Fingerprinting Network Sessions for the Discovery of Cyber Threats

C.D. Klopper[1], J.H.P. Eloff[2]
[1]Masters in IT Big Data Science, University of Pretoria, Pretoria, South Africa
[2]Department Computer Science, University of Pretoria, Pretoria, South Africa

u26305667@tuks.co.za
jan.eloff@up.ac.za

Abstract: Artificial intelligence (AI) assisted cyber-attacks, within the network cybersecurity domain, have evolved to be more successful at every phase of the cyber threat lifecycle. This involves, amongst other tasks, reconnaissance, weaponisation, delivery, exploitation, installation, command & control, and actions. The result has been AI-enhanced attacks, such as DeepLocker, self-learning malware and MalGan, which are highly targeted and undetectable, and automatically exploit vulnerabilities in existing cyber defence systems. Countermeasures would require significant improvements in the efficacy of existing cyber defence systems to enable the discovery and detection of AI-enhanced attacks in networks in general. The challenge is that rule-and-anomaly-based intrusion detection approaches would need to be evolved into a dynamic self-learning approach before being able to discover "undetectable" network threats. The problem is that, when considering current state-of-the-art network cybersecurity countermeasures, this has not yet been achieved. One of the key challenges in achieving this is the inability to extract meaningful information from network packets. The novel solution proposed in this paper is to fingerprint network sessions. Each fingerprint is represented by a two-dimensional matrix that can be visualised, comprising a unique session key, the protocol discourse and the transmitted data. This is achieved by extracting information, summarising network session key events, encoding the received data, and merging it with existing fingerprints. The unique key and transmitted data are encoded using a Hilbert curve, while the protocol discourse is encoded into a tornado diagram. The resulting visualised network session fingerprints reveal hidden patterns that are ideal for subsequent pattern recognition, reinforcement learning (RL) or support vector machines (SVM) training to discover AI-enhanced cyber threats as they evolve.

Keywords: Cybersecurity, Cyber Defence, Network Session Fingerprints, Packet-Based Feature Extraction, Visual Data Mining

1. Introduction

One of the biggest challenges of the 21st century (Sobers 2022) is defending a company's cyber assets from cyber-attacks. According to Sobers (2022), there has been a significant increase in hacking and data breaches since the start of COVID-19. Findings in (Sobers 2022) characterise unprotected data and poor cyber security practices as one of the key challenges in improving cyber defence.

A widely used term that describes the importance of cyber defence is "zero-day attack" (Kaspersky 2022). Zero-day attack refers to a situation where a large organisation or an anti-virus company has discovered a vulnerability for which there are zero days to mitigate. With AI-assisted cyber-attacks within the cybersecurity domain, "zero-day attacks" will be extended to the time when new advancements in new cyber threats are discovered. The most notable example is ransomware. According to Indusface (2022), in 2021 ransomware attacks increased by 92.7%. This is likely due to the increased use of more advanced machine learning (ML) and deep learning (DL) techniques that make new ransomware malware more undetectable (Du, Raza, Ahmad, Alam, Dar and Habib 2022).

Cybercriminals using AI, BotNets and generative adversarial networks (GAN) have advanced their ability to perform highly sophisticated AI-assisted attacks that have become increasingly more difficult to discover, detect and protect against (Kaloudi and Li 2020) and (Nguyen and Reddi 2019).

The University of New South Wales (UNSW) and the Australian Centre for Cyber Security (Moustafa and Slay 2015) performed a simulation using an IXIA PerfectStorm tool to record data for research in cyber security. UNSW performed intrusion detection during the simulation on the nine types of attacks that were synthesised during the simulation. The intrusion detection and prevention performance are depicted in Figure 1.

From Figure 1, four cyber-attacks were completely undetected (analysis, backdoors, shellcode and worms) including "zero-day attacks". Only a fraction of the attacks were detected, namely denial of service (DoS - 3.7%), exploits (0.6%), fuzzers (4.2%), generic (3.5%) and reconnaissance attacks (0.2%). This result proves the inability of traditional rule and anomaly threat detection as a cyber defence to successfully detect threats that were simulated. State-of-the-art cyber defence systems should be capable of discovering and detecting 99% of threats for all categories to tolerate automated responses.

