Automation of DDoS attack investigation in industrial control systems using Bayesian networks on Python

Valeriy Lakhno[1,*,†], Miroslav Lakhno[2,†], Olena Kryvoruchko[3,†], Serhii Kaminskyi[3,†] and Vadym Makaiev[1,†]

[1] National University of Life and Environmental Sciences of Ukraine, 15 Heroiv Oborony str., 03041, Kyiv, Ukraine
[2] e-Docs.UA, 21A Degtyarivska str., 04119, Kyiv, Ukraine
[3] State University of Trade and Economics, 19 Kyoto str., 02156 Kyiv, Ukraine

Abstract

This paper investigates the possibility of using Bayesian Networks (BN) to analyze and confirm the involvement of a specific computer in a DDoS attack on industrial control systems (ICS). The primary focus is on developing a Python software product that automates the calculation of probabilistic estimates from the collected evidence to confirm various hypotheses about the seized computer's involvement in a DDoS attack. Automation of the analysis through the developed Python software product will eliminate subjective errors and bias, speed up data processing, and ensure objective conclusions based on the available evidence. The hypotheses and corresponding evidence related to the use of BN for modeling complex relationships between events during the execution of DDoS attacks from the suspect computer are considered. It is shown that the proposed approach facilitates more in-depth and accurate analysis of cybercrimes related to DDoS attacks and can significantly improve the investigation processes and decision-making in ensuring the security of ICS.

Keywords

industrial control systems, DDoS attacks, investigation, evidence analysis, Bayesian network, Python

1. Introduction

In the modern digital world, where more aspects of life are transitioning online, cybercrime and cybersecurity have become urgent problems hindering societal development. These problems require adequate solutions through the collective efforts of specialists in various fields, from IT to law, since many cybercrimes, such as DDoS attacks on computer systems and networks (CSN), can have significant consequences for individuals, organizations, and even states [1, 2]. The scenarios used by cybercriminals are quite creative and constantly evolving, making cybercrime increasingly sophisticated and complex.

As demonstrated in [3, 4], DDoS attacks pose a significant danger to industrial control systems (ICS). These systems are often used in enterprises and critical infrastructure such as energy, water supply, transport, and manufacturing. Attacks on ICS, including DDoS attacks, can lead to severe consequences, such as operational disruptions, economic losses, and threats to human safety. For instance, in 2013, an attack targeted the U.S. water supply systems [5]. The attack could have caused equipment failures controlling water distribution and wastewater treatment, posing a serious public health threat. Cybersecurity specialists managed to prevent such a scenario at an early stage of the attack's development. In 2016, a DDoS attack targeted the railway management systems in Sweden [6]. The attack caused system disruptions, leading to train delays and cancellations. In 2017, a DDoS attack on a semiconductor manufacturer caused failures in their production management system, resulting in significant production delays and economic losses. Even this brief overview demonstrates that DDoS attacks pose a serious threat to ICS, disrupting their normal operation and causing significant negative consequences. These attacks can halt production processes, lead to economic losses, and even pose safety threats [7]. Therefore, in this paper, we investigate the possibility of developing a Python software product that, based on the mathematical apparatus of Bayesian Networks (BN), helps automate the analysis and calculation of probabilistic estimates from collected evidence to confirm or refute hypothesis. Such a tool will be extremely useful for the effective investigation of DDoS attacks, facilitating the work of specialists and improving the accuracy of conclusions.

A key role in investigating unauthorized interference in CSN, such as organizing DDoS attacks, is the search for evidence in the non-material (digital) environment. From a software-technical perspective, the elements of CSN during an investigation at the site of a potential cyberattack, such as a DDoS attack, require extreme caution, considering

CPITS-II 2024: Workshop on Cybersecurity Providing in Information and Telecommunication Systems II, October 26, 2024, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
lva964@nubip.edu.ua (V. Lakhno);
valss725@gmail.com (M. Lakhno);
olena_909@ukr.net (O. Kryvoruchko);
s.kaminskyj@knute.edu.ua (S. Kaminskyi);
makaiev.vadym@gmail.com (V. Makaiev)

0000-0001-9695-4543 (V. Lakhno);
0000-0001-6979-6076 (M. Lakhno);
0000-0002-7661-9227 (O. Kryvoruchko);
0000-0002-4884-1517 (S. Kaminskyi);
0009-0008-5561-4508 (V. Makaiev)

factors such as the large volume of electronic information, the presence of intellectual property rights on parts of the information, hidden data inaccessible to the regular computer user, and the risks of data loss due to careless actions or the potential programmed automatic execution of data destruction algorithms.

