A Research Process that Ensures Reproducible Network Security Research

Sebastian Abt and Harald Baier
da/sec – Biometrics and Internet Security Research Group
Hochschule Darmstadt, Haardtring 100, 64295 Darmstadt, Germany
{sebastian.abt,harald.baier}@h-da.de

***Abstract*—Access to ground-truth data is limited in network security research, especially at large-scale. If data is available, sharing is typically not possible due to privacy concerns and contractual requirements. Hence, reproducibility of research and comparability of results is difficult. For a prevailing empirical domain of research, the resulting lack of transparency is a methodological problem which especially affects network security management in practice. To address this problem, in this paper we propose a research process that ensures reproducibility by embodying both, synthetic and real-world data. Our motivation for this is to combine best of both worlds: synthetic data is used to establish ground-truth and real-world data to assure validity of results. To the best of our knowledge, no such process has been formulated until today.**

I. Introduction

The dependence of our society and economy on the Internet and computer networks is constantly increasing. In an era where launching bandwidth-exhausting attacks is just one mouse-click and little money away [1], network security management is becoming fundamental, especially when under attack. To achieve this, two important aspects of network security management are to appropriately implement the right network protection solution and to constantly validate its efficiency and effectiveness. To be able to identify the right network protection solution, experiments published in network security research must be repeatable and results reproducible. Otherwise, approaches can not be compared. To be able to constantly validate efficiency and effectiveness, access to contemporary labelled reference data sets, i.e. ground-truth data, is required. Unfortunately, such reference data is hard to find. As an attempt to quantify the unavailability of ground-truth data and its implications, we performed a retrospective analysis of data sets used in network security research published on highly-ranked security conferences between the years 2009 and 2014 in previous work [2]. In this analysis, we showed that for only 10% of the work we reviewed data sets had been publicly released together with an accepted conference paper. Furthermore, the analysis revealed that data sets provided in public data repositories are rarely being used. From these results we concluded that our community is facing a lack of available ground-truth, which we refered to as missing labelled data problem. This missing labelled data problem heavily affects network security research. Sonchack et al. [3] argue that the inability to share data and the resulting absence of ground-truth effectively hinders the scientific process and advancement. In fact, the key component of the scientific process that is affected by the inability to share ground-truth data is repeatability of research and reproducibility of results.

While we understand the difficulty of sharing real-world data, which constitutes the scientific ideal with regard to reproducibility, we also experience the increasing requirement of finding a solution for the dilemma we face. As discussed by different authors (e.g., [4], [3], [5], [6]), one way to establish ground-truth and to bridge the data gap is utility of synthetic data. It is commonly argued that synthetic data is, by definition, free of sensitive information that prohibits sharing. On the other hand, we find the prevalent persuasion in the network security community that synthetic data is insufficient for evaluation of approaches due to its lack of realism [7].

To that end, we tend to agree with both lines of argumentation, arguing for and against synthetic data. From our work in that specific field and from discussions with colleagues, however, we especially identified that the network security community is lacking a sound research process that accounts for both the requirements to *(i)* perform realistic experiments and to be able to *(ii)* repeat experiments and reproduce results. In other words: we learned that researchers do not know how synthetic data could be used in a sensible way. To close this gap, in this paper, we propose a research process that ensures reproducibility in network security research by combining real-world as well as synthetic data in the research process. To the best of our knowledge, no such research process has been published earlier at the time of writing.

The remainder of this paper is structured as follows: Section II discusses related work. In Section III we derive the contemporary research process our community follows as of today by analysing attack detection systems published in literature and discuss its shortcomings. In Section IV we describe our proposed research process that ensures reproducibility in network security research. Section V concludes and discusses future work.