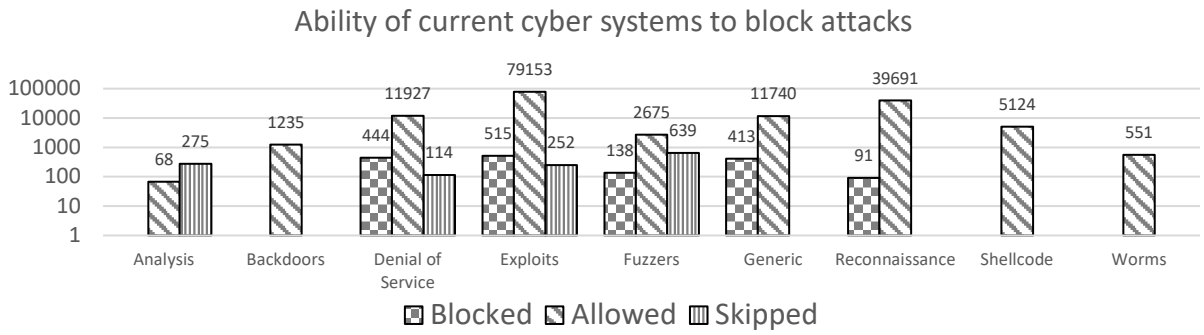

Figure 1: UNSW cyber simulation results

The simulation performed by UNSW was completed in 2015; since then, the cyber threat landscape has evolved with AI-assisted cyber-attacks. In a study conducted by Kaloudi and Li (2020), AI-assisted attacks were identified in nearly every phase of the cyber threat lifecycle.

The cyber threat lifecycle stipulates the general tasks a cyber attacker needs to complete to infiltrate an organisation. The key tasks are:

- During the reconnaissance task, a cyber attacker collects information and intelligence to aid the planned cyber-attack.
- During the weaponization task, the collected information is used to improve the effectiveness of the cyber-attack.
- During the delivery task, the cyber-attack needs to successfully bypass the boundary safeguards such as intrusion detection defence systems.
- During the exploitation task, the cyber-attack attempts to infiltrate the target organisation's network.
- During the installation task, the cyber-attack obtains a secure connection to open the organisation’s network for malicious attacks.
- During the command & control task, the cyber-attacker has remote control of the organisation’s network.
- During the actions task, the cyber-attacker executes the intended malicious activity.

In Kaloudi and Li (2020), the following key cyber-threats were identified that demonstrated the highest levels of AI-assistance (Refer to Kaloudi and Li (2020) for a complete list of threats across the cyber threat lifecycle):

- Smart Malware is a self-learning malware that can deploy malicious attacks, while it seems like accidental failures by manipulating control systems. Smart Malware represents advancements in AI-targeted reconnaissance.
- MalGAN is an algorithm based on a GAN that generates adversarial malware to bypass black-box malware detection systems. MalGAN can, upon detection, generate new malware examples to remain undetectable. DeepDGA is a next-generation approach to bypass domain name detection systems based on deep neural networks (DNN). Both MalGan and DeepDGA represent advancements in AI-concealed delivery.
- DeepLocker is an extremely targeted evasive malware and only deploys once it confirms its target. No defences have been implemented against DeepLocker. DeepLocker represents advances in AI-multilayered C2.

It is concluded in (Kaloudi and Li 2020) and (Nguyen and Reddi 2019) that AI-enhanced attacks have evolved to highly targeted, undetectable malware that automatically exploits vulnerabilities in existing cyber defence systems.

Therefore, with cyber threats evolving with AI-assistance across every phase of the cyber threat lifecycle (Kaloudi and Li 2020), cyber defence systems require significant improvements to improve efficacy in networks in general. The challenge is that rule and anomaly based intrusion detection approaches would need to be evolved into a dynamic self-learning approach (Kaloudi and Li 2020) which is capable of reacting to dynamic changing threats in real time.