As demonstrated in [8, 9], using the BN apparatus to prove the involvement of a specific computer in a DDoS attack is a powerful tool. Bayesian Networks (BN) allow for modeling complex cause-and-effect relationships between various aspects of digital evidence and drawing substantiated conclusions based on available data. This is especially important for establishing the fact of a specific computer's involvement in carrying out a DDoS attack, which requires analyzing numerous factors and probabilities. As shown in [8], BN can effectively integrate data from various sources, including network activity logs, system configurations, and user information, significantly enhancing the accuracy and reliability of investigations. Thus, the use of BN in cybercrime investigations opens new prospects for improving the efficiency and reliability of identifying participants in DDoS attacks. All the above has prompted our interest in this topic.

2. Methods and models

A crucial aspect of finding and securing digital (electronic) evidence is adhering to the "best evidence rule" [10–14]. Compliance with this principle depends on using specialized knowledge in collecting electronic evidence, which IT specialists possess. This helps safeguard data from accidental deletion or damage and prevents cases of programmed self-destruction of files, for example, when an incorrect password is entered into the directory. Given the above, when searching for digital evidence, it is important to consider the identified evidence, such as tools for executing DDoS attacks. Suppose, during the investigation of a DDoS attack, a computer suspected of carrying out the attacks was seized. During the analysis of this computer's contents, specialized programs (Low Orbit Ion Canon, HULK, PYLORIS, TORS HAMMER, etc.) or scripts for launching DDoS attacks may be found. The work history or logs may contain records of launching tools commonly used for DDoS attacks and connections to command servers used to manage botnets.

The development of the research outlined in [8] involves creating a practical Python-based software product. This product will automate the calculation of probabilistic estimates of collected evidence to confirm hypotheses based on the mathematical apparatus of Bayesian Networks. This program will significantly simplify the work of both IT specialists and forensic investigators involved in investigating DDoS attacks, providing accurate and reliable results comparable to those obtained with the GeNIe package.

Python is one of the most popular programming languages due to its simplicity and readability, allowing for the quick and efficient development of complex algorithms. Additionally, Python has a rich set of libraries and frameworks for statistical analysis, machine learning, and working with Bayesian Networks. For example, libraries such as pgmpy (used in our product), scikit-learn, PyMC3, and networkx provide powerful tools for building, training, and visualizing BN. This greatly simplifies the development process and allows focusing on solving specific tasks rather than creating tools from scratch.

The development environment used was PyCharm, one of the most powerful and convenient development environments for Python, offering many tools that simplify the writing, debugging, and testing of code. It is worth noting that Python and PyCharm run on all major operating systems (Windows, macOS, Linux), ensuring the possibility of developing and using the program across different platforms. In our view, using Python and PyCharm to develop a software product automating evidence analysis with BN provides the optimal combination of convenience, power, and flexibility. This allows the creation of efficient, reliable, and easily maintainable solutions for cybersecurity tasks, including investigating DDoS attacks on ICS.

The main hypothesis (*H*), see Table 1 and Fig. 1, according to assumption (*H_DDOS_T* arg *et*), is that the seized computer could have been used to carry out a DDoS attack on the target CSN. This hypothesis may include at least two sub-hypotheses. *H*1 is that the seized computer was used to gain access to the target CSN, a *H*2 is that the seized computer was used to organize the DDoS attack. Evidence (*E*) for each sub-hypothesis might include, for example, the presence of the target CSN's IP address on the seized computer or the matching of the seized computer's IP address with the attacker's IP address identified by the provider.

