteristics and relationships to develop algorithms for metrics calculation, and to train neuro-fuzzy network for attackers' behaviour forecasting. Compared with the other approaches, summarized in this paper, our approach is focused on the accurate determination of relations among raw data and attacker behaviour characteristics. Each step of the proposed approach will be discussed in detail in the following research. Moreover, in the scope of our future research, we will analyze if there is the influence of the attacker profiling and attributing on the attack forecasting.

Author Contributions: Investigation, E.D. and E.N.; literature analysis, E.D. and E.N.; common approach, E.D. and E.N.; writing—original draft, E.D. and E.N.; writing—review and editing, E.D., E.N. and I.K. All authors have read and agree to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Howard, J.D.; Longstaff, T.A. *A Common Language for Computer Security Incidents*; Sandia National Labs.: Albuquerque, NM, USA; Livermore, CA, USA, 1998.
2. Abomhara, M.; Køien, G.M. Cyber security and the Internet of Things: Vulnerabilities, threats, intruders and attacks. *J. Cyber Secur. Mob.* **2015**, *4*, 65–88. [CrossRef]
3. Aliyev, V. Using Honeypots to Study Skill Level of Attackers Based on the Exploited Vulnerabilities in the Network. Ph.D. Thesis, Chalmers University of technology, Göteborg, Sweden, 2010.
4. Gheyas, I.A.; Abdallah, A.E. Detection and prediction of insider threats to cyber security: A systematic literature review and metaanalysis. *Big Data Anal.* **2016**, *1*, 6. [CrossRef]
5. Husák, M.; Komárková, J.; Bou-Harb, E.; Čeleda, P. Survey of Attack Projection, Prediction, and Forecasting in Cyber Security. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 640–660. [CrossRef]
6. Yang, S.J.; Du, H.; Holsopple, J.; Sudit, M. Attack Projection. In *Cyber Defense and Situational Awareness*; Springer: Cham, Switzerland, 2014; pp. 239–261.
7. Ahmed, A.A.; Zaman, N.A.K. Attack intention recognition: A review. *IJ Netw. Secur.* **2017**, 244–250.
8. Abdlhamed, M.; Kifayat, K.; Shi, Q.; Hurst, W. Intrusion Prediction Systems. In *Information Fusion for Cyber-Security Analytics*; Springer: Cham, Switzerland, 2017; pp. 155–174.
9. Rocchetto, M.; Tippenhauer, N.O. On Attacker Models and Profiles for Cyber-Physical Systems. In *Lecture Notes in Computer Science, Proceedings of the ESORICS, 2016*; Askoxylakis, I., Ioannidis, S., Katsikas, S., Meadows, C., Eds.; Springer: Cham, Switzerland, 2016.
10. Corman, J.; Etue, D. Adversary ROI: Evaluating Security from the Threat Actor's Perspective. In Proceedings of the RSA Conference Europe 2012, San Francisco, CA, USA, 9–11 October 2012.
11. Heckman, R.M. Attacker Classification to Aid Targeting Critical Systems for Threat Modelling and Security Review. 2005. Available online: www.rockyh.net/papers/AttackerClassification.pdf (accessed on 22 January 2020).
12. Cardenas, A.A.; Amin, S.M.; Sinopoli, B.; Giani, A.; Perrig, A.; Sastry, S.S. Challenges for Securing Cyber Physical Systems. In *Workshop on Future Directions in Cyber-physical Systems Security*; DHS: Newark, NJ, USA, 2009.
13. LeMay, E.; Ford, M.D.; Keefe, K.; Sanders, W.H.; Muehrcke, C. Model-Based Security Metrics Using Adversary View Security Evaluation (ADVISE). In Proceedings of the 2011 Eighth International Conference on Quantitative Evaluation of Systems, Aachen, Germany, 5–8 September 2011.