According to Kaloudi and Li (2020), a cyber defence system capable of defending against AI-assisted cyber threats needs to incorporate RL methods. This is because a system based on RL methods can decipher complex and dynamic changes in cyber threats by learning from its experiences, and then respond to the threat in real-time.

The problem is that when considering the current state-of-the-art network cybersecurity countermeasures, a dynamic RL-based cyber defence has not yet been achieved. Caminero, Lopez-Martin, and Carro (2019) developed an adversarial environment (AE) RL algorithm for intrusion detection. In Caminero et al. (2019) and Lopez-Martin, Carro and Sanchez-Esguevillas (2020), the input source was pre-determined network features, which are not applicable to a real-time environment. The novel AE-RL model proposed by Caminero et al. (2019) scored a precision of 81% (DoS), 49% (Probe), 79% (R2L) and 9% (U2R), which is not suitable for an automated response. Lopez-Martin et al. (2020) expanded to additional RL models, including deep q network (DQN), deep dueling q network (DDQN), policy gradient and actor critic techniques. The DDQN scored a precision of 92% (flooding), 0% (impersonation), 93% (injection) and 96% (benign), which is not suitable for an automated response. Malialis (2014) developed a distributed RL for network intrusion response in real-time. In Malialis (2014), the input source was real-time network packets. The proposal is to use a distributed RL defence system to throttle distributed denial of service (DDoS) attacks. The group score (considering various input sources) for % of legitimate traffic unaffected was 72%.

One of the key challenges in achieving a dynamic RL-based cyber defence is the inability to extract meaningful information from raw network packets, which is crucial for an RL-based cyber defence to self-learn. This is because, once in operation, a dynamic RL self-learning cyber defence will need to detect threats in real time from network traffic. To overcome this challenge, the novel solution proposed in this paper is to fingerprint network sessions, comprised of the relevant raw network packets. Raw network packets contain a lot of information and vary in size. A fingerprint will overcome this by extracting only the meaningful information, while standardising the size of the input into the RL model. This allows the RL cyber defence system to learn from raw network packets, without being encumbered by irrelevant information. Each fingerprint is represented by a two-dimensional visualisation and comprises a unique session key that is available in network feature datasets, and the protocol discourse and transmitted data which can be extracted only from raw network packets.

Therefore, this is a novel topic for research, since no previous research considered RL techniques using raw network packets, nor using raw network packets to determine and encode the protocol discourse and transmitted data as proposed in this paper. The main research question for the research project at hand is as follows: “Can a fingerprint be developed that captures the hidden patterns contained in network sessions, which were previously undetectable, and can it be visually encoded?”

2. Relevant and Background Work

There are two different areas of research that contribute to the research at hand, namely:

1. Research that focuses on extracting meaningful features from network traces; and
2. Research that focuses on visual data mining and encoding data into visual information.

The relevant research for each field of research will be discussed in more detail in the remaining sections.

2.1 Extracting features from network traces in the form of PCAP files

PCAP files are generated from performing network traces. The contents of the PCAP are all the network files that were captured during a network trace, which include source and destination IP addresses, ports, protocol data and the actual data to be transmitted.

Several papers have developed algorithms that determine and process features for training (ML) directly from PCAP files. To evaluate these approaches, each will be considered in the method by which they aggregated the network traffic (Zhou, Niu, Zhang, Peng, Wu, and Hu 2020). in Table 1.

Table 1: Summary of relevant research in extracting features from network traces

Aggregation	Description	Relevant research	Performance
▪ Protocol level	▪ All network packets transmitted are considered per protocol	▪ Feature Map (Sun, Ochiai, and Esaki 2020) – uses a Hilbert curve to develop an activity map for 9 protocols for every 128 seconds and federated learning	▪ The highest accuracy scores were 92% and 87%.