II. Related Work

To the best of our knowledge, we are not aware of any paper that proposes a research process that aims at ensuring reproducibility in network security research. The lack of available data and the need of ground-truth, however, is acknowledged throughout many different papers: Most notably, Ringberg et al. [5] discuss the necessity of simulation for evaluating

978-3-901882-77-7

anomaly detectors. Especially, they argue that simulation is a prerequisite in order to maintain experimental control and propose to train and test systems using synthetic data. Afterwards, they argue that validation on real-world data should be performed to understand how an approach performs in reality. In [3], Sonchack et al. especially discuss the challenge of evaluating performance of large-scale collaborative systems. In this context, the difficulties resulting from unavailability of data multiply as not only samples from one environment are required, but from many. In earlier work [4], we give a plea for utilising synthetic data when performing machine learning based cyber-security experiments. Amongst others, we argue that synthetic data is the only viable solution towards achieving reproducibility in cyber-security research. The necessity to fill this data gap reflects in a number of papers. Recently, simulators that are capable of generating synthetic data have been proposed in [8], [6], [9], [10]: Lindauer, Wallnau et al. [8], [6] propose an approach that generates insider threat data by simulating user activity following a "drama as data" concept. In [9], Sonchack et al. describe the host behaviour simulator LESS that is capable of automatically deriving and configuring host behaviour from background traffic. LESS is tailored towards large-scale simulations and can be used to generate synthetic traffic records. The technique used to generate synthetic data here is superposition of different traffic sources and has been used in different earlier work [11], [12], [6]. A different approach is followed in work proposed by Abt et al. [10]. In that work, the complex node automation framework *cnaf* is introduced, which provides API-bindings to maintaining virtual machine clusters and to automation of applications (e.g., sending/receiving emails, web browsing, instant messaging) for the Python programming language. An earlier approach to generating labelled data sets by means of simulation is the DARPA IDEVAL project [13]. However, the DARPA IDEVAL data sets date back to 1999 and do not contain any relevant information of today's networks. An approach towards compiling a contemporary state-of-the-art data set is described by Song et al. [14]. Based on our observations, simulation and data synthesis are not broadly used in the network security domain. Broader support for simulation is found in the general network research (e.g. network protocol development) community. As network security research often involves capturing traces on the network level, utilising the toolchains of these communities may be an option. Typically, these simulators employ discrete event simulation (DES) [15]. Prominent DES-based toolchains utilised in network research, which may as well be used to generate synthetic data, are: ns-2 and its successor ns-3 [16], OMNeT++ [17] and TOSSIM [18].

III. The Contemporary Process

A. Invariant Attack Detection

Research in network security is an arms race with miscreants. As a result, a plethora of detection and mitigation approaches has been published by our community over time (cf. [19], [20]). Very often, these approaches are centred around machine-learning (ML) techniques. As detection of network attacks can essentially be understood as classification problem, the commonly encountered use of ML techniques is unsurprising. When reviewing different approaches, we identify a common invariant architecture of attack detection systems as illustrated in Figure 1. This invariant is made up of three phases:

1) In a learning phase, an algorithm is trained. More specifically, in that phase relevant features are extracted and classification models are computed using these features. Feature extraction and model building typically require human inspection of the input data. Depending on the specific ML technique, a labelled reference data set may be required (path T.1a) in that phase or not (path T.1b). As a result of this phase, a description of the features as well as resulting models is typically persisted in a knowledge base.

2) In a detection phase, the description of features is used to extract features from previously unseen input data. Using the previously generated models, feature vectors are classified by applying a specific ML technique. As a result, events may be observed in case an attack has been detected. In some cases, these events are filtered using additional external information in order to reduce false alarms.

3) Finally, in a reaction phase, alerts are generated and specific actions may be triggered to mitigate the attack.

B. Research Process

From this invariant in system design, we try to derive and model the research process that is currently established in our community. Specifically, we infer this process by reflecting the invariant described in Section III-A to content required in research papers. Our motivation behind this proceeding is as follows: we hypothesise that the primary motivation of work of a researcher is to successfully *publish* a solution to a specific problem as this improves reputation and counts towards tenure and graduation. For that, a paper has to convince reviewers that the problem to be solved is real, the solution is either novel or outperforms existing approaches using a relevant performance measure and argumentation is logically sound. Hence, in that domain, a successful paper has to cover at least the following topics:

- Problem to solve
- Available data and collection process
- Model and ML technique applied
- Relevant features
- Experimental environment and conditions
- Evaluation results

Consequently, we expect researchers to devote their time to these specific topics during research.