14. Cardenas, A.A.; Roosta, T.; Sastry, S. Rethinking security properties, threat models, and the design space in sensor networks: A case study in SCADA systems. *Ad Hoc Netw.* **2009**, *7*, 1434–1447. [CrossRef]
15. Matusitz, J. Cyberterrorism: Postmodern state of chaos. *Inf. Secur. J.* **2008**, *17*, 179–187. [CrossRef]
16. Denning, D.E. Activism, hacktivism, and cyberterrorism: The internet as a tool for influencing foreign policy. *Netw. Netwars Future Terror Crime Militancy* **2001**, *239*, 288.
17. Schneier, B. Attack Trees—Modeling Security Threats. *Dr. Dobb's J.* **1999**, *24*, 12.
18. Hariri, S.; Qu, G.; Dharmagadda, T.; Ramkishore, M.; Raghavendra, C.S. Impact Analysis of Faults and Attacks in Large-Scale Networks. *IEEE Secur. Priv.* **2003**, *1*, 49–54. [CrossRef]
19. Ingols, K.; Chu, M.; Lippmann, R.; Webster, S.; Boyer, S. Modeling Modern Network Attacks and Countermeasures Using Attack Graphs. In Proceedings of the 2009 Annual Computer Security Applications Conference (ACSAC'09), Honolulu, HI, USA, 7–11 December 2009.
20. Kheir, N.; Cuppens-Boulahia, N.; Cuppens, F.; Debar, H. A Service Dependency Model for Cost-Sensitive Intrusion Response. In Proceedings of the 15th European Symposium on Research in Computer Security (ESORICS), Athens, Greece, 20–22 September 2010.
21. Kotenko, I.; Stepashkin, M. Attack Graph based Evaluation of Network Security. *Lect. Notes Comput. Sci.* **2006**, *4237*, 216–227.
22. Kotenko, I.; Stepashkin, M.; Doynikova, E. Security Analysis of Computer-aided Systems Taking into Account Social Engineering Attacks. In Proceedings of the 19th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP 2011), Los Alamitos, CA, USA, 9–11 February 2011; pp. 611–618.
23. Noel, S.; Jajodia, S.; O'Berry, B.; Jacobs, M. Efficient minimum-cost network hardening via exploit dependency graphs. In Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC'03), Las Vegas, NV, USA, 8–12 December 2003.

24. Wang, L.; Jajodia, S.; Singhal, A.; Noel, S. k-Zero Day Safety: Measuring the Security Risk of Networks against Unknown Attacks. In Proceedings of the 15th European Conference on Research in Computer Security; Springer: Berlin/Heidelberg, Germany, 2010; pp. 573–587.
25. GhasemiGol, M.; Ghaemi-Bafghi, A.; Takabi, H. A comprehensive approach for network attack forecasting. *Comput. Secur.* **2016**, *58*, 83–105. [CrossRef]
26. Wang, L.; Islam, T.; Long, T.; Singhal, A.; Jajodia, S. An Attack Graph-Based Probabilistic Security Metric. In *Lecture Notes in Computer Science 5094, Proceedings of the Data and Applications Security XXII (DBSec 2008)*; Atluri, V., Ed.; Springer: Berlin/Heidelberg, Germany, 2008.
27. Kotenko, I.; Doynikova, E. Improvement of attack graphs for cybersecurity monitoring: Handling of inaccuracies, processing of cycles, mapping of incidents and automatic countermeasure selection. *SPIIRAS Proc.* **2018**, *57*, 211–240.
28. An, S.; Eom, T.; Park, J.S.; Hong, J.B.; Nhlabatsi, A.; Fetais, N.; Khan, K.M.; Kim, D.S. CloudSafe: A Tool for an Automated Security Analysis for Cloud Computing. Available online: https://arxiv.org/abs/1903.04271v1 (accessed on 25 January 2020).
29. Rashid, T.; Agrafiotis, I.; Nurse, J.R.C. A New Take on Detecting Insider Threats: Exploring the Use of Hidden Markov Mode. In Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats, Vienna, Austria, 28 October 2016; ACM: New York, NY, USA, 2016; pp. 47–56.