Aggregation	Description	Relevant research	Performance
▪ Connection level	▪ All individual network traffic transmitted are considered per packet	▪ Packet2Vec (Goodman, Zimmerman, and Hudson 2020) – uses a modified embedding technique (Word2Vec) to vectorise network packets for automated feature extraction	▪ 0.46% malware samples: ▪ 63% precision, 93% recall, 75.1% F1
▪ Network flow level	▪ All network packets transmitted are considered per: ▪ two hosts, with specific IP addresses, ports, and protocol ▪ for each flow separately (there could be multiple flows between two hosts, IPs and ports that represent distinct communication events)	▪ Graph model (Nari and Ghorbani 2013) – uses dependencies between flows and flow information to develop behavioural profiles	▪ Existing anti-virus more superior
		▪ PcapNg (Velea, Ciobanu, Gurzau, and Patriciu 2017) – is a proposal to extract and store key network flow information (not applied to intrusion detection)	▪ Untested approach
		▪ Wavelet analysis (Du, Ma, Li, Li, Sun, and Liu 2018) – transforms the network flow into Daubechies1, Coifiets1 and Discrete Meyer wave functions, and then extracts wave features for anomaly detection	▪ 89% test precision (unclear amount of malware samples)
▪ Session level	▪ All network packets transmitted are considered per: ▪ two hosts, with specific IP addresses, ports and protocol	▪ DL-IDS (Sun et al. 2020) – uses CNN and LSTM to extract spatial and temporal features (developed a transmitted data image with one-hot-encoding)	▪ Overall F1 score of 93.32%, with a low sample % of malware, and threats separated in time
▪ Host level	▪ All network packets transmitted between hosts are considered	▪ Host and Network (Zhou et al. 2020) – uses statistical and behavioural features, SVM, XGBOOST and Random Forest classifiers, with a confusion classier to detect intrusions	▪ Best performance achieved is 95.2% accuracy (unclear amount of malware samples)

In conclusion, there have been many different approaches (Graph Models, Packet2Vec, Wavelet) using many different aggregating methods (flow, session, connection, host level). The feature map (Sun, Ochiai, and Esaki 2020) and host & network (Zhou et al. 2020) approaches measure performance in accuracy, which, given the fact that most datasets have small amounts of malware samples, is not meaningful. Performance is better measured using precision or an F1 score. The best performing approach is DL-IDS (Sun, Liu, Li, Liu, Lu, Hao, and Chen, 2020), which achieved an overall F1 score of 93.32%. In achieving a dynamic self-learning RL cyber defence with an automated response, higher levels of F1 scores are required. From Sun et al. (2020), there does, however, seem promise in using session aggregation and including an image of the transmitted data within the fingerprint. In addition, there is no implementation of RL learners directly from PCAPs evidenced in Table 1.

Therefore, for this research, the fingerprint will be developed using the session aggregation approach and include a visualisation of the transmitted data, building on the successes achieved by DL-IDS.

2.2 Research that focuses on visual data mining and encoding data into visual information

Visual data mining techniques have become the key adaptation that is applied to data to develop representations for inputs into AI or machine learning algorithms (Wu, Wang, Shu, Moritz, Cui, Zhang, Zhang, and Qu 2021). Visual data mining is already widely used in transformation, assessment, comparison, querying, recommendation, mining and reasoning applications (Wu et al. 2021).

Regarding large datasets, one visual data mining technique, namely pixel-orientated visualisation, has developed a formal representation for encoding data using space-filling techniques (Keim 1996). The primary principle of a space-filling curve is to provide a continuous curve that passes through every point of a spatial region.

One problem, according to Keim (1996), is finding a meaningful arrangement for mapping the pixels because there are many possibilities for arranging the data. Techniques that provide clustering of closely related data are more meaningful and allow for discovering patterns that would otherwise be difficult (Keim 1996). This is vital for the research in this paper, which is aimed at maximising the discovery potential of patterns within network traffic.

