Concept and Practical Evaluation for Adaptive and Intelligible Prioritization for Network Security Incidents

Leonard Renners[*], Felix Heine[*], Carsten Kleiner[*], Gabi Dreo Rodosek[†]

[*]*University of Applied Sciences and Arts, Hannover.*
[†]*Universitaet der Bundeswehr Muenchen, Neubiberg.*

ABSTRACT

Incident prioritization is nowadays a part of many approaches and tools for network security and risk management. However, the dynamic nature of the problem domain is often unaccounted for. That is, the prioritization is typically based on a set of static calculations, which are rarely adjusted. As a result, incidents are incorrectly prioritized, leading to an increased and misplaced effort in the incident response. A higher degree of automation could help to address this problem. In this paper, we explicitly consider flaws in the prioritization an unalterable circumstance. We propose an adaptive incident prioritization, which allows to automate certain tasks for the prioritization model management in order to continuously assess and improve a prioritization model. At the same time, we acknowledge the human analyst as the focal point and propose to keep the human in the loop, among others by treating understandability as a crucial requirement.

Keywords: *Incident Prioritization, Network Security, Cyber Security, Adaptive Learning.*

Published online: 2nd December 2019

1 INTRODUCTION

An important part of the analysis of network security incidents is to estimate a priority, describing the importance of the situation and determining the order and magnitude of the incident response. This prioritization becomes necessary due to the fact that the number of incidents outweighs a limited working power. In the worst case, some incidents are continually left unaddressed, but in any case the response time can be delayed by the handling of other incidents. Therefore, the limited time of the analysts needs to be assigned cautiously. Although prioritization itself is nowadays a part of many approaches and tools towards incident detection and risk assessment exist, the dynamic nature of the problem domain is often unaccounted for.

The prioritization is typically based on a set of static calculations, which are rarely adjusted. Oftentimes, there is no explicit process to identify errors and even then, improvements are made and evaluated manually on a best guess basis. The problem can be again attributed to the necessary manual effort, which is already a problem in the application domain in general. For one, reports still show an increasing number of vulnerabilities, threats and cyber incidents on the whole (IBM, 2017) At the same time, the available human domain experts already depict the bottleneck regarding these issues (Bhatt, Manadhata, & Zomlot, 2014). Incorrect prioritization further aggravates this problem, as errors in the rating lead, among other things, to false positives and respective misplaced efforts. In fact, false positives can even result in mistrust in the system, which leads to potential harmful incidents being ignored, as they are assumed to be false alarms as well (FireEye, 2014).

Moreover, the number of devices to be monitored and secured is also continuously increasing at a rate larger than the human resources, even further aggravating the problem of accurate and timely incident prioritization. Unfortunately, the automation of respective tasks within tools for network security, and most importantly Security Information and Event Management (SIEM) systems, is still lackluster and the prioritization has been identified as one of the major challenges in achieving SIEM optimization (Ponemon Institute LLC, 2017).

In this paper, we target these problems explicitly. That is, we focus on different aspects of automation and assistance for tasks related to the incident prioritization. In turn, this results in more accurately prioritized incidents and an improved distribution of the limited working power for the incident response. However, we also believe that the human resource has to be the focal point of network security. Therefore, we argue for an approach

that adds intelligibility as a crucial requirement for any effort towards an increased automation.

Contribution

This paper presents concepts for an adaptive and intelligible prioritization of network security incidents. In particular, we integrate and propose the combined use of individually presented additions to the incident prioritization process. We specify an approach to define and learn prioritization rules, collect feedback data and generate as well as evaluate adaptations for existing calculation directives. For the entire process, comprehensibility is considered an important requirement. Thus, our approach introduces an increased degree of automation in this domain, while aiming to keep the prioritization as well as the adaptation intelligible.

Outline

The remainder of this paper is structured as follows: First, related work is covered in Section 2. In Section 3, we discuss requirements and introduce our process and conceptual models for an adaptive and intelligible incident prioritization. Section 4 then presents an overview of the evaluation of the main components from the presented approach. Future directions of research are identified and discussed in the subsequent Section 5. The final Section 6 provides a conclusion and reviews our ideas and contribution.

2 RELATED WORK

The following Table 1 summarizes the areas of related work with regard to the two major requirements outlined in the introduction: (1) incident prioritization with (2) a higher degree of automation (learning/adaptation). Within the table, a full circle (●) corresponds to a fulfilled requirement, whereas the semi-filled circle (◑) indicates a partial fulfilment and the empty circle (○) depicts an unaddressed requirement. We thereby show open topics and underline the contributions of this work, as we target both aspects explicitly.

Table 1: Overview of areas of related work.

Area	Prioritization	Automation (Learning/Adaptation)
SIEM	●	○
Risk assessment	●	○
IDS	◑	◑
Outlier and anomaly detection	◑	●
Organizational (human) learning	◑	◑
Multi-objective optimization	○	●

Approaches that address the prioritization of network security incidents do not yet explicitly account for the dynamic aspect of the domain in terms of an automated learning or adaptation. In contrast, advances towards more automated processes by implementing machine learning techniques for the learning and adaptation do not target incident prioritization. The most closely related approaches still target different outcomes (primarily classification) and thus underly other requirements, which lead to distinct results. Individual approaches from the areas shown in Table 1 will be shortly discussed in the remainder of this section in the same order as displayed in the overview.

Generally, the realm of commercial products and especially SIEM systems pose an area of related work, as the goal of incident prioritization has been established within these systems alongside different realizations. However, most of the systems are based on static calculation formula, which sometimes refer to user-defined or manually changeable prioritization concepts. For example, QRadar defines an event magnitude on the basis of three sub-priorities. Each of those is influenced by rules within the event processing, which add or subtract scores based on specific event properties. In contrast, ArcSight employs one particular calculation formula. However, this formula uses external concepts, which again can be influenced by the SIEM configuration. In that sense, both systems operate on similar terms and use a configurable, but fixed, priority calculation. And neither offers methods to learn or adapt the prioritization model or configuration automatically.

From the research community, similar approaches for risk assessment and threat evaluation frameworks have been proposed to calculate the risk and thereby priority of alert and incident data. Townsend and McAllister (Townsend & McAllister, 2013) provide a prioritization framework in which they define the priority, called threat, of an incident as a combination of likelihood, impact and risk. They further describe each aspect in an abstract and textual description. Kim et al. (Kim, Kang, Luo, & Velasquez, 2014) propose a similar approach, but add specific, quantitative measurements for each factor and concrete calculation formulas. Yet again, these priorities and calculations are based on the static framework definitions.

Within the research community, alert and incident detection, prioritization and response has in general gained a lot of attention in recent years. Multiple approaches have been proposed towards data integration and correlation, but also regarding automatically induced models. However,

most progress has been within, and limited to, the intrusion detection domain. For this area of work, we would like to refer to the different survey papers, e.g. the taxonomies presented by Axelsson (Axelsson, 2000) or Debar et al. (Debar, Dacier, & Wespi, 2000) and, with more focus on the computational intelligence, the review by Wu and Banzhaf (Wu & Banzhaf, 2010). Here, the discussion will primarily concern prioritization and the integration of human analysts, especially towards adaptation.