30. Bar, A.; Shapira, B.; Rokach, L.; Unger, M. Identifying Attack Propagation Patterns in Honeypots Using Markov Chains Modeling and Complex Networks Analysis. In Proceedings of the IEEE International Conference on Software Science, Technology and Engineering (SWSTE), Beer Sheva, Israel, 23–24 June 2016; pp. 28–36.
31. Deshmukh, S.; Rade, R.; Kazi, F. Attacker Behaviour Profiling Using Stochastic Ensemble of Hidden Markov Models. 2019. Available online: https://arxiv.org/abs/1905.11824 (accessed on 25 January 2020).
32. Oosterhof, G.M. Cowrie—Medium-Interaction Honeypot. Available online: https://github.com/micheloosterhof/cowrie (accessed on 25 January 2020).
33. Jhawar, R.; Lounis, K.; Mauw, S. A Stochastic Framework for Quantitative Analysis of Attack-Defense Trees. In *Lecture Notes in Computer Science 9871, Proceedings of the Security and Trust Management (STM 2016)*; Barthe, G., Markatos, E., Samarati, P., Eds.; Springer: Cham, Switzerland, 2016.
34. Katipally, R.; Yang, L.; Liu, A. Attacker Behavior Analysis in Multi-stage Attack Detection System. In Proceedings of the Cyber Security and Information Intelligence Research (CSIIRW'11), Oak Ridge, TN, USA, 12–14 October 2011; ACM: New York, NY, USA, 2011. Available online: https://www.utc.edu/center-academic-excellence-cyber-defense/pdfs/paper-csiirw-2011-attacker-behavior.pdf (accessed on 25 January 2020).
35. Pricop, E.; Mihalache, S.F. Fuzzy approach on modelling cyber attacks patterns on data transfer in industrial control systems. In Proceedings of the 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI 2015), Bucharest, Romania, 25–27 June 2015.
36. Mallikarjunan, K.N.; Shalinie, S.M.; Preetha, G. Real Time Attacker Behavior Pattern Discovery and Profiling Using Fuzzy Rules. *J. Internet Technol.* **2018**, *19*, 1567–1575.
37. Kudłacik, P.; Porwik, P.; Wesołowski, T. Fuzzy approach for intrusion detection based on user's commands. *Soft Comput.* **2016**, *20*, 10–16. [CrossRef]
38. Rid, T.; Buchanan, B. Attributing Cyber Attacks. *J. Strateg. Stud.* **2015**, *38*, 4–37. [CrossRef]
39. Watters, P.A.; McCombie, S.; Layton, R.; Pieprzyk, J. Characterising and Predicting Cyber Attacks Using the Cyber Attacker Model Profile (CAMP). *J. Money Laund. Control* **2012**, *15*, 430–441. [CrossRef]
40. Fraunholz, D.; Krohmer, D.; Duque Antón, S.; Schotten, H.D. YAAS—On the Attribution of Honeypot Data. *Int. J. Cyber Situat. Aware.* **2017**, *2*, 31–48. [CrossRef]
41. Perry, I.; Li, L.; Sweet, C.; Su, S.H.; Cheng, F.-Y.; Yang, S.J.; Okutan, A. Differentiating and Predicting Cyberattack Behaviors Using LSTM. In Proceedings of the 2018 IEEE Conference on Dependable and Secure Computing (DSC), Kaohsiung, Taiwan, 10–13 December 2018.
42. Çayirci, E.; Rong, C. *Security in Wireless Ad Hoc and Sensor Networks*; John Wiley & Sons: Hoboken, NJ, USA, 2009.
43. Kdd Cup 1999 Data. UCI KDD Archive. 1999. Available online: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (accessed on 25 January 2020).

44. The CAIDA DDoS Attack 2007 Dataset. 2007. Available online: http://www.caida.org/data/passive/ddos-20070804_dataset.xml (accessed on 25 January 2020).