Keim (1996) have identified four possible arrangements, line-by-line, column-by-column, Hilbert, and Morton, which are the most appropriate for data with a natural ordering. While line-by-line and column-by-column would provide a space-filling curve, it would result in points being close together in 1D, not being close together after being mapped to 2D. This is not ideal for this research. Hilbert and Morton, on the other hand, map 1D items so that they remain close in 2D. In addition, network packets contained in the network trace PCAP files are generally stored in the order in which they were received. Therefore, network trace data has a natural ordering, which is meaningful for visual data mining.

According to Keim (1996), both the Hilbert and Morton curves have excellent clustering capabilities, because of the resulting 2D space remaining relatively consistent with the 1D input, as depicted in Figure 2. The clustering capabilities provide the most meaningful representations for discovering hidden patterns.

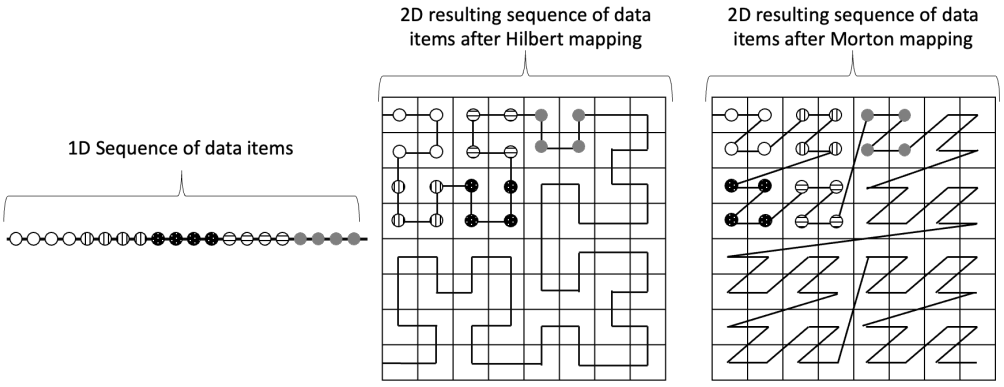

Figure 2: Example 1D mapping to 2D for Hilbert and Morton curves

The Hilbert curve, however, is consistent in that all data items in the 2-dimensional encoding remain close to the data items that were close in the 1-dimensional form, unlike with the Morten curve, as depicted in Figure 2. This property of the Hilbert curve is vital for encoding network session packets.

Therefore, for this research, to ensure that the resulting fingerprint is effective in extracting meaningful information from network packets, the fingerprint will be encoded using the Hilbert curve approach.

3. Fingerprinting Network Sessions

To develop a fingerprint that can be created in real-time as network packets are received, it should be limited to the information to which a firewall has access, since inspecting packets too deeply will add significant delays and will duplicate the functionality of the web application firewall (WAF). At a high level, a firewall has four functions and access to source and destination IP addresses, ports, protocol, associated flags and the transmitted data (Ingham and Forrest 2002), as depicted in Figure 3.

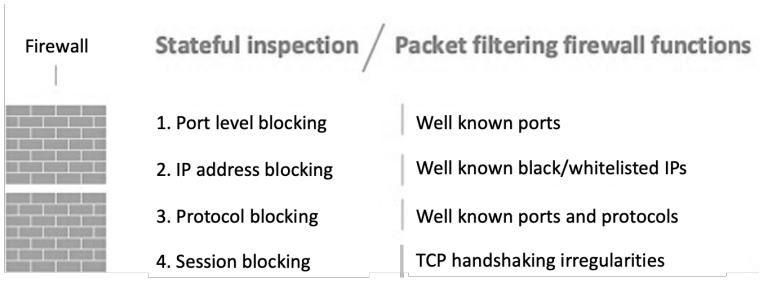

Figure 3: Firewall functions and data access

The methodology to develop a fingerprint based on the information available to a firewall and subsequent cyber threat detection can be structured in the following different steps (steps 4 to 5 are not in the scope of this paper):

1. Collect PCAP data
2. Extract only relevant data from PCAP
3. Formulate a representation to fingerprint unique sessions