Interesting work has been proposed by Veeramachaneni et. al. (Veeramachaneni, Arnaldo, Korrapati, Bassias, & Li, 2016) as they present an analyst-in-the-loop system for big data analytics. Their framework is composed of four building blocks: big data behavioral analytics, outlier detection, feedback and supervised learning. They describe an application of their system, combining supervised and unsupervised learning techniques, by which major improvements of the performance of the outlier detection are achieved. With respect to our concepts, the most important aspect of the framework is the feedback mechanism and its use in model improvement. However, since the approach focuses on outlier detection, the feedback consists of the classification labels, which in turn are used to learn a supervised model to be used in conjunction with the unsupervised model from the big data behavioral analytics. This is greatly different from our approach, as we propose an incorporation of feedback in an already existing model, which originated possibly, but not necessarily, from supervised learning. Furthermore, their feedback and learning refer to the classification, which poses different challenges and requirements than the prioritization.

Das et. al (Das, Wong, Dietterich, Fern, & Emmott, 2016) proposed a feedback improved anomaly detection system. They provide an approach to employ an unsupervised anomaly detection and build a supervised weighing for an accuracy on top to decrease the ranking of similar instances of nominal data. In these aspects, their work is rather similar to the approach of Veeramachaneni et. al. as they also apply supervised learning on top of an unsupervised model. Thus, our work also differs regarding the classification in contrast to prioritization, as well as concerning the multiple objective functions used within our approach. However, their work, similar to our approach, focuses on the issue, which events to show to the analyst to gather feedback. Das et. al. assume that the highest scored anomalies should be shown to the analyst, which neglects false negatives. We explicitly target this type of error as an application scenario and consider concepts to deal with resulting requirements.

Another approach was proposed by Ben-Asher and Yu (Ben-Asher & Yu, 2017), which follows a similar line of thought. They describe a so-called synergistic architecture for human-machine intrusion detection. This architecture is composed of three primary building blocks: the analyst, data collection and a detection engine. The analyst is supposed to actively interact with the other two components. As a result, an improvement in the automated tasks is achieved, whereas the automation helps in reducing the workload of the analyst, leading towards a better control of the detection instead of the alert evaluation. The general idea and especially the motivation for automation and human interaction align well with our approach, but this work differs in the architecture and its application scenario. We consider a SIEM environment (which is on a higher level of abstraction) and highlight the prioritization aspect. Therefore, we develop more specific means to decide where interaction (i.e. feedback) is needed and how it can be incorporated to improve the prioritization model.

Work in a different, but also interesting direction was introduced by Shedden et. al. (Shedden, Ahmad, & Ruighaver, 2010). They propose a learning focused research in the incident response domain. Learning in this case involves advancing human knowledge as well, as they define single and double loop learning for incident response. They underline the need for the latter one, i.e. questioning the incident handling in the first place and also organizational structures. As a result, they propose to use the incident response process to increase the analyst's knowledge and the usefulness within the company. However, these processes focus on organizational learning and general knowledge induction from the incident response process and is not concerned with the technical model and evaluation. Hence, in comparison, our approach can rather be seen as a tool to facilitate, especially single-loop, learning. And as such, we focus on a task not further detailed by Shedden et. al., but which does not contradict the concepts they propose.

The human factor within an adaptation approach and more specifically as the target of multi-objective optimization is part of the concepts proposed by Kelley et al. (Kelley, Drielsma, Sadeh, & Cranor, 2018). They present an optimization approach for the adaptation of privacy policies based on user feedback in a location sharing application. They also define the problem regarding different objective functions and discuss methods to meet these requirements in an automated search for improvements of the privacy settings. However, these aspects also depict the only similarity to our approach and as such, the work only concerns one particular part of the approach proposed in this paper. Furthermore, their application scenario,

adaptation of privacy settings, yields different challenges and responses. Consequently, the algorithms for the assessment and adaptation are different from our work.

3 APPROACH FOR AN ADAPTIVE AND INTELLIGIBLE INCIDENT PRIORITIZATION

This section introduces our approach for an adaptive and intelligible incident prioritization. Coming from an understanding that sees flaws in the prioritization as an unalterable circumstance, we define concepts for a continuously improving incident prioritization to cope with resulting challenges. Therefore, we propose the following process model depicted in Figure 1.

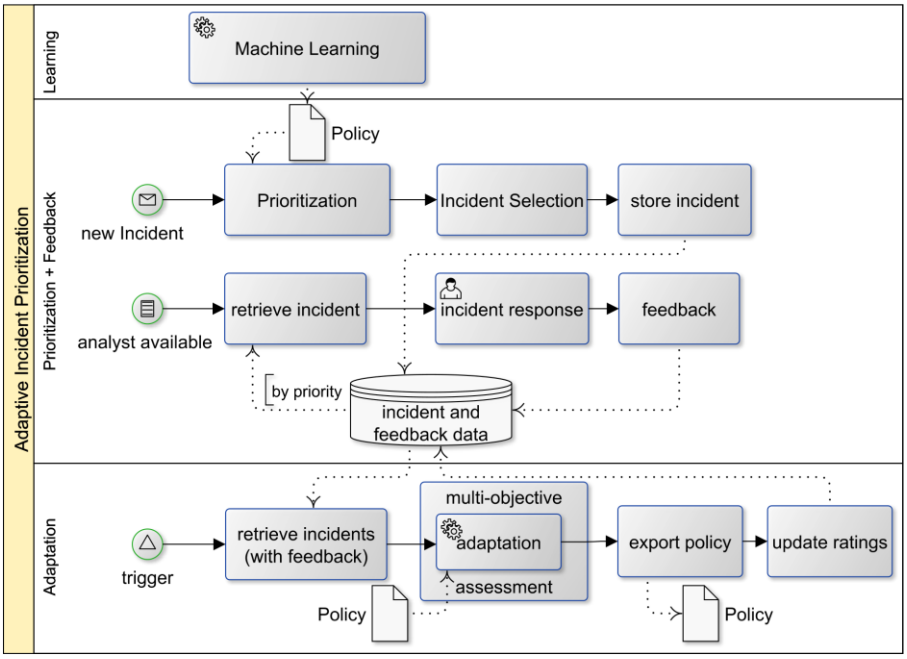

Figure 1: Overview of the main processes for an adaptive incident prioritization.

We consider the incident prioritization and an adjusted priority in terms of incident selection (middle lane). The latter is introduced due to the feedback process, which takes place at incident response time. The adjusted priority directs the response and thus feedback to rare and unseen instances, while the feedback is an appraisal of the priorities by the analyst in order to assess and later adapt the system. The other two parts deal with the increase of

automation and describe the automated induction (top lane), assessment and adaptation (bottom lane) of the prioritization policy. The policy is hereby a central element and defines the actual rules for the priority calculation. Note that the policy, although displayed multiple times for better readability, actually represents a single instance, similar to the database.

The individual process paths and lanes are loosely coupled, although some dependencies between the different tasks exist. For example, an incident needs to be prioritized before it receives a response and similarly, feedback is necessary for the adaptation process, as will be detailed later in this section. Furthermore, the processes are envisioned to be performed in a continuous loop, where each path can start repeatedly due to the depicted events. That is, each new incident will be prioritized and similarly, an incident response is executed for every incident, in order of their priority and depending on the availability of an analyst. The adaptation can as well be triggered on the basis of different events, e.g. manually, after a specific amount of time, after a certain number of feedback has been recorded or when the average error surpasses a given threshold. Yet again, the individual aspects influence each other, as new incidents with different priorities may change the order of the incident response and the adaptation is used to modify the prioritization policy, which again results in reordering of open incidents and further influences future prioritization. That being said, the processes themselves can generally be executed asynchronously and in parallel.

Within the remainder of this section, we first establish an understanding of the prioritization process, including the possibility for feedback and an adjusted prioritization for an improved feedback collection. The second part is concerned with the increase of automation. That is, we discuss how the quality of a prioritization policy can be assessed and finally consider means for the automated induction of a prioritization model and adaptations to an existing policy.