Presenting the BN structure as shown in Fig. 1 offers many advantages. For example, visualization helps to more easily understand the complex probabilistic relationships between hypotheses and evidence. The connections between nodes (hypotheses and evidence) are visible, facilitating understanding of the structure and logic of reasoning. The graphical representation allowed us to intuitively evaluate the influence of each piece of evidence on the sub-hypotheses and the main hypothesis. In general, such a software product will help experts and users better understand the basis of their decisions and how various pieces of evidence affect the hypothesis's probability. This will contribute to more reasoned and confident decisions in investigating such crimes. It is worth noting that graphical representation makes the information accessible to a wide audience, including those who may not have in-depth knowledge of mathematics and statistics. This facilitates discussion and explanation of conclusions among team members and stakeholders. Additionally, visualization helps identify gaps in the data and dependencies that may require further investigation or data collection, contributing to a more comprehensive and detailed analysis of the situation.

For implementing the Python program, we structured sub-hypotheses and corresponding evidence for the main hypothesis (see Table 1).

From a legal perspective, seized objects (computer equipment and its components) are considered potential sources of evidence, and any unprofessional actions involving them may result in the loss or inadmissibility of such evidence. In this regard, a well-justified position emphasizes the need for advanced specialized training for investigators involved in cybercrime investigations, aligned with modern challenges and the future development of the information technology sector.

Table 1

Structuring Sub-hypotheses and Evidence in the Python Product for Automating Analysis and Probabilistic Estimates Calculation of Collected Evidence for Hypotheses Confirmation Based on the Mathematical Apparatus of Bayesian Networks

Main Hypothesis *H*: A seized computer was used to launch a DDoS attack on the target computer	
Sub-hypothesis *H*1: The seized computer was used to access the target computer Evidence for Sub-hypothesis *H*1: *E*1: The IP address of the target computer was found on the seized computer. *E*2: The URL address of the target computer was found on the seized computer. *E*3: The IP address of the target computer matches the access IP address (as specified by the provider). *E*4: Log entries of access to the target computer at the relevant time were found.	Sub-hypothesis *H*2: The seized computer was used to conduct the DDoS attack Evidence for Sub-hypothesis *H*2: *E*5: Evidence of the suspect's qualifications was found. *E*6: The IP address of the seized computer matches the attacker's IP address at the time of the attack. *E*7: DDoS tools were found on the seized computer. *E*8: Evidence of the user creating DDoS tools was found. *E*9: Log entries of searching for DDoS tools on the Internet were found. *E*10: Log entries of downloading DDoS tools from the Internet were found. *E*11: A botnet control program was found. *E*12: Evidence of the user creating the botnet control program was found. *E*13: Log entries of a DDoS attack launched on the target computer through the botnet were found. *E*14: Log entries of connecting to the botnet were found. *E*15: The IP address of the seized computer matches the botnet control IP address at the time of the attack.