4. Train RL to discover and classify cyber threats, with as little information as possible (by rewarding discovery with incomplete fingerprints more) and optimise to maximise performance
5. Detect cyber threats from PCAP test data

3.1 Collect PCAP data

There are several publicly available PCAP data sets; however, a dataset provided by DARPA was selected as the basis for this research. UNSW-NB15 (Moustafa and Slay 2015) made the DARPA data set available in 2015. This data set includes attacks that include, amongst other threats, DoS, worms, backdoors, and fuzzers. These attacks include "zero day attacks". The DARPA dataset contains 100 GB raw network packets (PCAPs). Due to the size of the dataset, and that it contains raw network packets with labels, the DARPA UNSW-NB15 dataset was selected as the optimal dataset for this research. It contains 82 Mil network packets in the training dataset alone. In addition, this dataset contains threat labels for categories of threats rather than actual attacks, which will be more meaningful in training a high-performing threat detection algorithm.

3.2 Extract relevant data from PCAP

The UNSW-NB15 PCAP dataset contains a wide assortment of packets, including IP and address resolution protocol (ARP), higher layer protocols and a wide array of lower layer protocols. Lower layer protocols include TCP, UDP, SCTP, GRE, ICMP, ESP, AH and VRRP. Not all protocols contain source and destination ports, such as all other lower-layer protocols, excluding TCP and UDP. Therefore, as minimum source and destination IP addresses (psrc and pdst in ARP) are collected and where ports are not available, a unique ID to facilitate fingerprinting is selected.

From the research conducted in extracting features from PCAPs, it is clear that there is a lot of meaningful data hidden within the sequence of events. Therefore, in addition to the unique identification for each fingerprint, the protocol discourse and transmitted data are also collected. The protocol discourse is the collection of packets that are sent by each party, represented by the length of each packet, and for TCP packets, including their associated flags. For all other packet types, a default flag is set. The transmitted data is the collective payload in bytes of each packet within a unique session.

Therefore, IP source, IP destination, IP length, TCP flags, source port, destination port, protocol, ARP p-source, ARP p-destination and the raw data are meaningful features for the purpose of this research.

3.3 Formulate a representation to fingerprint unique sessions

Key limitations to consider for formulation of the fingerprint are the following:

1. A Hilbert curve has an output that is always in squares of 2x2, 4x4, 8x8, 16x16, 32x32, 64x64, etc.
2. Neural network models input and output shape need to remain constant; any change will result in retraining since there will be a mismatch in weights.

With the limitations in consideration, the fingerprint design has three distinct sections that have different encoding approaches. These are a unique header, the protocol discourse and the transmitted data.

Each fingerprint section is discussed in more detail below.

3.3.1 Header section

The header of the fingerprint needs to be unique for every session. This is achieved by using the source and destination IP addresses, ports and protocols as depicted in Figure 4.

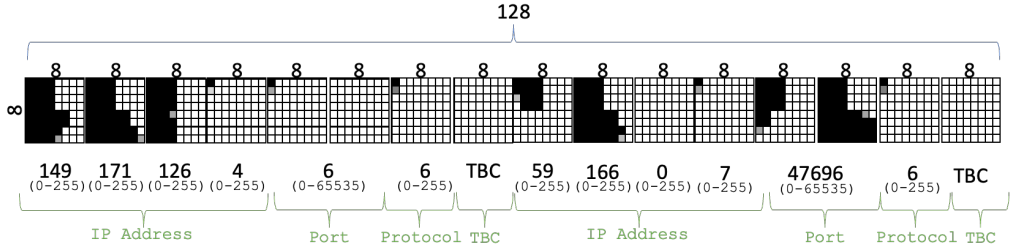

Figure 4: Fingerprint header design

IP version 4 addresses have four sections that range from 0 to 255, each separated by a ".". Using four colours and an 8x8 Hilbert curve, each IP section can be encoded as depicted in Figure 3.

Both TCP and UDP port numbers range from 0 to 65535, which can be encoded using four colours and two 8x8 Hilbert curves. This is achieved by counting 255 in the first Hilbert for every 1 counted in the second Hilbert curve.