In order to be able to provide meaningful content for the above given topics, we expect researchers to solve the data availability problem first. Generally, data can already be available within research groups or has to be acquired first. The analysis we performed in earlier work [2], however, showed that data is individually collected in most cases. Hence, we expect a significant amount of time in the research process

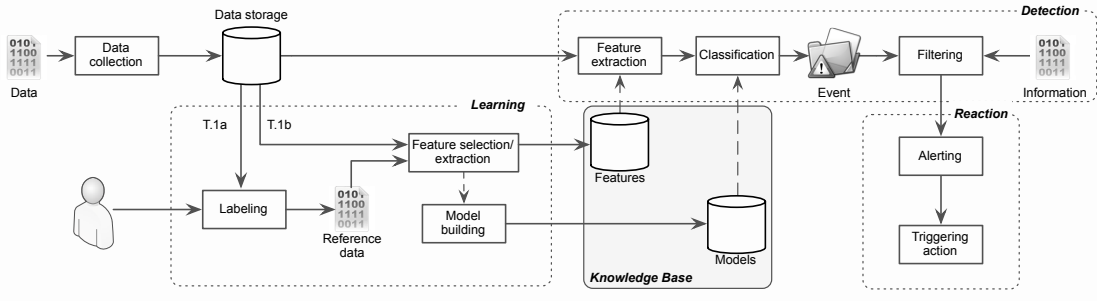

Figure 1. Invariant architecture of attack detection systems commonly found in network security literature.

to be devoted to finding environments that can provide access to relevant data, agreeing on terms and conditions on how to use these data, as well as developing components for data collection. If successful, the result of this step is a real-world data set that can be used for experiments and evaluation. Before, however, analysis and post-processing of the data is typically required to develop a sound understanding of the characteristics of the available data and information covered within the data, i.e. knowledge about the problem instance and its specifics. Relying on the insights gained during this exploration, a researcher can choose appropriate ML techniques and extract relevant features from the input data in order to build the classification models by performing different experiments. This actually constitutes the creative and, in best case, innovative step in the research process and, obviously, may take long time. Typically, this exploration and knowledge discovery period is circular in the sense, that the whole process is iterated as new insight is gained during the process and until the experimental results are convincing. Afterwards, the resulting approach is typically evaluated using data previously unused and relevant performance measures are computed. In the end, the whole process and the results are documented in a research manuscript and submitted to a conference or journal for review.

This whole process is illustrated in Figure 2. In this figure, rectangular boxes represent steps in the process that require engineering (i.e., data collection, feature extraction, experimentation), parallelograms represent resulting data (i.e., real-world data, real-world features, real-world results) and elliptical boxes reflect rather scientific process steps (i.e., exploration, knowledge). While we understand that the data exploration and knowledge discovery steps are central from a scientific perspective, both steps are hard to quantify and especially are difficult to reproduce. Hence, from our point of view of developing a research process that ensures reproducibility in network security research, both process steps play a tangential role. Instead, we identify three main tasks in this process: ❶ data collection, ❷ feature extraction, and ❸ experiment.

C. Discussion

The contemporary research process we derive from a design invariant identified in existing work typically focuses on a single available data set. As has been investigated in [2], this data set is most commonly manually compiled in real-world. Simulation and synthetic data as well as existing public data archives are rarely used. As argued by Sonchack et al. [3] and Ringberg et al. [5], single-source real-world data typically do not reflect all characteristics relevant for developing large-scale intrusion detection or anomaly detection systems. Especially not if collaborative systems have to be evaluated. The reason for this lies in the fact that data is typically collected at a single vantage point only and, hence, information available is highly dependent on the location of the vantage point (e.g., corporate view vs. Internet service provider view) as well as collection method (e.g., information available in Netflow vs. pcap). On the other hand, data collection in enterprise environments is typically granted only after signing restrictive non-disclosure agreements. The reason for this is trivial: enterprises fear the loss of reputation if information about vulnerabilities and other weaknesses are disclosed. Additionally, enterprises need to adhere to data protection law and are required to honour and protect privacy of employees and customers. The latter requirements are especially reasons why data collection at Internet service providers is difficult. As a consequence, obtaining real-world data is very time consuming and data sharing is hindered. Given these restrictions, ensuring reproducibility in the network security research domain is almost impossible.

