Data-driven Approach to Information Sharing using Data Fusion and Machine Learning for Intrusion Detection

Lars Christian Andersen*[1], Katrin Franke†[2] and Andrii Shalaginov‡[2]
[1]mnemonic
[2]Norwegian Information Security Laboratory, Center for Cyber- and Information Security, Norwegian University of Science and Technology

Abstract

Intrusion Detection System (IDS) sensors are employed in various locations in computer and communication networks to identify possible malicious activities. One of the main challenges with IDS is the high false positives rate, which creates a high unnecessary workload for human analysts at Security Operation Centre (SOC). Similarly, the exponential growth of captured sensor raw data combined with the application of Threat Intelligence (TI) creates a complex data flow. Considering mentioned challenges, this paper presents a model of heterogeneous sensor and TI data fusion and reduction in intrusion detection. We summarize found literature and qualitative research interviews with security experts from law enforcement and public and private organizations. Building on our qualitative research we identified feature subsets for corresponding data fusion that produce accurate classification model in Machine Learning (ML)-aided analysis. Proposed data fusion process model was successfully evaluated on a real-world dataset from a SOC. This work contributes to development of data-driven approach for automated classification of IDS events using reduction of raw log data.

1 Introduction

The number of security incidents worldwide is increasing and the security community relies on the ability to detect and to react to such threats. Historically, information security is a continuous cycle where vulnerabilities are discovered, exploited by malicious actors, and patched by the information security community. As new vulnerabilities and exploits are observed, signatures or patterns indicating malicious activity are created. These signatures are used by Intrusion Detection Systems (IDS) to detect malicious activity in networks. The IDS create alarms for human analysts for which to decide on what action to be taken. Unfortunately, many of these alarms

*larsch@mnemonic.no †kyfranke@ieee.org ‡andrii.shalaginov@ccis.no

This paper was presented at the NISK-2016 conference; see http://nikt2016.hib.no/.

are False Positive (FP), that is wrongly raised alarms. It has been observed that up to 99% of the triggered alarms are FP [1], and finding the True Positives (TP), correctly raised alarms, are labour-intensive. The high work load can lead to errors and thus False Negatives (FN), that is misclassification of a correct raised alarm. The work load of the human analyst can be decreased by aggregation and correlation of alarms. However, this is not enough in a large scale Security Operation Centre (SOC). The need for systems to reduce and streamline the process is present.

Applying Machine Learning (ML) approaches to event classification can provide great benefits to the daily operation of a SOC [2]. However, several problems are arising when considering the performance of the classification process. Blindly applying ML to data will not result in desired performance in most cases, and may only increase the computational complexity [3]. Furthermore, there is little knowledge about which features are the most reliable, hence sufficient classifier performance cannot be guaranteed. Also one can see that there is a knowledge gap that requires up-to-date identification of relevant attacks indicators. Identifying the most reliable and trustworthy features in aggregated and correlated data is needed. In addition to this, these features may provide a more efficient way of sharing information for situation awareness [4] and Threat Intelligence (TI). We can see that there exist data-driven TI solutions like one provided by Sqrrl [5] and corresponding governmental guidelines [6]. So, understanding the data is crucial to ensure that chosen features provide the best problem-wise classification.

The contribution of the paper is two-folded. (i) Identification of requirements for data fusion in intrusion detection based on relevant literature and research interviews with security experts. A model for data fusion and reduction is also proposed adhering to the specified requirements. (ii) An automated identification of reliable and trustworthy features in correlated and aggregated intrusion detection events for ML-aided classification. For this we performed data-driven experiments on real-world data from the SOC of mnemonic. The remainder of paper is organized as following. Section 2 introduces the concept of data fusion and TI as well as importance of Feature Selection (FS) for information sharing. Further, Section 3 outlines our two-folded methodology that covers both requirements for data fusion based research interviews with security experts and automated data reduction method based on FS. A complete model structure is given in the end of the section. Practical aspects of the experiments are described in the Section 4. Later, the analysis of accordance between human expert interviews and data-driven approaches is provided in Section 5 along with the ML-aided classification accuracy. The conclusions can be found in Section 6.

2 Background

An overview of the state-of-the-art in data fusion in security operations is presented. Next, state of-the-art in reliable FS is discussed.