Protocols range from 0 to 255 and are encoded in a similar approach as IP sections. The last 8x8 Hilbert curve is reserved for future use and is required to complete the 128 columns required for a 128x128 Hilbert curve, which is used for the transmitted data section. One possible future use considered is to record the frame sizes for IP, TCP, UDP and lower-level protocols in the last 8x8 Hilbert.

3.3.2 Protocol discourse section

The protocol discourse section of the fingerprint needs to portray the exchange of packets between the source and destination hosts for every network session. This is achieved by using the packet length, associated flags and where the packet originates from by using the positive and negative y-axis, as depicted in Figure 5. This visualisation is an adaptation of the typical tornado diagram (Eschenbach 2006).

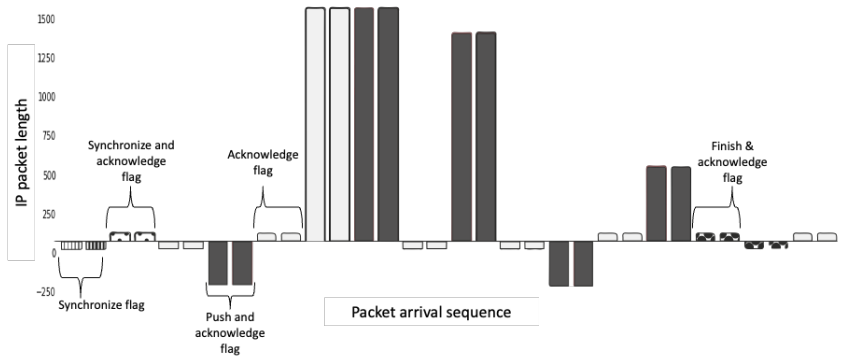

Figure 5: Protocol discourse design

In Figure 5, the protocol exchange is visualised for a TCP session between 59.166.0.7 on port 53421 and 149.171.126.4 on port 80. The exchange is initiated with a request to synchronise, which is acknowledged, after which an initial setup is acknowledged and push and acknowledged before large packets are sent and acknowledged. This is finalised in the end with a finish and acknowledge before closing the exchange.

The fingerprint has enough capacity to capture 128 interactions between two hosts, which can contain multiple flows within the same unique session.

3.3.3 Transmitted data section

The transmitted data section of the fingerprint needs to encode the packet data for all the packets within a unique session, or until the 128x128 Hilbert curve is completed. A fully used Hilbert curve is depicted in Figure 6.

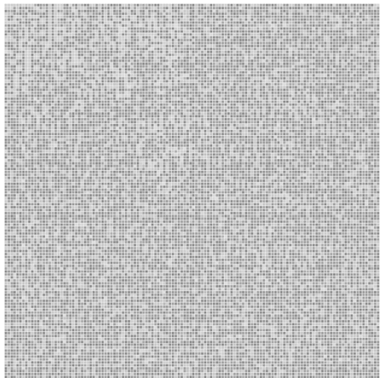

Figure 6: Transmitted data design

Data is transmitted in bytes, which are made up of 8 bits or 0 and 1s. As a result, each byte is converted into decimal range from 0 to 255, which can be encoded into grey scale colours. Therefore, each element of the 128x128 Hilbert curve can depict a byte using grey scale colours.

3.3.4 Proposed dynamic and automated self-learning cyber defence system

Data in the UNSW is extracted in batches for each PCAP separately. Data preparation involves preparing the data for fingerprinting as depicted in Unified Modelling Language diagram below in Figure 7.