Prioritization

Model

The first aspect is the definition of an incident prioritization model, also as a basis for the remaining concepts. A suitable model should be human readable and modifiable. For one, this is a lesson learned from the intrusion detection domain. Maloof et al. established that “human understandability of learned concepts is important because if a system could act in a way that is harmful to humans, then the concepts responsible for this behavior require modification” (Maloof & Michalski, 1995). Clearly, a system for the

prioritization of incidents falls under this definition as cyber-attacks are threatening to harm humans and companies, which a timely and proper incident response may prevent. Additionally, especially in the network security domain, we need to establish trust in the automated decisions, for which an understandable processing is certainly advantageous. The possibility to comprehend the realization of an outcome allows for a deeper understanding and better interpretation, which again is a necessary requirement to reasonably propose and validate modifications of the prioritization model. We propose to use a rule-based model in order to express prioritization directives. Expert systems have proven to be both understandable for humans and enable an automated processing of the application logic. Furthermore, they can be used to explicitly model influential factors for the priority calculation. An according model has first been established in (Renners, Heine, & Rodosek, Modeling and learning incident prioritization, 2017) and a short overview is given in Figure 2.

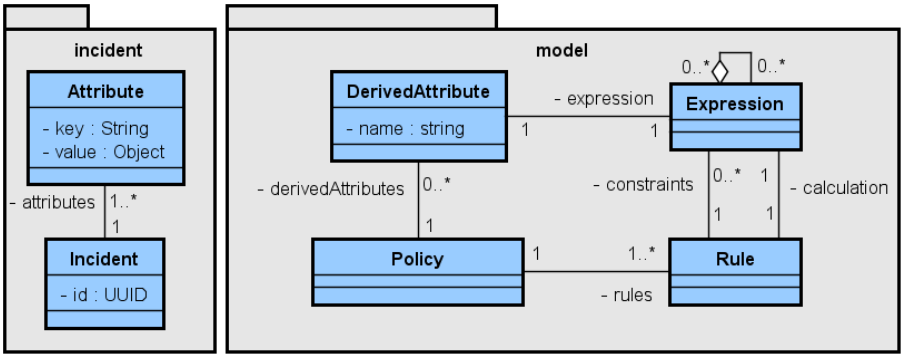

Figure 2: Conceptual class model for rule-based incident prioritization

An *Incident* is generically defined as an entity with *Attributes*. Each attribute is a key-value pair with a name and a value. We then define a *Policy* to express and implement prioritization directives for those incidents. Each policy has a set of *Rules* and *Derived Attributes*. The derived attributes are used to encapsulate re-usable knowledge, for example a check, whether the source ip-address of an incident is within a certain network range. This knowledge is formulated in so-called *Expressions*. These are either Boolean expressions (resulting in **true** or **false**) or arithmetic expressions (resulting in a number), which both may refer to the attributes of the incident. The actual rules to express the prioritization have *Constraints* and a *Calculation*. Constraints are again Boolean expressions which need to evaluate to **true** in order for the rule to fire. The calculation then is an arithmetic expression (which may also contain Boolean expressions) to compute the actual rating.

Naturally, the derived attributes can and should be used within both parts of the rules, simplifying the overall policy and removing redundancies of information.

The prioritization is generally performed by evaluating the policy for incoming incidents, which means first evaluating the derived attributes, then identifying applicable rules and lastly performing the respective priority calculation.

Incident Selection

We add a second step after the calculation of an initial priority to derive an adjusted priority to lower the risk of false negatives. This step is helpful for our application scenario as false negatives are a double threat for incident prioritization. For one, incidents often times are part of an attack chain and the real harm follows after an initial breach. For example, establishing ways for data exfiltration or escalating privileges and spreading onto more critical systems. Thus, a timely response is generally essential to prevent and contain the development of incidents, and false negatives receive a late response as their priority is set too low. The second reason for a concentration on false negatives is given by the related adaptation use-case. A strong synergy between the feedback and the model monitoring and adaptation exists. Yet again, the feedback is directly related to the incident response and as a result, false negatives receive a late response and cannot be used in time to improve the system, thus retaining the errors within the prioritization for the future. We therefore introduce incident selection as a second part of the prioritization process to alter the original priority. The process has to focus the feedback on strong candidates for false negatives, for example based on uncertain decisions or rare characteristics. However, one also has to keep in mind the problems with false positives and the potential increase in work due to the incident selection with regard to a delay for other, regularly and correctly prioritized incidents.

We propose to target this challenge by focusing on uncertainty in the prioritization process, combined with a threshold to account for the most important incidents. That is, we define a confidence measure, which is used to derive an adjusted priority in order to consolidate the feedback process and, built on top, the adaptation. The origin of this measure can be based on two factors: the incidents themselves and the prioritization model. In terms of the proposed policy model, the desired behavior can be enforced by adding another parameter for the confidence to the prioritization rules. However, this approach has the drawback to require an explicit definition of certainties, which again can be quite subjective and error-prone. The

parameters could, however, also be derived from the learning process, for example by leveraging the number of instances and resulting error metrics on a leaf- and thus rule-local level.

The second way is detached from the prioritization model and instead focuses on the incidents. In this case, we pay attention to rare and unseen instances, because these are most likely to contain errors. Furthermore, if a similar incident has already received a response, it is more likely that a potential error has been expressed in terms of feedback as well, which in consequence already enables a correction by the adaptation. For the implementation, a clustering approach can be used to generate a similarity measure for incidents. A promising candidate is the Local Outlier Factor (LOF) as it is a density-based clustering, which can result in a normalized value to describe the outlierness of instances. Outliers are the prime targets for an earlier feedback, as they are the very definition of rare and unseen instances. Additionally, approaches have already been developed to implement an iterative and thus continuous cluster construction, which can work conjointly with the continuous prioritization of incidents.

Both approaches are not necessarily contradictory, but could be applied in combination. Additionally, the threshold to constrain the incident selection needs to be defined and the process in general should be configurable to be deployed in a specific environment. For the realization, we propose the following formula for the incident selection (cf. (Renners, Heine, Kleiner, & Rodosek, 2018)):

$$Pr_a(I) = \begin{cases} Pr_o(I), & Pr_o(I) \geq th \\ \underbrace{Pr_o(I) + (\overbrace{\widetilde{C(I)}}^{[0,1]} * \overbrace{\hat{f}_c}^{[0,1]} * (th - Pr_o(I)))}_{[0,th]}, & Pr_o(I) < th \end{cases}$$

An adjusted priority $Pr_a(I)$ is directly determined by the original priority $Pr_o(I)$, if it exceeds a configurable threshold th. Below this threshold, the confidence $C(I)$ is used to adjust the rating. The value of $C(I)$ can originate from either the policy, the clustering, or a combined value. The factor f_c is introduced to further enable a customization of the influence of this confidence. To represent the requirement that the adjusted ratings may not surpass the threshold, both, the values of the confidence and its parametrization must have values within $[0,1]$. The given equation thus defines the adjusted priority in such a way that the values above the threshold remain unchanged, but the incidents below are adjusted toward the threshold with regard to the confidence of the prioritization process. Since the adjusted priority can never exceed the threshold value, a prioritized

response for severe incidents is ensured. All things considered, we can extend the prioritization process with the incident selection, as depicted in the following Figure 3.