Data fusion in Security Operation

In intrusion detection, a common problem is the high rate of FP. As a result, there has been numerous work on decreasing the FP level as well as the general level of alerts [7, 8, 9]. Nguyen et al. [2] identified current gaps in existing alert management. Thereafter they proposed an efficient alert management approach reducing unnecessary alerts from IDS. Their approach uses two modules: alert

verification module which validates alerts with vulnerability; aggregator module which removes redundant alerts. The aggregator module reduces the volume of alerts by aggregating alerts belonging to the same attack within a time window. This is performed by sending alerts to predefined sub aggregator for each class of attack. Each of these sub aggregators combines relevant alerts and create a meta alert, efficiently reducing the volume of alerts. Their aggregation approach uses simple fusion by fusing when all features are overlapping. In their experiment, features IP, port, and time were used. The approach also allows for aggregation of meta alerts. For evaluating the effectiveness, they used following measure:

$$reduction \quad rate = \frac{filtered \quad alerts}{total \quad number \quad of \quad alerts} \tag{1}$$

Based on their testbed with three different IDS, they achieved reduction rate between 44.4% and 59.5% over five attack classes with an average of 50.39%. Many other studies have focused on the FS process, and the measures used. The earliest approach for FS in ML focused on filtering [10]. Work like Schlimmer [11] and Almuallim and Dieterich [12] approached the problem by finding the minimal combinations of features which are consistent with the training data. Other filtering methods have been proposed in seminal work such as Kira and Rendell [13]. A more recent work by Hall and Holmes [14] presents a benchmarking for several FS methods. The performance of each method was assessed based on the classification accuracy of two well-known classifiers Naive Bayes and C4.5.

Data-driven Threat Intelligence

When discussing state-of-the-art in data-driven TI, industry is where to look. In the last few years, numerous companies and product lines have surfaced applying Big Data technologies and mindsets to the classical security operation. The common denominator of many of these product lines is that they focus on automation of the process of combining TI and various internal data sources. Companies like **SQRRL** [5], **Recorded Future**[1], and **Digital Shadows** [4] apply data-driven approaches for achieving situational awareness and for detection. More specifically, they apply ML methods to unify a large amount of various data sources.

Reliable feature selection

To the authors' knowledge, there has not been much work approaching the reliability of the FS process. Nguyen et al. [15] performs an analysis of the main factors affecting the reliability in FS: (i) choice of FS method and (ii) search strategies for relevant features. A formal definition of a reliable FS process is given taking into account the main factors analysed: (i) steadiness of the classifier, and (ii) consistency of the search strategy. A method for addressing the main causes of low reliability in FS is proposed as Generic Feature Selection (GeFS) measure. The reliable FS process can then be seen as a maximisation problem finding of feat set that maximises GeFS(x). This method is applied to two datasets, the ECML/PKDD 2007 dataset and a new CSIC 2010 dataset created by the authors. The first dataset contains attack requests that are constructed blindly [15]. Therefore, authors produce their own dataset generated from an e-commerce web application achieving more than 90% accuracy of classification. Berg et al. [3] applied the GeFS

[1] https://www.recordedfuture.com/

method to the problem of botnet malware detection. The authors conduct their own experiments to construct a botnet malware dataset. Static and dynamical approaches are used creating a dataset of 7,308 features. Data analysis shows that many features are linearly correlated. Considering mentioned results, we can see that the proposed GeFS greatly reduce the number of features while on average increase the detection rate. Compared to similar FS methods, both the feature reduction and average detection rate are better. There is, however, no comparison of the steadiness and consistency of the resulting features in their work. As result we believe that generally accepted and verified FS methods like GeFS can be used for our purpose.

Information Sharing

The principle of information sharing has been applied in various fields ranging from military sector to health sector. However, much of the approaches for information sharing is proprietary, and methods and formats used is created on a per scenario. The government of New South Wales (NSW) in Australia has published for guides for information sharing between different entities [6]. The entities are government agencies, non-government sector, research and public sectors. The guides are a part of the NSW Government ICT Strategy. They are designed to help entities prepare, manage and capture the benefits of information sharing. NSW government has also created a framework for information management. The framework aims to support the management and use of data and information for the government and contains a set of standards, policies, guidelines, and procedures. It creates a common frame of reference which supports the sharing and re-use of information by other entities. These previous works by governments provide good guidelines for information sharing, and they have identified the entities often performing information sharing. However, a general approach, and may not be directly applicable to information sharing in regards to TI.