IV. A Reproducibility-Ensuring Process

To overcome the implications and limitations arising from the contemporary research process, in this section we propose a novel research process that embodies simulation and synthetic data in order to ensure reproducibility of network security research.

As argued in related work (e.g., [4], [3], [5]), we are convinced that use of synthetic data is currently the only viable approach towards achieving reproducibility and experimental control in network security research at large-scale. On the other hand, synthetic data is hardly used in contemporary research [2]. We believe the reason for this to be twofold:

1) Quality of synthetic data is often challenged. Specifically, it is typically not known how results achieved on synthetic data relate to real-world. Synthetic data is usually generated by simulation and simulation relies on specific models of real-world. As these models hide specific aspects of reality in order to be able to terminate simulation in finite time, synthetic data

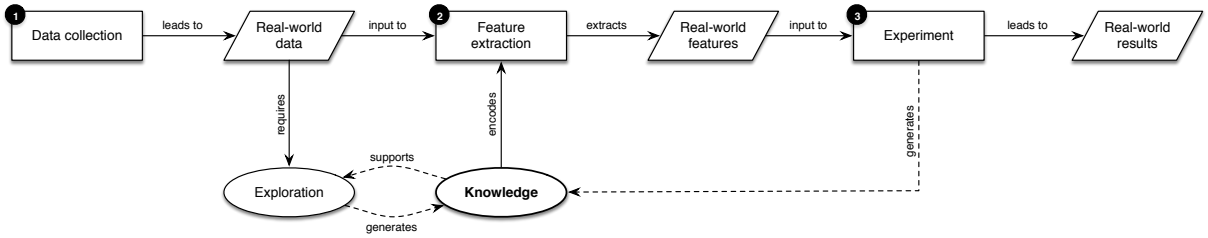

Figure 2. Illustration of the contemporary network security research process. Rectangular boxes represent engineering steps, parallelograms represent data and elliptical boxes represent scientific steps. Solid lines denote data flow, while dashed lines denote knowledge transfer.

inherently cannot contain the variety of subtleties found in real-world [7]. Furthermore, synthetic data is often assumed to contain artefacts, such as periodicity or determinism, that may not be found in real-world, but heavily influence classification accuracy. Interestingly, however, as of today no proof exists that shows that synthetic data in fact can not be used.

2) As use of simulation and synthetic data seems not to be widely encouraged in the network security community as of today, we notice a general lack of adequate synthesis toolchains. As discussed in Section II, this situation is much better in other disciplines (e.g., network protocol development and testing). Recently, work has been published towards developing specific simulators for network security research (e.g., [10], [9], [6]). We believe that this evolution supports our hypothesis, but are convinced that this process is still at its infancy.

We hypothesise that the second reason is a direct consequence of the first reason. For the first reason, however, we have no final explanation as to why it has manifested itself in our community for many years, without seeing any attempt to proof or falsify it. During our work towards finding an answer, we realised that our community is missing a description of an adequate research process that guides researchers on how to incorporate synthetic data and how to proof utility and quality of synthetic data. With this work, we aim at closing this gap.

A. Research Process

The research process we propose directly builds upon and extends the contemporary process as described in Section III-B. As this process constitutes the de-facto standard in our community and is what researchers are used to, we assumed it would be wise to base our work on that. When designing our process, our motivation was to introduce simulation and synthetic data in a non-intrusive, natural way. Especially, our aim was to re-use existing process steps best possible in order to reduce the amount of additional work as time-to-publication is a relevant metric, specifically for young and tenure-track researchers. Nonetheless, we are aware that the research process we propose, if completely followed, will require additional resources – either time or manpower. This is the tradeoff for achieving reproducibility.