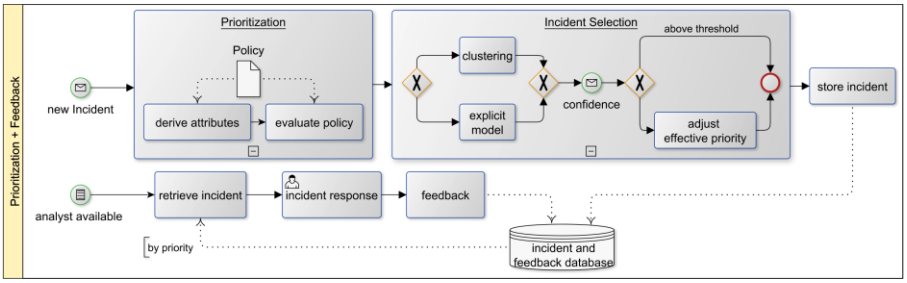

Figure 3: Prioritization and feedback process including incident selection as a confidence-based extensions to enable more diverse feedback and improve its effectiveness within the adaptation.

Feedback

The feedback itself is given in form of correct priority values for the incidents. Positive feedback, i.e. confirming a correct rating, is also possible. The feedback is considered the main data basis for the model assessment. It allows to monitor the quality of predicted priorities and can be used within the later process to evaluate different models and even to automatically propose adaptations.

Increasing Automation for Incident Prioritization

The main goal of our approach is an increased automation in tasks related to incident prioritization. The prioritization itself is already implemented as an automated process in most common products and approaches. The creation, monitoring and improvement of a given policy, however, remains a rather static and manually performed task. Therefore, this part of the proposed approach targets the policy management and discusses the origin and improvement of a prioritization policy.

Policy Model Assessment

First of all, we need means to evaluate the quality of a prioritization policy. This helps to recognize the necessity for an improvement and can direct as well as validate the automation effort. As mentioned before, the feedback is collected for this purpose and therefore plays a major role in this process. However, the understandability adds further important requirements that need to be considered. Therefore, we modeled the problem using multiple

objective functions and identified the following aspects for the assessment of prioritization policies:

Quality is used to describe how accurate the model is able to produce correct priorities. For that matter, the prioritization results can be compared to the corrected feedback data. It can be evaluated using a usual error metric, e.g. the mean average error (MAE). The quality is calculated solely referring to the feedback data. Especially with the introduced incident selection, representatives of incident clusters receive an earlier feedback and the adaptation can correct errors, which are also applicable to incidents that did not receive a response yet. Otherwise, the adaptation would approximate these potentially wrong priorities.

Complexity indicates how difficult to understand the prioritization model is and the main task of this objective function is to lead to an understandable policy. Thus, this objective function refers to the aspect of trust and the possibility to implement and understand changes by an intelligible model. As it is based on the size and comprehensibility of the policy, i.e. the rule base, a measure of the complexity can be defined as the cumulative sum of all elements of a policy. In other words, the complexity of a policy is viewed as the sum of the complexity of each rule, which in turn is based on the sum of the complexity of the expressions in its constraints and calculation. Finally, the complexity of an expression is defined recursively as the number of operators of the expression and all its nested expressions.

Similarity/Distance models the degree of deviations between two policies, i.e. indicating the comprehensibility of the changes. The goal of this objective function is to remain as close as possible to the original one. The similarity can be defined with respect to the complexity and describes how different two policy instances are. In this case, it can be given by the complexity of non-overlapping parts of two policies. The distance can be calculated finding the best match for each element of one policy, summing up the distances and adding up the complexity of each unmatched element in the end. We first need to identify and remove identical elements from the policy. In a top-down approach, i.e. starting at the compound components like rules or derived attributes, down to an expression level, the policy elements are iterated and matched to potential counterparts. These matching elements are then removed from either policy. Finally, the sum of the complexity of the remaining elements depicts the actual distance between two policies.

These three objective functions depict a potential field of tension as the individual requirements may contradict each other. For example, more precise predictions may be achieved by a more complex model as specific cases can be treated individually. Similarly, a reduction of the complexity may still depict extensive disruptions in terms of the similarity, making changes harder to grasp. In order to yield actionable results, the individual objective functions require a method to be combined for an overall comparison. The simplest, but also effective way to achieve this is the application of a weighted sum. That is, weights need to be defined for the three objective functions and their summation yields a final, comparable score. Naturally, these weights should be configurable and are also dependent on the value ranges and a potential normalization of the measures. A normalization of the complexity and distance can be performed by viewing their values in relation to the complexity of the original policy. The error metric typically already has suitable properties. Thus, the assessment can eventually be used to assist with the task of comparing, discarding and approving adaptations. Note, that this comparison is also possible for manual adaptations and generally offers the advantage to shift the attention to a more holistic view.

Policy Induction

The first aspect with respect to an increased automation is the initial policy creation. That is, even before gathering feedback, the initial task of creating a prioritization policy is a difficult and labor intense work. Therefore, we propose to employ learning mechanisms, similar to learning intrusion detection models, to induce prioritization rules. The biggest difference to classical alert classification is the desired outcome of incident prioritization. Instead of classifying an alert into a specific category (of two or more), the outcome is a numeric value. Therefore, the same learning mechanisms cannot be applied. As another requirement, the resulting model should be compatible with a human readable and modifiable policy model. With respect to the model assessment, the quality naturally depicts the most important aspect. However, the focus on an understandable policy is also a desirable property. Clearly, the distance cannot be considered in an initial induction of a policy as there is no reference model.

We developed a Domain Specific Language (DSL) for the proposed policy model. Thus, a policy can be specified manually. However, a second option is automating the process by learning a prioritization model from labeled incident data. One possible candidate for the induction is given by model trees. Within the intrusion detection domain, decision trees depict an established mechanism to induce understandable models for the task of alert

detection and classification. Model trees are a special case of decision trees, but instead of classification, they perform calculations within their leaf nodes. That way, they are generally suitable to predict numeric values. Additionally, their structure can be interpreted in terms of rules and, as such, a transformation into the policy model is possible. In fact, even an abstraction of knowledge, i.e. learning derived attributes, is possible from the split decisions in the tree structure. And to a limited extent, the complexity can be configured by the tree size and minimum number of instances for leaf nodes. More details on model induction for incident prioritization have been the topic of earlier work and can be found within (Renners, Heine, & Rodosek, Modeling and learning incident prioritization, 2017) as well.

Generating Policy Adaptations

The other advance towards further automation within our approach is given by the actual generation of policy adaptations. The process follows a rather similar goal as the one for learning, but has to consider the third objective function, similarity. Therefore, we propose to target this problem by employing learning algorithms, which do not solely rely on the feedback data for the model induction. Instead, we assert to take the existing policy as a starting point and orientate the learning of adjustments along the model assessment. The goal of the adaptation is an improvement of the incorrect prioritization with respect to the feedback data, while maintaining a simple model, which stays as close as possible to the existing policy. Consequently, the adaptation algorithm has to use the multi-objective assessment guide the process towards better adaptations, regardless of the applied techniques. Although the outcome, a prioritization policy, is equivalent, the adaptation can and has to take all objective functions into consideration in order to produce suitable results. Thus, the initial algorithms for the policy induction cannot be used directly. An isolated learning of two models from similar data may yield very different results. This is especially true for a learning approach like model trees, where a variation in the higher part of the hierarchy has a large influence on the remaining model construction. We considered several ideas, including evolutionary algorithms, as they are a prime candidate for multi-objective optimization, but the best results so far have been achieved by the following greedy approach. The idea is to directly use the current policy model and perform changes to the policy elements sequentially and greedily. After each individual alteration, the change is either maintained or rejected, depending on an improvement with respect to the objective functions. To determine possible alterations, we mainly exploited the policy structure. As a result, we propose a combination of the following two ideas.