3 Methodology

Below, the methodology for qualitative assessment of information sharing and quantitative evaluation of ML-aided data fusion is given.

Research Interviews

Selecting interview as part of the methodology was done for several reasons. It is, in information security, important to have communications between academia and industry. The continuous process of research, implementation, application, and feedback allows for new technology and techniques to be developed and used in the current and future fight against cyber criminals. By interviewing security experts in industry, feedback can be collected which then are used for further research. It is important to state questions without limiting their response to ensure as much information as possible is collected. Therefore, qualitative interview is best fitting [16]. The interview was divided into three main parts; *Information Sharing* discussed topics related to the sharing of information, focusing on sharing partners, trust, and technologies; *Threat Intelligence* discussed topics related to what and how intelligence was used in the organisations, how advanced current use was, as well as the effect of such intelligence; *Data Fusion* discussed topics related to how current fusion processes were designed, the potential requirements for such a system,

as well as how such processes can be designed more efficiently. The interview subjects selected are from various fields of the information security community: private organisations, public organisations, and legal enforcement. The group of interview subjects consist of experts in both technical and operational positions. It is important to note that the interviews were performed for information collection purposes, and not statistical purposes. Questions were asked to understand the current problems and solutions in information security industry.

Requirements for Data Fusion process model

Based on the corresponding literature and interview process, requirements for a data fusion, reduction, and sharing process model is identified. By seeing the advantages of previously proposed fusion process models, we seek to design a process model decreasing or removing identified flaws. Further, by identifying how industry performs fusion and sharing, combined with the current flaws in these approaches, we seek to design a process model based on both academia and industry. The following requirements have been identified:

Cyclic Ensuring that the model clearly describes a cyclic process is important. The fusion process should be a continuous cycle to ensure situational awareness.

Detailed definitions According to Bedworth and O'Brien [17], a process model should provide a sub-division of the problem which is rich and detailed enough to allow reuse of specific knowledge. By breaking the problem into sub-problems, we can create a set of problems which are easily solvable and implementable.

Automation Human analysts can only do so much, and including automation for increasing efficiency as well as providing decision-support is imperative. Automation in terms of sharing and inclusion of data allows for an efficient system which is continuously up-to-date with the existing threat environment.

Sharing In the current fight against cyber criminals, the sharing of TI to trusted external parties is important. According to Gartner [18], 60% of digital business infrastructure will rely on TI to ensure operational resilience by 2019. The sharing process should be a two-way flow which allows for the inclusion of new TI into the fusion process. The standardisation of sharing is necessary to allow for automation.

Feedback As in most of the earlier proposed fusion models, an explicitly defined feedback process must be included. A feedback flow should be at all levels of the fusion model to ensure findings are used continuously to increase the quality of the fusion process.

Concurrent processes The fusion processes should be concurrent. By having concurrent fusion processes, we can enable independent and parallel operation, which are critical in complex systems computing large amounts of data.

Intelligence-driven The model should include the acquisition, consumption, analysis, and distribution of intelligence.

TI fusion from trusted external parties, considering that the quality of the TI may vary. There may be overlap in the provided data, and fusion of TI from various sources should be performed. The content and format of TI also vary depending on the level of TI. Therefore, the fusion of TI is essential to increase awareness.

Centralised management With requirements for a cyclic process as well as a feedback process, centralised management is preferred for managing this. Centralised management is necessary with the increasing amount of sensors.

Distributed fusion With the increasing amount of sensors and log sources

we are approaching big data. Distributing the fusion process is necessary due to problems related to big data, and especially important when designing for scalability.

Automated data reduction by feature selection

As mentioned before, FS can significantly reduce the dimensionality of data and, therefore, speed of data processing. For data-driven approach, common feature selection methods implemented in Weka were applied. These FS methods were chosen because they are peer-reviewed and implemented in Weka [19, 20, 15, 21]. We applied the following methods:

InfoGain - calculates the information gained with the attribute with respect to the class. Let H be Shannon entropy [22], c be class, and A be attribute. Information gain can then be presented as $IG(c, A) = H_c - (H_c|H_A)$. This is a filter method, and evaluates attributes in isolation from another.