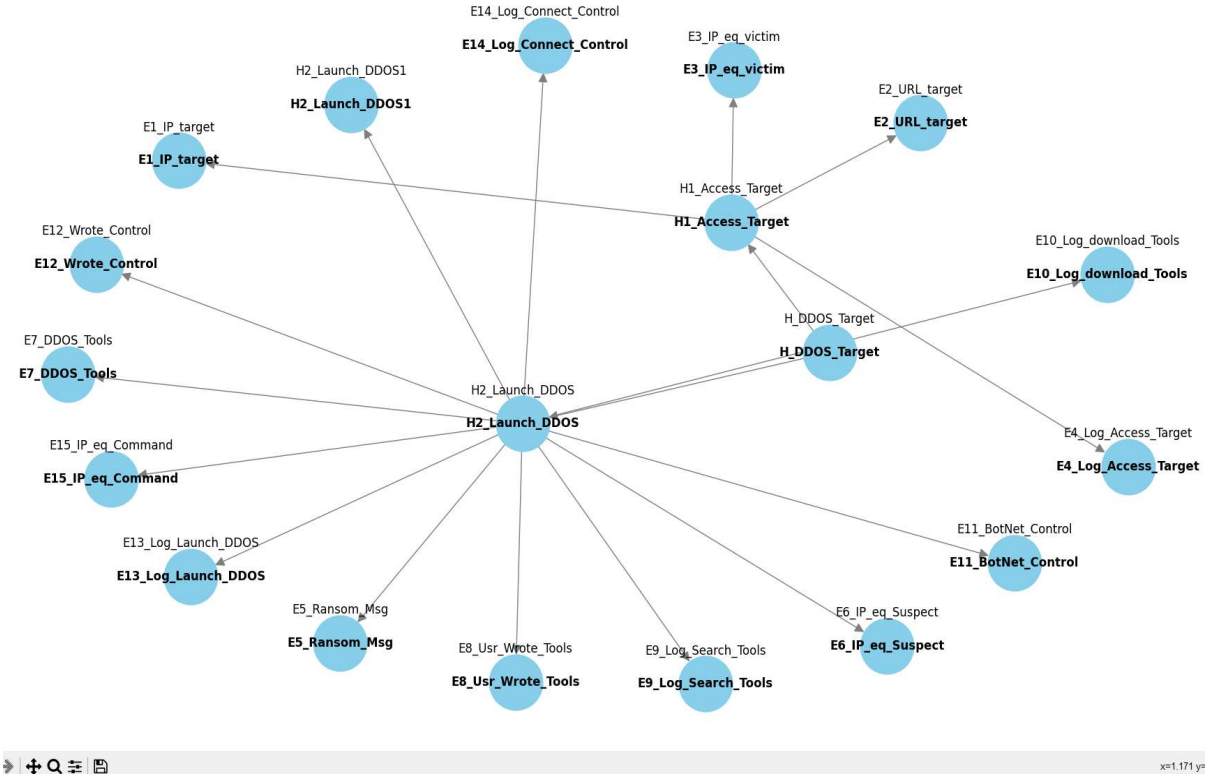

Figure 1: Structure of a Bayesian network, visualizing the main hypothesis, sub-hypotheses, and corresponding evidence

Fig. 2 shows a general view of our software product with a results output block displaying the probabilistic assessments of the collected evidence to support various hypotheses (Main hypothesis—the seized computer (CSN) was used to launch a DDoS attack on the target computer, along with two sub-hypotheses described earlier). In addition to this output format, the obtained results can be visualized more clearly in the form of histograms, as shown in Fig. 3.

This format of visualizing conclusions in the form of histograms, obtained for the probabilities of various evidence during the investigation of DDoS attacks from the suspect's computer, makes the process of analyzing evidence more convenient and easier to interpret.

Automation largely eliminates subjective errors and bias that can occur during manual analysis of evidence. The use of a Bayesian network (BN) allows for more precise consideration of the probabilities of various events and their interrelations, which often leads to objective conclusions. It is important to note that automated systems, such as the one proposed in this work, significantly accelerate the process of analyzing large volumes of data.

This is especially important in time-constrained environments during cybercrime investigations, as the use of Bayesian networks allows for the effective representation of complex dependencies between various pieces of evidence and hypotheses. Additionally, automation enables the use of advanced algorithms and analytical methods that may not be available during manual data processing. This leads to higher-quality and deeper evidence analysis, increasing the chances of successfully investigating crimes

related to the implementation of DDoS attacks on ICS (industrial control systems).