Changing Expression Values (Ada-Exp): The first strategy focuses on the smallest units within the policy model, the expressions. As the expression structure in general is too universal to systematically derive changes, we focus on a specific type, which compares the incidents attribute values. Hence, this approach mainly deals with incorrect specifications of expressions used to define lists and ranges of data, e.g. a set of system assets, or an address range for network segments. Mainly, this knowledge would be used within the derived attributes to encapsulate and re-use the information. Consequently, these errors lead to incorrect calculations, although the rules are actually logically sound, but the decisions are based on wrong reference values. We introduce changes to these value ranges based on the properties, i.e. attribute values, of the incorrectly prioritized incidents. The algorithm iterates over all attributes of the incidents and for each attribute, all relevant expressions within the policy are retrieved. Then, for each distinct value of that attribute within the faulty data, the value side of each expression is inverted to either include or exclude the attribute value. As mentioned, these changes are performed greedily, i.e. the change is retained in case of an improvement of the policy or reverted otherwise.

Replacing and Refining Rules (Ada-Rule): The second concept is developed to target errors in the rules of the policy themselves. The approach therefore adopts the insights gained from studying model trees and their rule structure as a candidate for policy learning. Learning a completely new policy from a model tree violates the idea of preserving a similar model, but the concepts of the approach are still relevant to the adaptation use-case. In particular, the same learning mechanism can be applied, but only for a specific rule within the policy. To that end, the error of the policy is broken down to the rule-level and error-prone rules are replaced by, potentially multiple, new rules. Figure 4 depicts a minimal example of the process.

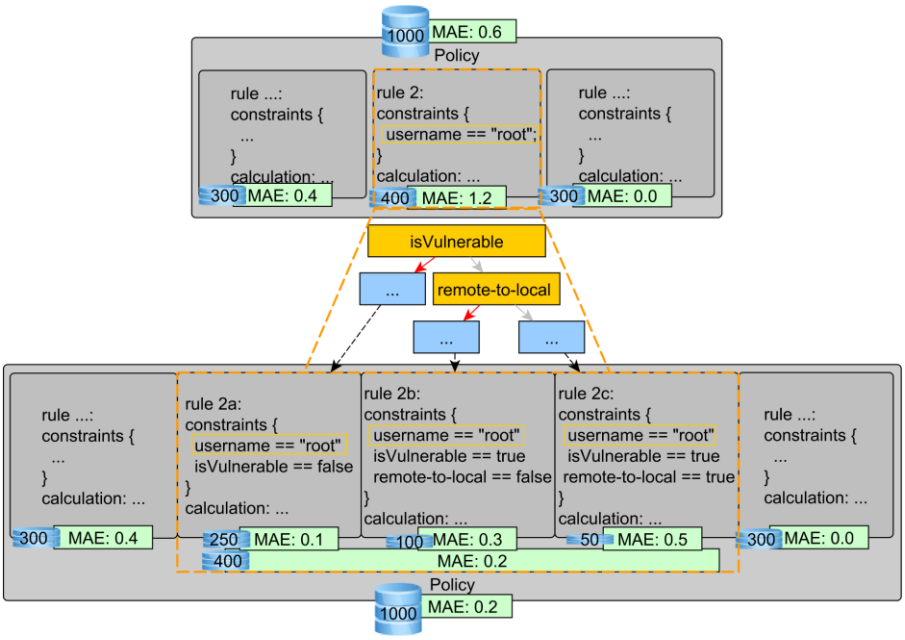

Figure 4: Exemplary depiction of the rule refinement using model tree induction to generate more specific sub-rules.

The incidents rated by the most error prone rule (here: number 2) are used to learn a new model tree. For each leaf node of that model tree, a new rule is created and added to the adapted policy, retaining the constraints from the original rule, but adding further constraints and adopting the calculation from the model tree. The constraints from the replaced rule have to be maintained, as they were responsible to narrow down the incidents to that point. The new sub-rules thus define processing of data actually reaching this far down in a more specific manner, including a potential correction of the actual calculation. As these changes are performed locally and rather small model trees are built, the overall structure of the policy can remain the same, which benefits similarity to the original policy. Attempting adaptations only in the order of error size thus focuses the changes on the most serious issues.

In a future production system, it is certainly desirable and recommended to use a combination of these two approaches. Adjustments aim at two different modification aspects of the policy model and thus also target distinct error types. Such a combination can for example be the sequential application of either approach (Ada-Seq) after the other. However, the general description of the approach naturally allows to choose or exchange

multiple and different strategies. The multi-objective assessment can in either case and for all candidates be used to compare and evaluate the different adaptations, to the point where a suitable representative is found to be used for future prioritization.

4 EVALUATION OF CONCEPTS AND ALGORITHMS

We implemented our approach for an adaptive and intelligible prioritization of network security incidents using the concepts described above and with the details stated in the referenced publications. On that basis, a first evaluation of the concepts has been performed on a real-world data set of QRadar events as well as on synthetic data with reference to the prioritization directives documented for the ArcSight system.

Initial Learning

In (Renners, Heine, & Rodosek, Modeling and learning incident prioritization, 2017) we have examined the capabilities of the model tree induction for learning prioritization policies. The results have shown that model trees can indeed be used to learn policies and that derived attributes depict a possibility to reduce and shift the complexity of the resulting model without negatively influencing the model's prediction quality. However, we have also seen that the complexity in general can be rather high, since model trees and linear models are limited in their capabilities to approximate prioritization directives. Yet, ultimately learning model trees can be used to generate an initial set of prioritization policies.

Adaptation

The core of the evaluation in this paper is an analysis of the possibilities of the adaptation approach under the application of the multi-objective model assessment. In a comparative assessment (evaluation part in Figure 5), we checked the results of the adaptation algorithms discussed in the previous section (namely **Ada-Rule**, **Ada-Exp** and **Ada-Seq**) against the outcome of isolated new model tree learning (**MT-Policy**) and in comparison to the predictions from the construction of an artificial neural network (**nn**). In order to perform a realistic assessment, a complex setup procedure has to be employed (cf. Preparation in Figure 5 which will be explained below and gives a complete overview of the evaluation of the adaptation.

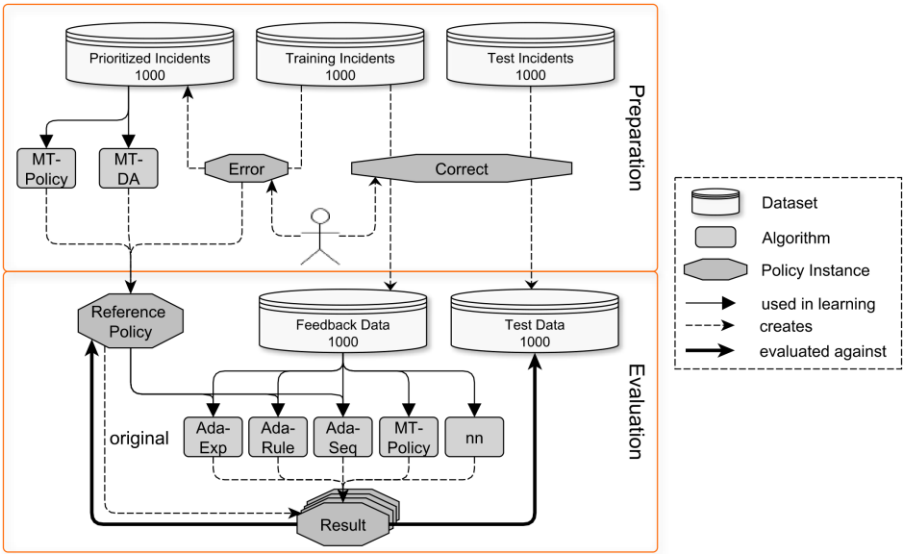

Figure 5: Overview of the evaluation setup using subsets of the two different datasets, diverse starting policies and various learning approaches.