Correlation-based Feature Selection (Cfs) - Correlation-based Feature Selection method proposed by Hall [23] based on the idea that feature sets of high quality contain features that are highly correlated with corresponding class.

ReliefF - ReliefF algorithm proposed by Kononenko [19, 24] which is an updated version of the Relief algorithm proposed by Kira and Rendell [13]. ReliefF takes into account attributes with strong dependencies and the distance between the examples.

To evaluate classification of the collected datasets, we decided to apply ML methods implemented in Weka [25]. Our main goal was to use community-accepted and peer-review methods that can be found in ML-related literature such as book by Kononenko et al. [19]. More specifically, we used the following classifiers: *C4.5* - a decision tree with pruning that avoid overfitting; *K-NN* - statistical learning method that predicts a class based on the nearest distance from questioned data sample to labelled one; *Naive Bayes* - a simple classifier that applies Bayes theorem assuming that features are independent, *Bayes Net* - a probabilistic method that uses conditional probabilities to build directed acyclic graph, *Random Tree* - a method that delivers the best three based on a stochastic process, *Random Forest* - uses a random set of features to build each of the random trees, *SVM* - one of the most powerful classifiers that builds a most likely separation hyperplane between the classes. To evaluate the performance of each classifier on each dataset, we used classification accuracy measure defined as following: $Acc = \frac{TP+TN}{TP+TN+FP+FN}$. This allows us to easily compare classifiers on the same dataset. Further, to avoid the overfitting of the ML model and accuracy bias by convergence to a local extrema, we used 10-fold cross-validation. This allowed us to evaluate the performance of the mentioned classifiers on unseen data with desired properties.

Proposed model for information sharing

The proposed model shown in the Figure 1 is an attempt to adhere to the previously defined requirements, and is a step towards full automation of data fusion and information sharing in the financial sector. The components are: S1-S3 - Sensors, T1-T3 - TI, Data Refinement - Sensors (L0), Data Refinement - TI (L0), Object Refinement - Sensors (L1), Object Refinement - TI (L1), Object database, Situation Refinement (L2), Threat Refinement (L3), Situational Database, Predictive Analytics Database, Information Sharing, Process Refinement (L4).