The process we propose is depicted in Figure 3. The notation and interpretation of symbols is equivalent to that described in Section III-B. Additionally, we introduce utility and quality checks, denoted with diamond shapes. In contrast to the contemporary process, our process has six process steps: ❶ data collection, ❷ simulation, ❸ feature extraction, ❹ utility control, ❺ experiment and ❻ quality control. Initially, the process starts again with data collection in real-world. This data is explored in order to gain specific knowledge about a problem's characteristics and to identify relevant features. Based on that knowledge, in contrast to the contemporary process, we propose not only to develop feature extractors, but also to configure simulation in order to generate synthetic data. This is a fundamental difference to the contemporary process. However, we also believe that this is a very obvious proceeding. If the knowledge about the problem, the data set's specifics as well as the whole environment is already developed, transferring it into simulation is very straight forward - assuming availability of appropriate simulators. If this step is completed, relevant features can be extracted from available real-world data as well as from synthetic data, leading to real-world as well as synthetic feature vectors, respectively. At that point in the process, we propose a first utility check to ensure quality of synthetic data at feature vector level. If the comparison of real-world and synthetic features yields significant differences, the exploration, knowledge discovery and simulation process steps should be repeated, as some basic misunderstanding of the underlying concepts can be assumed. So, in fact, by including simulation and utility checks in the process, we not only work towards reproducibility, but also explicitly introduce a methodology that enables researchers to assess the knowledge gathered during exploration at an early phase, i.e. it enables self-control.

If the utility check is passed, the experiment can be performed on both real-world as well as synthetic feature vectors. This step, again, is a straight-forward extension of the equivalent step found in the contemporary research process. The fundamental difference is that all experiments have to be performed twice: on, both, real-world and synthetic data. As outcome, we obtain results on synthetic and real-world data and introduce an additional quality control step in order to conclude the synthetic data's quality. If results are good enough, the developed approach as well as real-world results can be published. In contrast to the contemporary process, however, the synthetic results, synthetic data as well as synthetic feature vectors can be published as well. By following this process, reproducibility of network security research is granted as

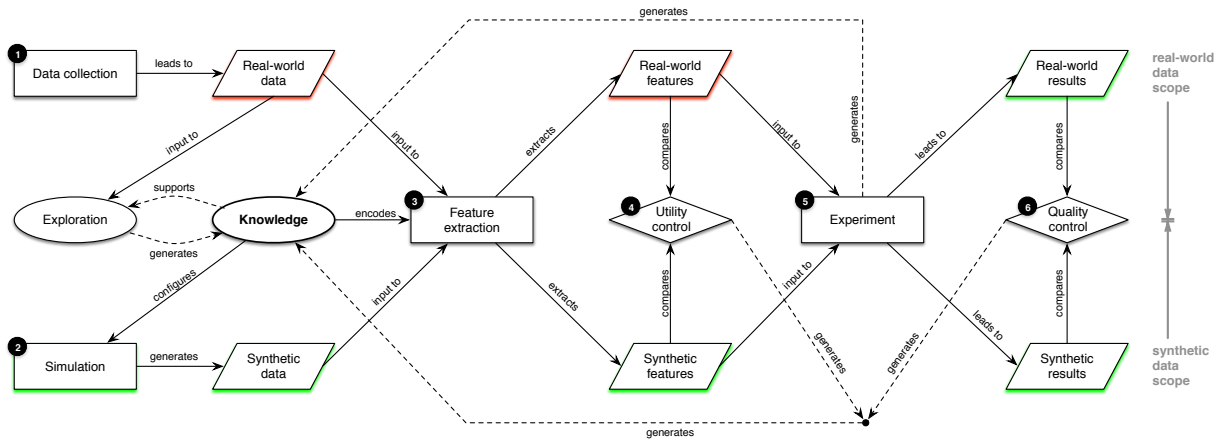

Figure 3. Illustration of our proposed research process that ensures reproducible network security research. Rectangular boxes represent engineering steps, parallelograms represent data, elliptical boxes represent scientific steps and diamond-shaped boxes represent quality gates. Solid lines denote data flow, while dashed lines denote knowledge transfer. Green marked boxes can be released together with the paper, red marked boxes may remain private.

synthetic data is free of sensitive information and, thus, can be shared without restrictions. Consequently, synthetic data serves as link for comparison of competitive approaches.