Different adaptation mechanisms are evaluated and in order to analyze the versatility and for a better comparability the evaluation is performed on varying starting policies and includes further learning algorithms.
In particular, the following steps are performed:

1. Acquisition of incident data and separation into **Training Incidents** and **Test Incidents**.
2. Manual specification of a policy (**Correct**) that correctly represents and defines the labels for the **Feedback** and **Test Data**.
3. Manual modification of the correct policy to introduce errors in the prioritization process and generate the first reference policy (**Error**).
4. Creation of priorities for the **Prioritized Incidents** by applying the incorrect policy (**Error**) to the **Training Incidents**.
5. Learning of the further reference policies **MT-Policy** and **MT-DA** from the **Prioritized Incidents**, which thus also contain errors with respect to the **Feedback Data**. The names are chosen in reference to the algorithms, i.e. to learn a policy using model trees and including an abstraction of derived attributes.
6. Application of the different adaptation approaches, depicted as gray boxes, using the **Feedback Data** and each different **Reference Policy**. In particular, adapting rules (**Ada-Rule**), changing expression values within derived attributes (**Ada-Exp**) and the

combined sequential approach (**Ada-Seq**) were applied. Although independent of a reference policy, the performance of regular model tree learning (**MT-Policy**) and an ANN (**nn**) are included in the evaluation.

7. Performance measurement of the adaptation results on the correctly labelled **Test Data** and in comparison to the reference policy itself (**original**).

For the adaptation, the whole feedback dataset is used to adapt the policy and the test incidents are used to evaluate the performance of the adapted model (result). The following figures show results for the adaptation evaluation. Figure 6 shows results for the ArcSight dataset starting with the error policy, whereas Figure 7 uses the same dataset but starting with the **MT-DA** policy. Finally, Figure 8 shows results starting with the error policy on the QRadar dataset.

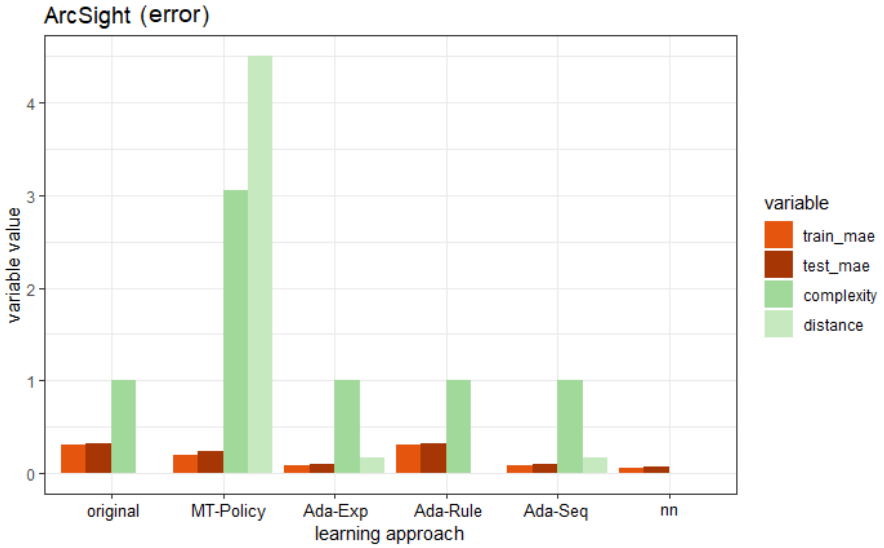

*Figure 6: Adaptation result quality for ArcSight data starting with **error** policy.*

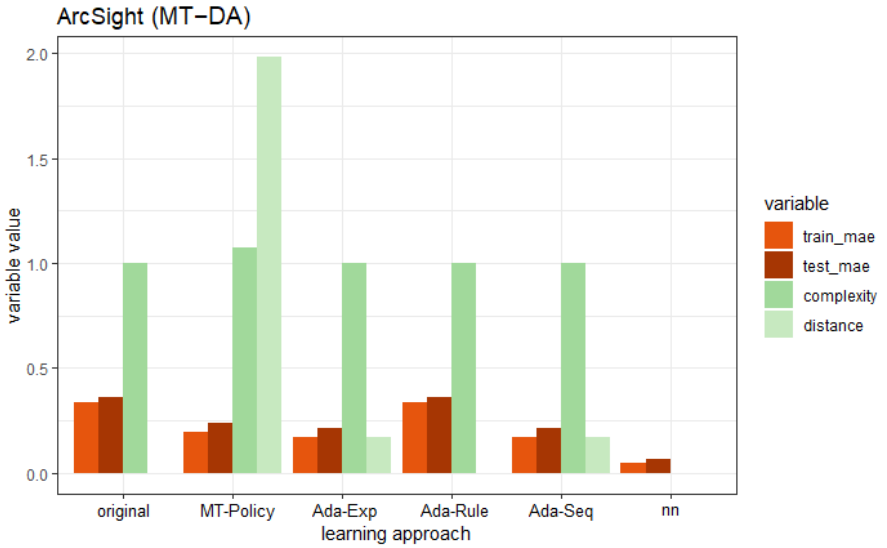

*Figure 7: Adaptation result quality for ArcSight data starting with **MT-DA** policy.*

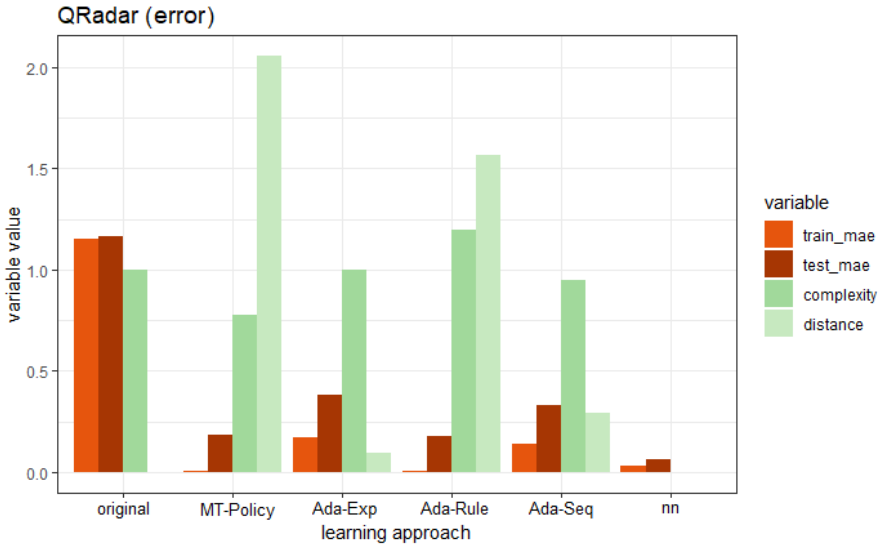

*Figure 8: Adaptation result quality for QRadar data starting with **error** policy.*

In all figures, different approaches are aligned along the x-axis for comparison: the reference policy (**original**), standalone stock model tree learning (**MT-Policy**), the three different adaptation approaches (**Ada-Exp**:

changing expression values, **Ada-Rule**: rule adaptation and **Ada-Seq**: combined adaptation) and finally the results of training an ANN (**nn**). Multiple measurement metrics corresponding to the multiple optimization goals are displayed as the bars of the chart indicated by the different colors. The Y-value thus describes the results on each of the respective metrics. As the measures are within similar value ranges, such a grouped display is possible. The similarity of the value ranges arises from the considerations for the weighted formula approach, thus using values relative to the original policy for the complexity and distance. The main comparison is, however, focused on the same metrics between different approaches. As for the complexity and distance, the ANN results are not comparable as ANN have a fundamentally different model structure. The goal of an ANN is not primarily comprehensibility and as such both model complexity and distance would result in extremely high measures for an ANN (e.g. proportional to the number of neurons). In order to be able to draw conclusions on the other approaches and metrics, these values are omitted here for ANN.