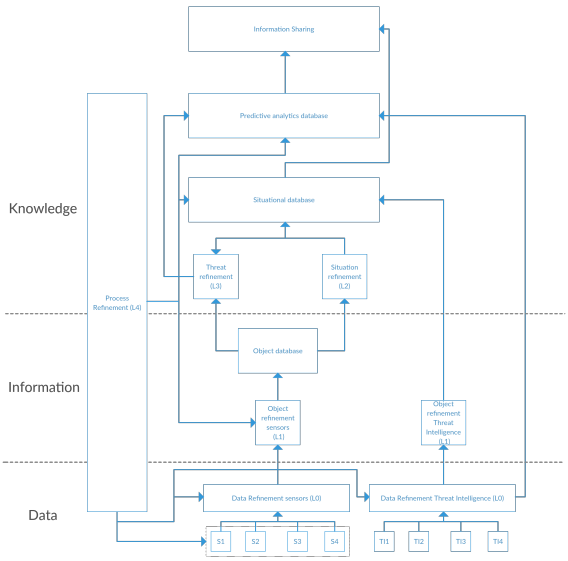


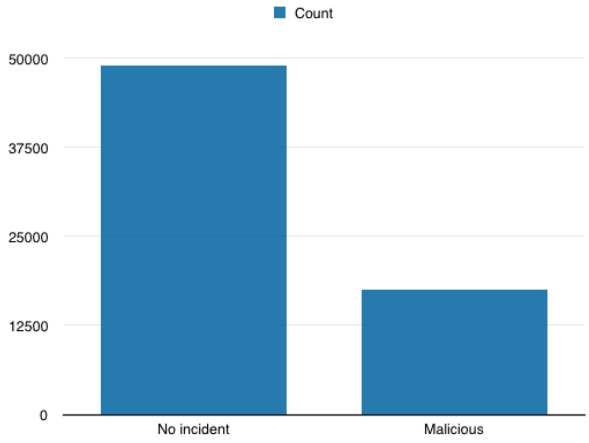
Figure 1: Data fusion model

4 Experimental setup

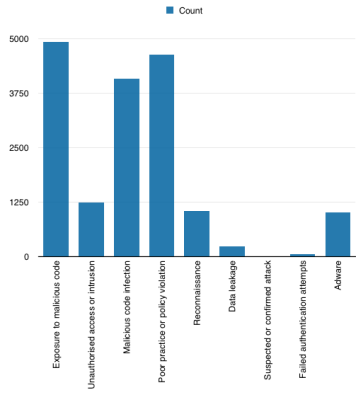
The dataset acquisition and its properties are described below. In addition, we present used hardware and software setup for the experiments.

IDS dataset acquisition

The acquisition of data is already performed by mnemonic as part of their Managed Security Service (MSS). Using advanced data fusion techniques, events from various IDSs and other information sources are aggregated and correlated. On average, around 3 billion events a day are reduced to around 2.5 million alerts. These are further aggregated and correlated for human analysts. Our dataset consist of 66,621 alerts over 60 days, which have been classified by analysts. There are in total 667 features and 10 classes as sketched in the Figure 2.



(a) Binary distribution



(b) Malicious classes

Figure 2: Distribution dissferent classes in the dataset

The features are divided into the following domains (corresponding machine labels for sharing are mentioned):

Exposure to malicious code Download of malicious code, or access to a site hosting malicious code. When sharing data on such events, elements like domain, IP, malware classification, and source country is of interest. `destination.networkAddress.address`, `properties.domain`, `attackInfo.attackIdentifier`, `destination.geoLocation.countryCode`.

Unauthorised Access or Intrusion Unauthorised users accessing system either by benign methods or exploitation. This is a successful attempt of an attacker actively avoiding implemented security measures to access unauthorised systems. When sharing data on such events, elements like source IP, access technique, and destination is of interest. `source.networkAddress.address`, `destination.port`, and `customerInfo.name`.

Malicious code infection A malicious code infections that is verified. Activity which indicates that the client or server is infected has been observed. Such activity may be e.g CC traffic, port scan, or DoS traffic. When sharing data on such events, elements like destination domain and IP, communication channel and timestamp is of interest. `destination.networkAddress.address`, `properties.domain`, `destination.port`, and `timestamp`.

Poor practice or policy violation Unsafe use of systems, or violation of company policy. The use of technologies often associated with malicious behaviour can be classified as this. E.g. use of TOR from company clients. This can also be an activity which violates the policy defined by the company. When sharing such events, elements like technology, communication channel, and destination domain and IP is of interest. `destination.port`, `protocol`, `destination.networkAddress.address`, and `properties.domain`.

Reconnaissance Activities often associated with reconnaissance activity such as port scan and automated exploitation attempt. When sharing such events, elements like technique, source IP, and destination domain and IP is of interest. `attackInfo.attackIdentifier, source.networkAddress.address, destination.port, protocol, count, destination.networkAddress.address, properties.domain`.

Data leakage Information can be leaked either by an attacker actively exploiting a vulnerability in the target system, making the system return potential sensitive information, or by target users performing actions which leak sensitive information. When sharing such events, elements like organisation, destination information, source information and technologies is of interest. `customerInfo.name, destination.networkAddress.address, properties.domain, source.networkAddress.address`, and `protocol`.

Suspected or confirmed targeted attack Such activity is often hard to detect due to its low profile. Therefore, elements interesting for sharing is often on a per case basis, however elements like organisation, techniques, technologies, and source information is some of the interesting elements. `customerInfo.name`, `destination.port`, `attackInfo.attackIdentifier`, `source.port`, `protocol`, `source.networkAddress.address`, and `properties.domain`.

Failed authentication attempts can either be attributed to a wrong username password combination, or to an attempt to access protected resources. When sharing, elements like user information, source information, destination information and technology is some of the interesting elements. `properties.ad_src__user__name`, `source.networkAddress.address`, and `protocol`.

Misconfigured device Activity related to devices functioning incorrectly. Misconfigured devices can cause network problems by not operating as expected, or by using more resources than it should. These types of events contain little information that are interesting to share.