B. Discussion

As can be seen from the above description, our process aims at combining the best of real-world and synthetic data: synthetic data are used as link to ensure reproducibility. Real-world data, on the other hand, is used not only to demonstrate that a specific approach indeed works in reality, but also to assure quality of synthetic data. This quality assurance is not only required in order to convince reviewers, but, unfortunately, also not trivial. To the best of our knowledge, quality assurance of synthetic data is not intensively discussed in our community. In fact, we are not aware of any published results in that direction. In Section IV-C we discuss one approach to utility and quality control, each. Nevertheless, we would like to note that this is just an initial concept which we have to explore in future work in greater detail.

Additionally, we realise that our process greatly benefits from data synthesis and simulation toolchains. As mentioned earlier, currently, research in that direction is sparse.

C. Quality of Synthetic Data

As discussed in the previous section, proving quality of synthetic data is a fundamental requirement in order to convince our community of the validity of results achieved using synthetic data and utility of synthetic data in general. In the process we propose, at least initially, we rely on real-world data for this task. Essentially, we introduce two quality gates that aim at assuring quality at two different layers. These additional checks introduced in our process directly follow from the definition of the missing labelled data problem we gave in [2]: A utility check at feature level is introduced in order to proof statistical similarity of synthetic data and real-world data and, hence, to show utility of the synthetic data set. Additionally, quality control is enforced at the result level in order to proof that equivalent results can be reached at the end of an experiment, regardless of whether synthetic or real-world data are used. Our intention of these checks as well as initial concepts on how these checks can be performed are further presented briefly in Section IV-C1 and Section IV-C2.

1) Utility Control: Assessing similarity of raw synthetic and real-world data at bit level is computationally expensive due to complexity and size of data sets typically dealt with in the network security domain (e.g., network traces). And even if one were able to compute similarity of two data sets at that level, semantics of data are typically not known as the state of all communicating devices is typically not available. Neither are researchers familiar with all specific peculiarities that may be found in data. Hence, we believe that similarity of two data sets can only be approximated for particular cases and at a higher level. In our process, we refer to that higher-level approximation as utility control. As network security research often involves ML techniques (cf. Section III-A), a natural level of abstraction for our utility control is the feature level. Features are extracted from input data in order to gain a specific view on the data set at hand. Irrelevant aspects are removed and relevant characteristics are typically encoded after applying transformations. At that level, a researcher is capable of analysing and understanding a specific data set and its phenomena. Consequently, we propose to perform statistical analysis and comparison of real-world features and synthetic features as utility control. If synthetic and real-world features are statistically similar, it is safe to assume that a ML algorithm is capable of learning the same concept from data. Thus, achieving high similarity in statistical comparison at feature level is assumed to be a predictor for equivalence of experiments. On the other hand, while dissimilarity at feature level may nevertheless lead to equivalent results of experiments (e.g., if relevant characteristics are not well represented in selected features), we encourage researches to repeat data exploration and knowledge discovery as some fundamental

misunderstanding may be prevalent.

However, it is important to note that by performing utility control at feature level, quality of synthetic data is not assured in general. Instead, this utility check just settles ground for quality assessment of a specific interpretation of the synthetic data set at hand and needs to be correlated to the final quality control step as proposed in our process.

2) Quality Control: The final quality control step aims at assuring quality of a specific interpretation of synthetic data, i.e. depending on the selected features. We assume that it is neither feasible, nor possible to assure quality of the whole synthetic data set generated in process step two as this may by far exceed the knowledge and experience of a researcher. Also, we believe that it is not sensible to try to assess the quality of any possible set of features not covered by the utility control in process step four. Hence, if the set of extracted features changes, utility and quality control have to be repeated as well in order to assure quality.

The aim of the quality control step is to assess the validity of results achieved using synthetic data. That means, this step aims at demonstrating that an approach under development achieves similar results, independent of being evaluated using real-world or synthetic data. To achieve this, we interpret quality as probability Q: Let $\mathbf{A}$ denote the set of all known approaches that solve a given problem and $a \in \mathbf{A}$ denote a specific approach. Furthermore, let $p_r(a) \in [0,1]$ denote the performance of a as measured on real-world data and $p_s(a) \in [0,1]$ denote the performance of a as measured on synthetic data. Then,