45. Shyla, S.; Sujatha, S. Cloud Security: LKM and Optimal Fuzzy System for Intrusion Detection in Cloud Environment. *J. Intell. Syst.* **2019**, *29*, 1626–1642. [CrossRef]
46. Orojloo, H.; Abdollahi Azgomi, M. Predicting the behavior of attackers and the consequences of attacks against cyber-physical systems. *Secur. Commun. Netw.* **2016**, *9*, 6111–6136. [CrossRef]
47. MASSIF FP7 Project. MASSIF Architecture. 2011–2013. Available online: https://rieke.link/MASSIF_Architecture_document.pdf (accessed on 22 January 2020).
48. Casey, T. Threat Agent Library Helps Identify Information Security Risks. Intel. Technical Report. 2007. Available online: https://www.sbs.ox.ac.uk/cybersecurity-capacity/system/files/Intel%20-%20Threat%20Agent%20Library%20Helps%20Identify%20Information%20Security%20Risks.pdf (accessed on 22 January 2020).
49. Shanmugam, B.; Idris, N.B. Hybrid Intrusion Detection Systems (HIDS) Using Fuzzy Logic. 2011. Available online: https://www.intechopen.com/books/intrusion-detection-systems/hybrid-intrusion-detection-systems-hids-using-fuzzy-logic (accessed on 22 January 2020).
50. Dickerson, J.E.; Dickerson, J.A. Fuzzy Network Profiling for Intrusion Detection. In Proceedings of the 19th International Conference of the North American Fuzzy Information Processing Society—NAFIPS (Cat. No.00TH8500), PeachFuzz 2000, Atlanta, GA, USA, 13–15 July 2000; pp. 301–306.
51. Shanmugam, B.; Idris, N.B. Improved Intrusion Detection System Using Fuzzy Logic for Detecting Anamoly and Misuse Type of Attacks. In Proceedings of the 2009 International Conference of Soft Computing and Pattern Recognition, Malacca, Malaysia, 4–7 December 2009; pp. 212–217.
52. Chen, C.T. Extensions of the TOPSIS for group decision-making under fuzzy environment. *Fuzzy Sets Syst.* **2000**, *114*, 1–9. [CrossRef]
53. Structured Threat Information eXpression (STIX™) 1.x Archive Website. Available online: https://stixproject.github.io/ (accessed on 25 January 2020).
54. About STIX. Available online: https://stixproject.github.io/about/ (accessed on 25 January 2020).
55. Rade, R.; Deshmukh, S.; Nene, R.; Wadekar, A.S.; Unny, A. Temporal and Stochastic Modelling of Attacker Behaviour. In *Advances in Data Science 941, Proceedings of the Communications in Computer and Information Science (ICIIT 2018)*; Akoglu, L., Ferrara, E., Deivamani, M., Baeza-Yates, R., Yogesh, P., Eds.; Springer: Singapore, 2019.
56. Singapore University of Technology and Design Official Web Site. iTrust. Dataset Characteristics. Available online: https://itrust.sutd.edu.sg/itrust-labs_datasets/dataset_info/ (accessed on 25 January 2020).
57. Kotenko, I.; Chechulin, A.; Branitskiy, A. Generation of Source Data for Experiments with Network Attack Detection Software. *J. Phys. Conf. Ser.* **2017**, *820*, 012033. [CrossRef]
58. Kotenko, I.; Fedorchenko, A.; Doynikova, E.; Chechulin, A. An Ontology-based Hybrid Storage of Security Information. *Inf. Technol. Control* **2018**, *47*, 655–667.
59. Doynikova, E.; Kotenko, I. Approach for determination of cyber attack goals based on the ontology of security metrics. In *IOP Conference Series: Materials Science and Engineering (MSE) 450, Proceedings of the International Workshop "Advanced Technologies in Aerospace, Mechanical and Automation Engineering" (MIST: Aerospace-2018), Krasnoyarsk, Russia, 20 October 2018*; IOP Publishing: Bristol, UK, 2018.