Adware can create vulnerabilities which can be exploited by attackers. When sharing such events, elements like destination information and communication technique is of interest. `destination.networkAddress.address`, `properties.domain`, `destination.port`, and `protocol`.

No incident is the most common type of events, as current security tools produce large amounts of FP. Information regarding these types of events may be interesting to share as part of a feedback loop if the TI has been collected from external sources.

Experimental environment

This work was performed using two hardware platforms. Some preliminary testing and visualisation were performed on MacBook Air 2015. In addition, the main feature selection process and consecutive training and evaluation process were performed using HP DL360. Corresponding software used in experiments are as following. ML-aided analysis was performed using *Weka 3.6*. Pre-processing was done by Python 3.5.1, Pandas 0.7.1, Scikit-learn 0.17, Pip 7.1.2 and Logstash 2.2.

5 Results & Analysis

The results of our proposed model are described below. Data-driven approach is cross-validated using interview results.

ML-aided automated feature selection for classification

The classification performance is presented in the Table 1. For each dataset, we have applied FS method and a classifier methods. We can see that on average the CFS method provides the best result for all three datasets. It has been observed in literature that feature sets generated using CFS equalled or bettered the accuracy of using the full feature set [26], and our experimental results reflect this well. The dataset was split into four stratified folds. Experiments marked with an * (asterisk) have been performed on two stratified folds of the full dataset, i.e. 50% of the dataset, while experiments marked with ** (two asterisks) have been performed on one stratified fold of the full dataset, i.e. 25%.

Using features selected by FS methods (not shown here), we observe that number of selected features is in range 5 - 9 features. Comparing these number against the total number of features, $n = 667$, a significant increase in computational performance is expected as well. When classifying security events for decision support for analysts, it is of interest to perform this in real-time or near real-time; thus, computational performance is important. Regarding classifier performance, K-NN performed best on average with an accuracy of 93.22% with Random Forest only 0.53% points behind with an accuracy of 92.69%. However, we should note that of these only Random Forest had an increase in accuracy for all three datasets when applying CFS feature set compared to the full feature set. The highest accuracy for each dataset is in bold, while lowest is underlined. We can summarize that the dataset has good quality features with reliable separation of classes (even though

Dataset	Feature Selec.	C4.5	K-NN	Naive Bayes	Bayes Net	Random Forest	Random Tree	SVM	Average
Original	Full set	*90.93%	**91.96%	*54.06%	*67.78%	*91.49%	**89.07%	**9.97%	70.75%
	ReliefF	*90.94%	*91.43%	*61.60%	*74.41%	*95.25%	***93.88%**	*25.07%	76.08%
	InfoGain	*90.94%	*93.05%	*54.06%	*67.87%	*88.79%	*93.26%	**9.79%	58.12%
	Cfs	*91.59%	*88.22%	85.84%	*85.05%	*91.72%	*91.52%	10.59%	77.79%
Malicious	Full set	92.00%	**94.73%**	78.80%	86.84%	93.91%	*71.60%	**48.20%	80.87%
	ReliefF	91.84%	**94.73%**	73.61%	84.33%	94.53%	75.11%	37.50%	78.81%
	InfoGain	91.97%	**94.73%**	78.80%	86.85%	92.19%	68.64%	**48.12%	80.19%
	Cfs	90.96%	94.60%	85.90%	88.33%	94.29%	*84.30%	**85.07%	89.06%
Binary	Full set	*88.70%	*93.43%	*91.23%	*90.47%	**91.11%	**87.86%	**84.96%	89.68%
	ReliefF	*83.29%	*93.27%	*92.41%	*92.12%	*95.00%	**93.43%	**82.57%	90.30%
	InfoGain	*88.72%	*93.43%	*91.08%	*90.41%	*89.43%	*89.57%	**85.37%	89.72%
	Cfs	*83.29%	***95.03%**	*92.49%	*93.49%	*94.59%	*94.66%	*82.11%	90.81%
Average		89.60%	93.22%	78.32%	84.00%	92.69%	86.08%	50.78%	

Table 1: Classification results

the dependencies might be non-linear), which made K-NN achieve the greatest performance. At the same time such non-linearity causes SVM to deliver the worst results, which can be explained by pure linear nature of this method.