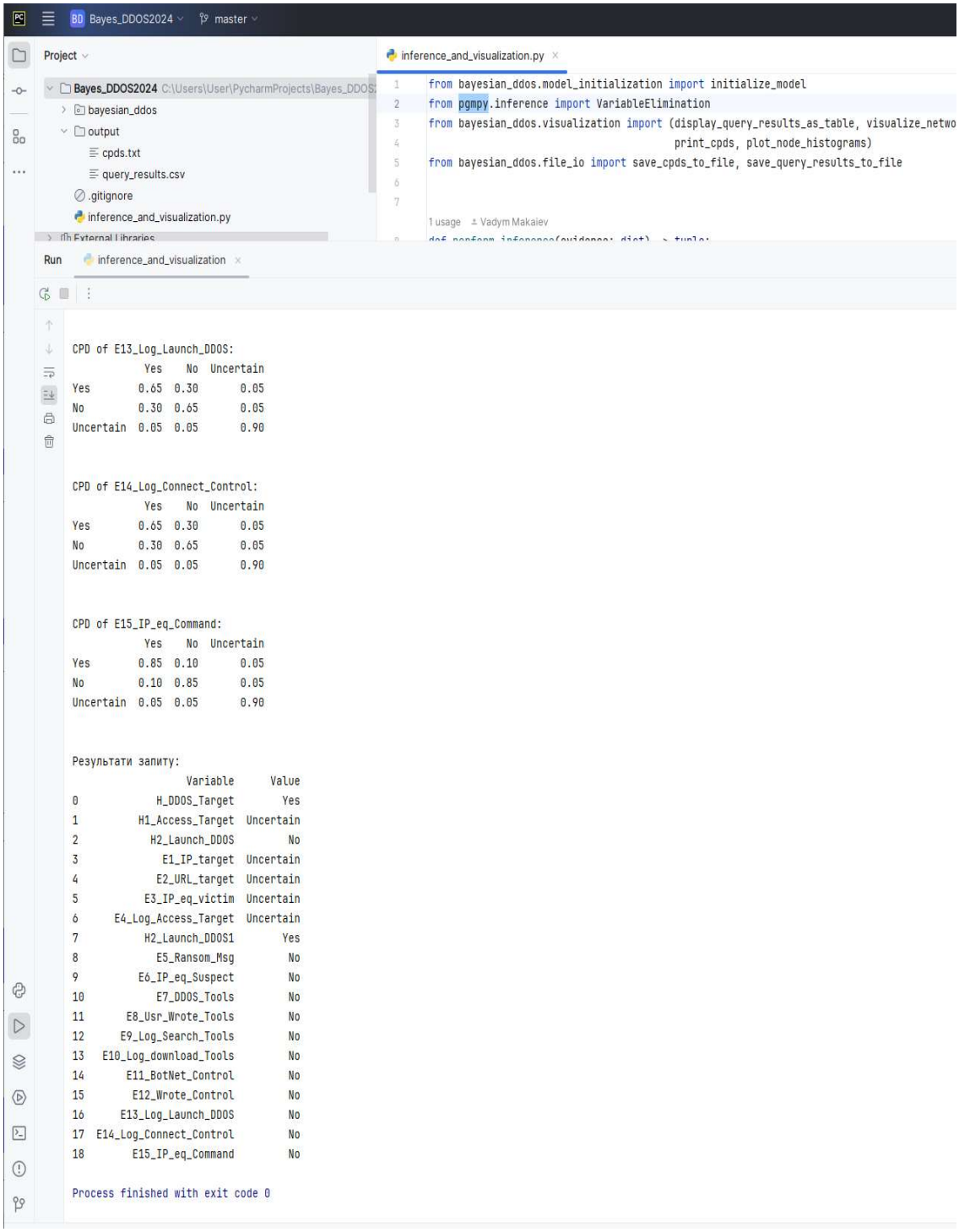

Figure 2: General view of the conclusions obtained during the calculation of probabilistic assessments of the collected evidence to support various hypotheses