$$Q[p_r(a) - \epsilon \leq p_s(a) \leq p_r(a) + \epsilon] \tag{1}$$

denotes the probability that $p_s(a)$ falls within ϵ-environment of $p_r(a)$, i.e. that $p_s(a)$ is ϵ-close to $p_r(a)$. In term (1), $0 \leq \epsilon \leq 1$ models the tolerance we accept the performance on synthetic data to deviate from performance on real-world data. As probability, $0 \leq Q \leq 1$ by definition. Hence, in order to actually control quality, we need to bound term (1) by a specific threshold $0 \leq \delta \leq 1$. If

$$Q[p_r(a) - \epsilon \leq p_s(a) \leq p_r(a) + \epsilon] > \delta,$$

we say that our synthetic data suffices quality δ with tolerance ϵ. Obviously, the larger valued δ and the smaller valued ϵ, the higher the quality of the synthetic data. If this requirement holds true for all $a \in \mathbf{A}$, i.e. if

$$\forall a \in \mathbf{A} : Q[p_r(a) - \epsilon \leq p_s(a) \leq p_r(a) + \epsilon] > \delta,$$

then we conclude that the synthetic data set indeed is of high quality.

As mentioned initially, the utility and quality control concepts we sketch here are still preliminary and at its infancy. Nevertheless, we are convinced that it is valuable to share ideas towards assessing quality of synthetic data as we notice a dearth of ideas and approaches in that direction. As an interesting question for future research we need to identify which subset $\mathbf{A}' \subseteq \mathbf{A}$ suffices to obtain this conclusion. Especially, we need to identify the confidence of this conclusion for $|\mathbf{A}'| = 1$, which would be the typical case when following our approach.

V. Conclusion and Future Work

In this paper, we present a research process that ensures reproducible network security research by embodying simulation and synthetic data in the research process. Our motivation for this is to combine best of both worlds: synthetic data is used to establish ground-truth and can serve as link for rigorous scientific comparison and peer-review. Real-world data on the other hand is used to demonstrate utility of the approach and can serve as quality benchmark for synthetic data. Especially the latter is an important aspect, as utility of synthetic data and validity of synthetic data is widely challenged in our community. Nevertheless, we are convinced that using synthetic data is the only viable approach towards achieving experimental control and reproducibility in network security research. To the best of our knowledge, we are the first to propose a research process with comparable capabilities.

As quality assurance of synthetic data is an important topic, we introduce two quality gates into our process. These quality gates have the capability to assure that for a specific interpretation of the data, synthetic data achieve results comparable to real-world data. To achieve this, we briefly sketch concepts on how to perform utility control and how to control quality. However, we acknowledge that the work on quality assurance is still at its infancy and requires future work. At the time of writing, we are not aware of any other concepts on assessment of synthetic data quality. We believe that this work is very essential to gain trust in synthetic data based research.

We are aware that the process we sketch here is a vision for the future, which especially causes significant additional work for early adaptors. Successful application requires not only additional research on each of the process steps, but also a change in how our community reviews research and acknowledges efforts in data synthesis and data sharing. If applied, however, we are convinced that this process heavily impacts the network security domain as it increases repeatability of research and reproducibility of results and, consequently, transparency of approaches. We believe that especially the latter boosts the transfer of research to real-world and supports proper network security management: if approaches published in research can easily be compared and benchmarked, appropriate network protection solutions can be selected. If open synthetic reference data is generated, efficiency and effectiveness of network protection solutions can be validated. Both aspects of network security management, i.e. selection of the appropriate network protection solution as well as constant validation of its efficiency and effectiveness, are important and difficult to achieve today.

VI. Acknowledgments

This work has been funded by the German Federal Ministry of Research and Education (BMBF) under grant number 03FH005PB2 (INSAIN) and supported by CASED.

REFERENCES

[1] J. J. Santanna and A. Sperotto, "Characterizing and mitigating the DDoS-as-a-service phenomenon," in *Proceedings of the 8th IFIP WG 6.6 International Conference on Autonomous Infrastructure, Management, and Security (AIMS 2014)*. Springer, 2014, pp. 74–78.

[2] S. Abt and H. Baier, "Are We Missing Labels? A Study of the Availability of Ground-Truth in Network Security Research," in *Proceedings of the 3rd Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*, ser. BADGERS '14. IEEE, September 2014.