The results show that all approaches have successfully decreased the error of the predictions, which was the main goal of the adaptation and primarily weighted at the algorithm configuration. In particular, Figure 6 shows that the approaches **Ada-Exp** and **Ada-Seq** provided the lowest error metrics combined with the best measures for similarity and distance, especially when compared to new learning of a policy (**MT-Policy**). This clearly proofs the advantages of an adaptation approach over new model learning. In comparison Figure 7 still shows advantages of these two approaches with regard to distance and complexity, yet this time with comparable error metrics. From Figure 8 one can also conclude similar results for another dataset, proving the general validity of the results. Here comparable error metrics for all approaches can be found, yet better similarity and distance metrics again for **Ada-Exp** and **Ada-Seq**. Specifically the advantages of **Ada-Rule** over **MT-Policy** are much smaller here. This can be attributed to some specific properties of this sample dataset with a restricted domain for priorities.

In summary, in all experiments one of the adaptation approaches outperformed the isolated learning regarding the understandability. Note that both factors, complexity and distance, have to be considered. In multiple cases, mainly on the ArcSight dataset, the adaptation results also outperformed stock model tree learning solely regarding the error-metrics on the training and test data. In a combined effort, totalizing the errors and

including distance and complexity metrics, an overall assessment of the quality further underlines the advantage of the adaptation approach.

Incident Selection

In a second step, we examined the influence of the incident selection on the aforementioned adaptation process. There are primarily two reasons why a low impact of the incident selection is desirable. On one hand, in a practical application not all incidents might get feedback from a human expert due to resource constraints. Thus a good priority computation based on a smaller number of incidents as training data will help in this scenario. In addition, the smaller the number of the incidents required to achieve a low prioritization error, the faster it is possible to adjust the priorities to their actual values. Thus a system requiring fewer incidents as training data will be more dynamical in the adaptation.

The incident selection is evaluated by providing partial **Feedback Data**, e.g. only 20% instances training incidents. These chunks of the dataset are then chosen in two ways. The number of instances is for one chosen in order of the original priority and secondly with the application of the incident selection and ordered by the adjusted priority. Thus, different instances are given as feedback data, although the same instances receive the same feedback. This allows to observe the impact of the different, partial feedback data on the effectiveness of the adaptation process and accordingly an assessment of the advantages of the selection process.

Figure 9: Incident selection impact for ArcSight data starting with error policy.

Figure 9 shows the results of the impact of the incident selection on the mean average error only exemplarily for the ArcSight dataset starting with the erroneous policy.

The first observation to note is that the results generally improve with the amount of training data. The largest impact of the reduced number of training data is on the isolated learning approaches, the model tree learner (**MT-Policy**) and the ANN (**nn**). Likewise, the incident selection also has the biggest effect on these algorithms. For the ANN, applying outlier detection to include different incidents in the training data has roughly bisected the resulting error on the test data. Although not as distinctive and steadily in the effect, the same trend can be recognized for the stock model tree learning. One explanation of the phenomenon that the adaptation approaches are not as affected is that a lot of information is already contained in the reference policy. Although it contains certain errors, it still also contains a lot of correct rules and concepts. Therefore, the models that were learned in isolation and only on the reduced data are much more biased by the limited data, as nothing is used in addition. In contrast, correct knowledge encapsulated in the existing model will be carried into the adapted model. This is underlined by the fact that some learning results even perform worse than the original, error prone policy. This also highlights the general idea and advantage of adaptation, as the information from the existing model is not lost, but carried towards the new model.

The incident selection also positively influences the remaining adaptation approaches. The prediction errors of equivalent experiments generally decrease with the application of the clustering approach. Of these algorithms, the modification of derived attributes (**Ada-Exp**) as well as its application in the combined approach (**Ada-Seq**) yielded the best results. This can be attributed to the starting policy again, since the manually defined policy error has been used, which mainly contains errors in the derived attributes. For policies on the basis of model trees, the rule-based adaptation (**Ada-Rule**) performs better than **Ada-Exp**, but the combined approach **Ada-Seq** with incident selection is again mostly comparable or even better than either of the individual adjustments.

Overall Evaluation Results

In summary, the evaluation shows that the adaptation approaches can outperform the isolated learning. Comparable results regarding the prediction quality (error) are achieved by the adaptation and the isolated model tree learner, where the adaptation algorithms sometimes even exceeded the isolated induction. Furthermore, the targeted algorithms better

take the complexity and distance into consideration, showing the main advantage considering all objective functions. Yet again, training an artificial neural network resulted in still better priority predictions, but the understandability of the model and transitions for the adjustment are not human-comprehensible by any means. Furthermore, the old model may also contain relevant information, but is only used in the adaptation algorithms. This is also shown by the evaluation of the incident selection. The largest impact of the choice of limited training data could be observed for the isolated learning approaches. The adaptation algorithms can retain knowledge from the old model, thus not solely relying on the limited data. Nevertheless, all learning approaches were positively influenced by selecting the incidents on the basis of the proposed confidence-based calculation in comparison to solely relying on the regular priorities. Thus, these results affirm the assertion that a more diverse data selection can improve the feedback and adaptation process. Further details on the adaptation and specifically the evaluation can be found in (Renners, Heine, Kleiner, & Rodosek, Design and Evaluation of an Approach for Feedback-based Adaptation of Incident Prioritization, 2019), which explicitly focuses on this part of our approach.

5 FUTURE RESEARCH DIRECTIONS

One interesting aspect of a combined analysis is to put the additional effort imposed by the feedback process, and especially by the incident selection, in perspective to the advantages generated by the continuous adjustment and adaptation. While the advantage from an algorithmic and learning perspective could be shown, the impact of the increased work accompanying this process are not yet analyzed satisfactorily.

Furthermore, specific feedback is already a challenge in itself and one could also consider the possibility that the given feedback is not necessarily accurate. Thus, two additional areas of research also become relevant in the context of this work. The first is the analysis and improvement of the sensitivity of the algorithms with the use of error-prone training data. A second direction of improvement of the feedback process is given by a different type of feedback, and its respective use in learning and adaptation approaches. Admittedly, it might be difficult to give precise numeric feedback, but it is for example possible to express trends, i.e. too high or too low, in combination with fuzzy concepts like *a little* or *a lot*. The tendency of human reasoning to focus on granules, which are not crisp, has already been established in the research community concerned with fuzzy systems, e.g. by Zadeh (Zadeh, 1997) and respective application and learning methodologies have been proposed (Klir, St. Clair, & Yuan, 1997). An

interesting approach would be to examine the applicability of those systems in the context of incident prioritization or more importantly in terms of learning and adapting a policy. One explicit possibility is the application of a transformation between fuzzy and crisp concepts, i.e. fuzzification and defuzzification (Leekwijck & Kerre, 1999), to employ the concepts of this thesis in conjunction with techniques from the fuzzy domain.