Trustworthy features in aggregated security events

Research interviews have been performed with 7 security experts on topics information sharing, TI, and data fusion resulting in 1-3 pages reports. Our key findings in regards to what is of most value for information sharing: *URI, IP, Domains, Detection rules, Hashes, Malware samples, Methods, Tools, Procedure.* Since the CFS method produced best results on average, we will use those features when comparing the selected feature. However, there are also elements which analysts define as important in the decision-making that is not selected by the FS.

URI can be used for detection of activities like exploit kit landing pages and callback. For an analyst, comparing two URI for determining whether the activity is an exploit kit landing page is often easy. However, this is unfortunately tough for ML classifiers without extracting features from the URI. Hence in our current experiment, URI should provide little value. However, the feature `normalizedURL` was selected by CFS on the binary dataset.

IP is often used for reputation purposes, and is a commonly shared indicator according to interview process. Observing a specific IP can indicate malware callback. Intuitively, the value of an IP feature should contribute little. However, CFS on malicious dataset selected the `destination.networkAddress.address` feature which is the destination IP. From this, we can deduct that certain IP were observed several times as either malicious or benign, and trends were observed.

Domain can also be used for reputation purposes, and is also a commonly shared indicator according to the interview process. Features related to domains were not selected by CFS in our experiment. However, domain names have previously been proved to contribute to detection of malware not only on reputation [27].

Detection rules Static and dynamic behavioural signatures like signatures for Snort, Suricata or Yara are predefined detection methods. Sharing of such signatures helps analysts avoid the time-consuming process where deep domain knowledge is often necessary. A related feature was selected in our experiments, namely `attackInfo.attackIdentifier`.

Hashes File hashes can be used for whitelisting or blacklisting of samples as it creates a unique id for each sample. For automated detection and response, such measures are simple but effective for low fruit malware. However, according to security trend reports [28, 29] threat actors often modify samples to create new unobserved hashes for each attack; therefore, hash is not as reliable as before. Such a feature is of little use in automated classification using ML methods. Our data-driven approach did not select features related to file hashes either.

Malware samples According to feedback from interviews, sharing of samples is common. Participants appeared to be willing to share samples, and saw great value in receiving such information. Features are extracted statistically and dynamically.

Methods, tools, and procedures Participants agreed on technical indicators providing some value in the detection of malicious activity.

We see that few of the elements security experts consider relevant is selected by the ML methods too. However, there are also some specific elements which were selected by the ML methods that were not mentioned by the security experts. Understanding the industry, sector, and residence country can provide much information on the threat actor.

6 Discussions & Conclusions

In this paper, we have proposed requirements for a data fusion process model enabling automation in the security operation and information sharing based on relevant literature. Further, we suggested a data fusion process model based on requirements and research interview findings. Information security experts have been interviewed in research interview process to be able to demonstrate the difference between features selected by data-driven approach and features selected by security experts. The proposed model defines how TI and sharing of TI should be included in the data fusion process, and is, therefore, a contribution towards the automation of information sharing and security operation. To the authors' knowledge, no previous fusion process models incorporate TI in the way we have proposed. We have shown that FS methods on aggregated IDS events increase the performance of automated events classification notably. The dataset of aggregated IDS events from real world networks; thus, we have demonstrated that ML classifier methods yield good results when applied to real-world data.

Later on have identified two subproblems based on the problem of IDS event classification and demonstrated how ML can solve these with acceptable performance. For each subproblem, we identified the best performing FS method as well as the best performing classifier method. More specifically, we have identified the CFS method as best performing feature selection method. Further, we identified K-NN and Random Forest as best performing classification methods. We have achieved a classification accuracy of 93.88% on the original problem, and 94.73% and 95.03% on the subproblems. This shows that ML can provide decision support in SOC. In addition to this we observed while there are some common features, there is a distinct difference between features selected by the data-driven approach and features chosen by security experts. Finally, we believe that this works is a contribution towards the much-needed automation in IDS event classification and security operation. It was bridged the gap between academia and industry by applying ML methods on real-world security events, and by performing research interviews with security experts from information security community.

References

[1] K. Julisch and M. Dacier, "Mining intrusion detection alarms for actionable knowledge," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 366–375, ACM, 2002.