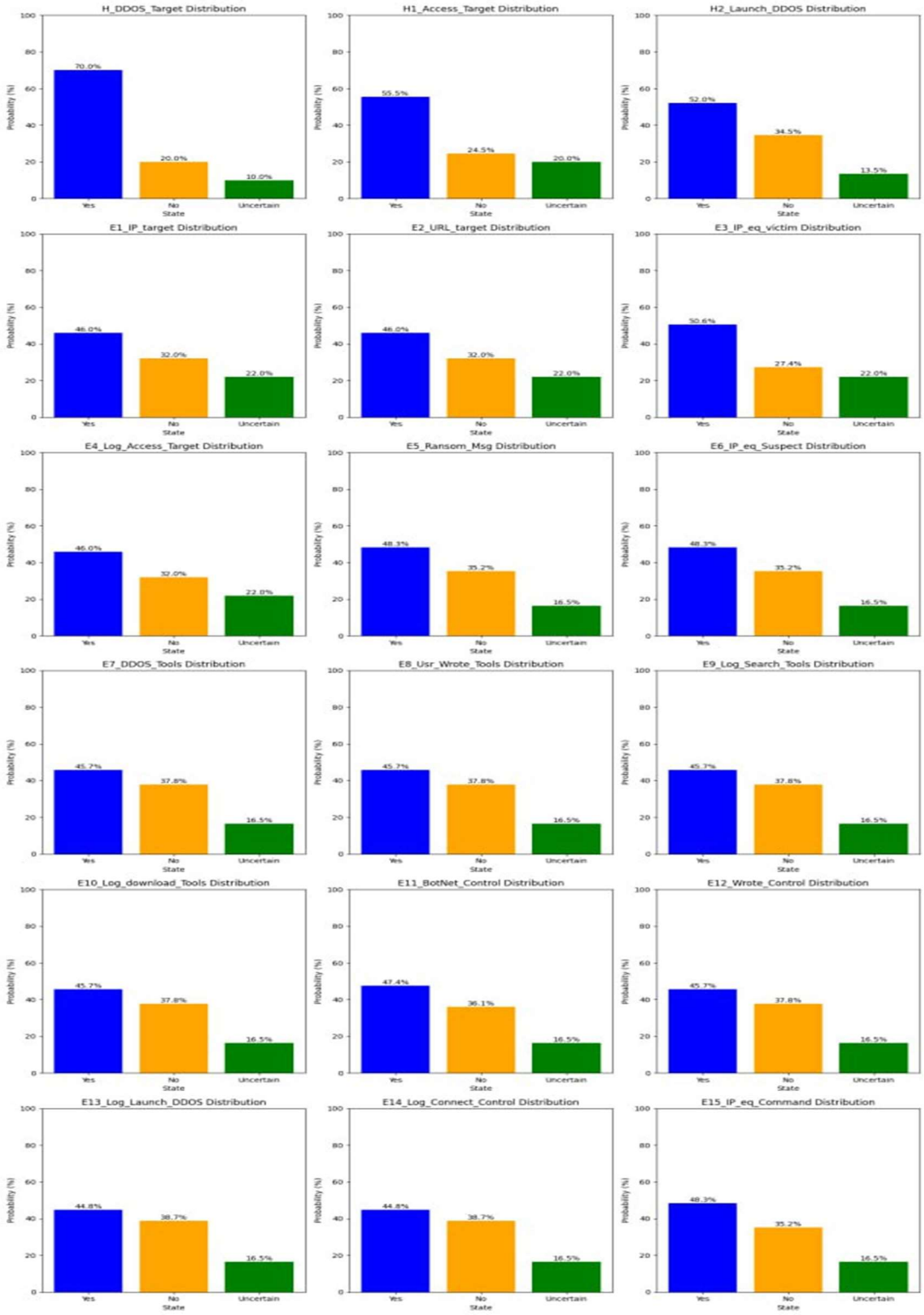

Figure 3: Visualization of conclusions in the form of histograms, obtained for the probabilities of various pieces of evidence during the investigation of DDoS attacks from the suspect's computer

The development of a software product in Python using Bayesian networks, in our view, ensures the standardization of analysis methods. This allows practicing specialists in the field of cybercrime investigations to apply a unified approach to various investigations, simplifying the training and preparation of specialists and ensuring consistency in methods and approaches. Automated systems, similar to the one presented above, provide quantitative probabilistic assessments that assist investigators and experts in making well-informed decisions. In particular, modeling various scenarios and their probabilistic evaluations enables more accurate forecasting of outcomes and the development of strategies for investigating such crimes in the future.

Finally, automation ensures the transparency of the analysis process, allowing the results to be easily reproduced and verified. This is critically important for the legal validity of conclusions and their presentation in court.

The prospect of further research lies in the addition of dialogue windows for expert interaction to the developed software product. This will significantly enhance the usability of the computational core based on the Bayesian network, which is particularly important for investigating applied cases related to DDoS attacks on industrial control systems (ICS). Expert dialogue windows will provide an intuitive and user-friendly interface, simplifying data entry and system interaction. This is crucial because experts investigating DDoS attacks are often not programming specialists. A simple and clear interface will allow them to effectively use the software product without requiring deep programming knowledge. Moreover, the introduction of dialogue windows will significantly reduce the time needed for data entry and processing. Experts will be able to interact with the system more quickly and efficiently, thereby accelerating the investigation process.