The last major area of future work is a practical integration of this approach with existing network security components. Most prominent are SIEM systems, because they depict the initial application scenario. Those systems provide an ideal environment for an approach such as this, because the detection of incidents as well as a lot of information for the prioritization can be provided by the encompassing system. However, the challenge is to apply a common interface for the data exchange and in particular a common understanding of an incident and priority related data. For example, the application of derived attributes to real world concepts could be a point of modification. Many systems already maintain data modeling concepts, like used in derived attributes, and it appears impractical to redesign this data. Instead, points of possible interaction and modification for the learning and adaptation components could be defined and used in the processes for an adaptive incident prioritization.

6 CONCLUSION

In this paper, we introduced our approach for an adaptive and intelligible prioritization of incidents. Acknowledging the potential for errors in the prioritization policy, we propose concepts to increase the degree of automation in the model creation and adjustment. Thereby, we target two challenges in the SIEM domain. Namely, scarcity of resources, particularly for manual tasks, and lackluster prioritization.

Our approach provides new concepts beside the common prioritization and adds further processes for the induction of a model as well as continuous feedback collection, assessment and adaptation of the prioritization policy. Beside a high-level definition of these tasks, we introduce several concepts for the realization of the different aspects. Altogether, these aspects can help to overcome some of the current downsides in incident prioritization, especially by focusing on automation of the policy management. Meanwhile, our approach aims to keep the analyst in the loop and considers understandability a fundamental requirement, as we believe that the human factor plays a major role and is practically irreplaceable at the current state of network security. An evaluation of the different parts of the approach showed promising results.

7 REFERENCES

Axelsson, S. (2000). *Intrusion Detection Systems: A Survey and Taxonomy.* Chalmers University.

Ben-Asher, N., & Yu, P. (2017). Synergistic Architecture for Human-Machine Intrusion Detection. *Cyber Security and Information Systems*.

Bhatt, S., Manadhata, P. K., & Zomlot, L. (2014, September). The Operational Role of Security Information and Event Management Systems. *IEEE Security Privacy*, (pp. 35-41).

Das, S., Wong, W. K., Dietterich, T., Fern, A., & Emmott, A. (2016). Incorporating Expert Feedback into Active Anomaly Discovery. *2016 IEEE 16th International Conference on Data Mining (ICDM)*, (pp. 853-858).

Debar, H., Dacier, M., & Wespi, A. (2000, July). A revised taxonomy for intrusion-detection systems,. *Annales Des Télécommunications*, pp. 361-378.

FireEye. (2014). *The SIEM Who Cried Wolf: Focusing Your Cybersecurity Efforts on the Alerts that Matter.*

IBM. (2017). *X-Force Threat Intelligence Index 2017.*

Kelley, P. G., Drielsma, P. H., Sadeh, N., & Cranor, L. F. (2018). User-controllable Learning of Security and Privacy Policies. *Proceedings of the 1st ACM Workshop on Workshop on AISec*, (pp. 11-18).

Kim, A., Kang, M. H., Luo, J. Z., & Velasquez, A. (2014). *A Framework for Event Prioritization in Cyber Network Defense.*

Klir, G. J., St. Clair, U., & Yuan, B. (1997). *Fuzzy Set Theory: Foundations and Applications.* Prentice-Hall, Inc.

Leekwijck, W. V., & Kerre, E. E. (1999, December). Defuzzification: criteria and classification. *Fuzzy Sets and Systems*, pp. 159-178.

Maloof, M. A., & Michalski, R. S. (1995). *A Partial Memory Incremental Learning Methodology and its Application to Computer Intrusion Detection.* Department of Computer Science, George Mason University.

Ponemon Institute LLC. (2017). *Challenges to Achieving SIEM Optimization.*

Renners, L., Heine, F., & Rodosek, G. D. (2017). Modeling and learning incident prioritization. *2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, (pp. 398-403).

Renners, L., Heine, F., Kleiner, C., & Rodosek, G. D. (2018). A Feedback-Based Evaluation Approach for the Continuous Adjustment of Incident Prioritization. *2018 1st International Conference on Data Intelligence and Security (ICDIS)*, (pp. 176-183).

Renners, L., Heine, F., Kleiner, C., & Rodosek, G. D. (2019). Design and Evaluation of an Approach for Feedback-based Adaptation of Incident Prioritization. *2019 2nd International Conference on Data Intelligence and Security (ICDIS)*, (p. accepted for publication).

Shedden, P., Ahmad, A., & Ruighaver, A. (2010). Organisational Learning and Incident Response: Promoting Effective Learning Through The Incident Response Process. *Australian Information Security Management Conference.*

Townsend, T., & McAllister, J. (2013). *Implementation Framework – Cyber Threat Prioritization.*

Veeramachaneni, K., Arnaldo, I., Korrapati, V., Bassias, C., & Li, K. (2016). AI^2: Training a Big Data Machine to Defend. *2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing*

(HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), (pp. 49-54).
Wu, S. X., & Banzhaf, W. (2010, January). The use of computational intelligence in intrusion detection systems: A review. *Applied Soft Computing*, pp. 1-35.
Zadeh, L. A. (1997, September). Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems*, pp. 111-127.

KEY TERMS

Cyber Security: Protection of computer and network systems from malicious attacks.
Security Information and Event Management (SIEM): Approach and tool for cyber security that provides a holistic view on the IT by gathering, correlating and analyzing information from different data sources.
Incident: Single or a series of unwanted or unexpected information security events that have a significant probability of compromising business operations and threatening information security.
Prioritization: Process of determining a priority, i.e. importance, of an information security incident.
Adaptation: Process to adjust a system to changed circumstances.
Machine Learning: Scientific area to employ statistical models in order to infer knowledge from existing knowledge, which is applicable to new and unseen data.

BIOGRAPHICAL NOTES

Leonard Renners has studied Computer Science and received his Masters Degree in 2013 at the University of Applied Sciences & Arts Hannover, Germany. He became a research associate at the research group Trust@HsH with a focus on network security and SIEM and currently pursues a Ph.D. in this domain in cooperation with the Universitaet der Bundeswehr Muenchen. His research fields are cyber security and SIEM systems, although ideas from machine learning increasingly complement these topics.

Felix Heine is a full-time professor in database and information systems at the University of Applied Sciences & Arts Hannover, Germany. He is co-leading the research group Trust@HsH since 2014 and his research interests include database and information systems, machine learning, esp. in the area of IT security, and data quality. He is a co-founder of the university's research cluster for smart data analytics.

Carsten Kleiner is a full-time professor in secure information systems at the University of Applied Sciences & Arts Hannover, Germany. He is co-leading the research group Trust@HsH since 2014 and his research interests include database and information systems, novel types of database systems and their applications, data analysis methods in security, data quality. He is also a co-founder of the university's research cluster for smart data analytics.

Gabi Dreo Rodosek is a full-time professor and chair for communication and network security at the Universitaet der Bundeswehr Muenchen, Germany. In addition, she is the director of the research institude CODE (Cyber Defense) and member advisory and supervisory boards of several companies and public institutions in Germany.

REFERENCE

Reference to this paper should be made as follows: Renners, L., Heine, F., Kleiner, C. & Dreo, G. (2019). Concept and Practical Evaluation for Adaptive and Intelligible Prioritization for Network Security Incidents. *International Journal on Cyber Situational Awareness*, Vol. 4, No. 1, pp99-127.