[3] J. Sonchack, A. J. Aviv, and J. M. Smith, "Bridging the Data Gap: Data Related Challenges in Evaluating Large Scale Collaborative Security Systems," in *6th Workshop on Cyber Security Experimentation and Test (CSET 13)*. Berkeley, CA: USENIX, 2013. [Online]. Available: https://www.usenix.org/conference/cset13/workshop-program/presentation/Sonchack

[4] S. Abt and H. Baier, "A Plea for Utilising Synthetic Data when Performing Machine Learning Based Cyber-Security Experiments," in *Proceedings of the 2014 Workshop on Artificial Intelligence and Security*, ser. AISec '14. ACM, December 2014, pp. 37–45. [Online]. Available: http://doi.acm.org/10.1145/2666652.2666663

[5] H. Ringberg, M. Roughan, and J. Rexford, "The Need for Simulation in Evaluating Anomaly Detectors," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 1, pp. 55–59, Jan. 2008. [Online]. Available: http://doi.acm.org/10.1145/1341431.1341443

[6] K. Wallnau, B. Lindauer, M. Theis, R. Durst, T. Champion, E. Renouf, and C. Petersen, "Simulating Malicious Insiders in Real Host-Monitored User Data," in *7th Workshop on Cyber Security Experimentation and Test (CSET 14)*. San Diego, CA: USENIX Association, Aug. 2014. [Online]. Available: https://www.usenix.org/conference/cset14/workshop-program/presentation/lindauer

[7] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *Security and Privacy (SP), 2010 IEEE Symposium on*. IEEE, 2010, pp. 305–316.

[8] B. Lindauer, J. Glasser, M. Rosen, K. Wallnau, and ExactData, LLC, "Generating Test Data for Insider Threat Detectors," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, vol. 5, no. 2, pp. 80–94, 2014.

[9] J. Sonchack and A. J. Aviv, "LESS Is More: Host-Agent Based Simulator for Large-Scale Evaluation of Security Systems," in *Computer Security-ESORICS 2014*. Springer, 2014, pp. 365–382.

[10] S. Abt, R. Stampp, and H. Baier, "Towards Reproducible Cyber-Security Research through Complex Node Automation," in *Proceedings of the 2015 Workshop on Reproducibility in Computation Based Research*, ser. RCBR '15. IEEE, 2015.

[11] Z. B. Celik, J. Raghuram, G. Kesidis, and D. J. Miller, "Salting public traces with attack traffic to test flow classifiers," in *4th Workshop on Cyber Security Experimentation and Test (CSET 11)*, 2011.

[12] D. Brauckhoff, A. Wagner, and M. May, "FLAME: A Flow-Level Anomaly Modeling Engine," in *1st Workshop on Cyber Security Experimentation and Test (CSET 08)*, 2008.

[13] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, "The 1999 DARPA off-line intrusion detection evaluation," *Computer networks*, vol. 34, no. 4, pp. 579–595, 2000.

[14] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, and K. Nakao, "Statistical Analysis of Honeypot Data and Building of Kyoto 2006+ Dataset for NIDS Evaluation," in *Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*, ser. BADGERS '11. New York, NY, USA: ACM, 2011, pp. 29–36. [Online]. Available: http://doi.acm.org/10.1145/1978672.1978676

[15] J. Banks *et al.*, *Handbook of simulation*. Wiley Online Library, 1998.

[16] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, and J. Kopena, "Network simulations with the ns-3 simulator," *SIGCOMM demonstration*, vol. 15, p. 17, 2008.

[17] A. Varga *et al.*, "The OMNeT++ discrete event simulation system," in *Proceedings of the European simulation multiconference (ESM'2001)*, vol. 9. sn, 2001, p. 65.

[18] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: Accurate and scalable simulation of entire TinyOS applications," in *Proceedings of the 1st international conference on Embedded networked sensor systems*. ACM, 2003, pp. 126–137.

[19] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in Cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.

[20] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir, "A survey of botnet technology and defenses," in *Conference For Homeland Security, 2009. CATCH'09. Cybersecurity Applications & Technology*. IEEE, 2009, pp. 299–304.