3. Conclusions

In this paper, the following main results were obtained:

It is shown that the use of Bayesian Networks (BN) in the developed Python software product will automate the process of analyzing collected evidence, eliminating subjective errors and bias often arising in the manual processing of data during cybercrime investigations.

It is demonstrated that automating the analysis will significantly reduce the time required to process large amounts of data, which is especially important in time-limited conditions when investigating cybercrimes, particularly DDoS attacks.

It is established that for the task of establishing responsibility for carrying out DDoS attacks, BN allows for accounting for the probabilities of various events and their relationships, leading to more accurate and objective conclusions. This is critically important for the legal justification of conclusions and their presentation in court.

It is demonstrated that developing a Python-based software product ensures the unification of analysis methods, allowing a consistent approach to different investigations, and simplifying the training and preparation of specialists.

It is shown that automation ensures the transparency of the analysis process, allowing for easy reproduction and verification of results, and enhancing trust in conclusions and their legal significance.

The presented approach and the developed software product can be effectively used to model various scenarios and their probabilistic assessments, allowing for more accurate predictions of cybercrime consequences and developing strategies for their investigation in the future. The work demonstrates that the proposed automation of cybercrime analysis using BN is an important step in improving the investigation and decision-making processes, particularly in the context of DDoS attacks on ICS.

References

[1] P. Anakhov, et al., Evaluation Method of the Physical Compatibility of Equipment in a Hybrid Information Transmission Network, Journal of Theoretical and Applied Information Technology 100(22) (2022) 6635–6644.

[2] V. Zhebka, et al., Optimization of Machine Learning Method to Improve the Management Efficiency of Heterogeneous Telecommunication Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 3288 (2022) 149–155.

[3] A. A. Cárdenas, et al., Attacks against process control systems: risk assessment, detection, and response, 6th ACM Symposium on Information, Computer and Communications Security (2011) 355–366.

[4] Z. Jadidi, et al., Automated detection-in-depth in industrial control systems, Int. J. Adv. Manufacturing Technol. 118(7) (2022) 2467–2479.

[5] N. Tuptuk, et al., A systematic review of the state of cyber-security in water systems, Water, 13(1) (2021) 81.

[6] C. Cheh, Protecting critical infrastructure systems using cyber, physical, and socio-technical models, Doctoral dissertation, University of Illinois at Urbana-Champaign (2019).

[7] V. Astapenya, et al., Conflict Model of Radio Engineering Systems under the Threat of Electronic Warfare, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, CPITS, vol. 3654 (2024) 290–300.

[8] H. Tse, K.-P. Chow, M. Kwan, A Generic Bayesian Belief Model for Similar Cyber Crimes, 9th International Conference on Digital Forensics (DF) (2013) 243–255. doi: 10.1007/978-3-642-41148-9_17.

[9] G. Yan, et al., Towards a Bayesian network game framework for evaluating ddos attacks and defense, ACM conference on Computer and communications security (2012) 553–566.

[10] K. L. Hui, S. H. Kim, Q.H. Wang, Marginal deterrence in the enforcement of law: Evidence from distributed denial of service attack (2013).

[11] P. Das, P. Sarkar, The Importance of Digital Forensics in the Admissibility of Digital Evidence, NUJS J. Regul. Stud. 7(60) (2022).

[12] O. Kryvoruchko, et al., Analysis of technical indicators of efficiency and quality of intelligent systems, Journal of Theoretical and Applied Information Technology, 101(24) (2023).

[13] A. Adranova, et al., Methodology forming for the approaches to the cyber security of information systems management, J. Theor. Appl. Inf. Technol. 98(12) (2020) 1993–2005.

[14] H. Hnatiienko, et al., Prioritizing Cybersecurity Measures with Decision Support Methods Using Incomplete Data, in: 21th International Scientific and Practical Conference "Information Technologies and Security", vol. 3241 (2021) 169–180.