[2] T. H. Nguyen, J. Luo, and H. W. Njogu, "An efficient approach to reduce alerts generated by multiple ids products," *International Journal of Network Management*, vol. 24, no. 3, pp. 153–180, 2014.

[3] P. E. Berg, K. Franke, and H. T. Nguyen, "Generic feature selection measure for botnet malware detection," in *Intelligent Systems Design and Applications (ISDA), 2012 12th International Conference on*, pp. 711–717, IEEE, 2012.

[4] Digital Shadows, "Cyber situational awareness - gain an 'attacker's eye view' of your organisation," 2015.

[5] Sqrrl Data Inc., "Sqrrl architecture." `https://sqrrl.com/product/architecture/`. Accessed April 15. 2016.

[6] NSW Government, "Nsw ict strategy, priorities: Information sharing." `www.finance.nsw.gov.au/ict/priorities/managing-information-better-services/information-sharing`. Accessed December 15. 2015.

[7] C. Kruegel, W. Robertson, and G. Vigna, "Using alert verification to identify successful intrusion attempts," *Praxis der Informationsverarbeitung und Kommunikation*, vol. 27, no. 4, pp. 219–227, 2004.

[8] K. Julisch, "Clustering intrusion detection alarms to support root cause analysis," *ACM transactions on information and system security (TISSEC)*, vol. 6, no. 4, pp. 443–471, 2003.

[9] A. Valdes and K. Skinner, "Probabilistic alert correlation," in *Recent advances in intrusion detection*, pp. 54–68, Springer, 2001.

[10] P. Langley *et al.*, *Selection of relevant features in machine learning*. Defense Technical Information Center, 1994.

[11] J. C. Schlimmer *et al.*, "Efficiently inducing determinations: A complete and systematic search algorithm that uses optimal pruning.," in *ICML*, pp. 284–290, Citeseer, 1993.

[12] H. Almuallim and T. G. Dietterich, "Learning with many irrelevant features.," in *AAAI*, vol. 91, pp. 547–552, Citeseer, 1991.

[13] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Proceedings of the ninth international workshop on Machine learning*, pp. 249–256, 1992.

[14] M. Hall, G. Holmes, *et al.*, "Benchmarking attribute selection techniques for discrete class data mining," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 15, no. 6, pp. 1437–1447, 2003.

[15] H. T. Nguyen, K. Franke, and S. Petrović, "Reliability in a feature-selection process for intrusion detection," in *Reliable Knowledge Discovery*, pp. 203–218, Springer, 2012.

[16] C. McNamara, "General guidelines for conducting research interviews." `http://managementhelp.org/businessresearch/interviews.htm`. Accessed February 1. 2016.

[17] M. Bedworth and J. O'Brien, "The omnibus model: a new model of data fusion?," *IEEE Aerospace and Electronic Systems Magazine*, vol. 15, no. 4, pp. 30–36, 2000.

[18] R. Contu and R. McMillan, "Competitive landscape: Threat intelligence services, worldwide, 2015." `http://www.gartner.com/technology/reprints.do?id=1-23HXD07&ct=141023&st=sb%29#h-d2e258`, 2014. Accessed December 10. 2015.

[19] I. Kononenko and M. Kukar, *Machine learning and data mining: introduction to principles and algorithms*. Horwood Publishing, 2007.

[20] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[21] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 8, pp. 1226–1238, 2005.

[22] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[23] M. A. Hall, *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, University of Waikato, Hamilton, New Zealand, 1998.

[24] M. Robnik-Sikonja and I. Kononenko, "An adaptation of relief for attribute estimation in regression," in *Fourteenth International Conference on Machine Learning* (D. H. Fisher, ed.), pp. 296–304, Morgan Kaufmann, 1997.

[25] "Weka 3: Data mining software in java." `http://www.cs.waikato.ac.nz/ml/weka/`. accessed: 10.09.2015.

[26] M. A. Hall, *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.

[27] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi, "Exposure: Finding malicious domains using passive dns analysis.," in *NDSS*, 2011.

[28] Verizon, "2016 data breach investigations report." `http://www.verizonenterprise.com/verizon-insights-lab/dbir/2016/`, 2016. Accessed April 27. 2016.

[29] Symantec, "Internet security threat report - volume 21, april 2016." `https://www.symantec.com/security-center/threat-report`, 2016. Accessed April 20. 2016.