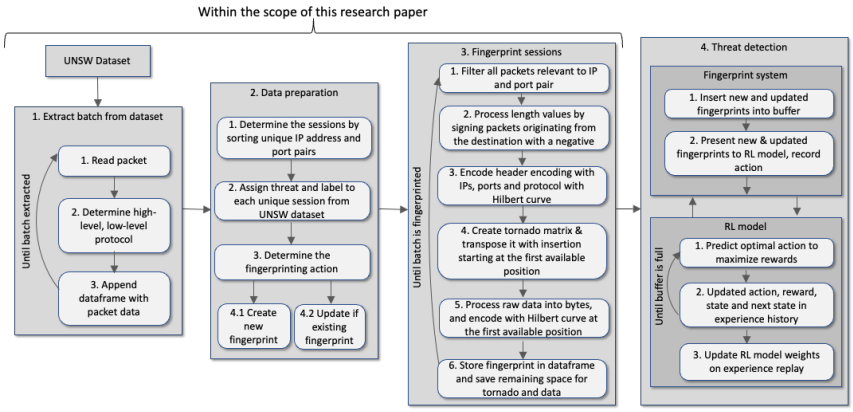

Figure 7: Proposed dynamic and automated self-learning cyber defence system

The fingerprinting system can present fingerprints to the RL model when a unique header is created from the first packet that is exchanged, and then to add new protocol discourse and transmitted data, packet for packet in real-time, or in batches when processing PCAPs.

The RL model predicts the optimal action, which could be assigning a threat or benign classification, or requiring more of the fingerprint to be updated.

The authors of the research at hand plan to train AI and ML models using the fingerprint data as inputs to significantly increase the detection rate of the nine threats within the UNSW dataset, including discovery of "zero-day attacks". The full scope of the proposed self-learning cyber defence system is captured in Figure 7.

4. Results

Using the algorithm and fingerprint system on the UNSW dataset, each unique session can be fingerprinted as benign or malicious. Figure 8 represents a unique fingerprint for a benign network session which contain no malware and Figure 9 represents a unique fingerprint containing malicious shellcode.

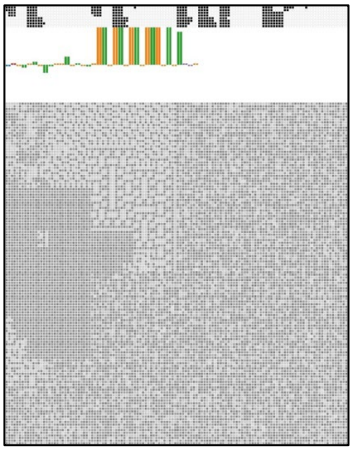

Figure 8: Benign network session fingerprint

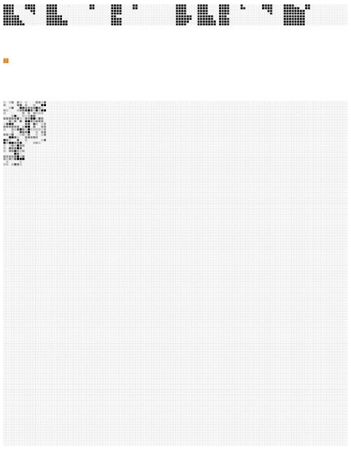

Figure 9: Malicious network session fingerprint

5. Conclusion and future work

The main contribution of this paper is the design of a unique fingerprint by extracting meaningful information from network packets and the fingerprinting system, which is achieved by combining advances in cybersecurity research and in visual data mining. The results in this paper demonstrate that visual ability to discriminate multiple malicious attack types from benign and from one another. Therefore, the results in this paper will lead to more research in RL in the field of cyber security, which will inspire the development of a self-learning dynamic RL cyber defence. Achieving meaningful extraction of information and enabling training of self-learning dynamic RL cyber defence systems, the discovery of undetectable malware, "zero-day attacks" and ransomware, will be possible since fingerprints will significantly simplify the decision boundary for malware detection.

The protocol discourse's unique properties represent the possibility of further study of the possible classification of the application from which the network session was generated. This is significant since there is no approach in place that can accurately classify traffic per application on the open internet accurately for all application in operation today.

Lastly, future work can also be conducted on how to further improve the effectiveness of the fingerprint to contain more information, while improving AI, ML or the effectiveness and performance of RL models.

Author's contributions – The author contributed the entire fingerprint design and subsequent RL model design, including the selection of features for the header, protocol discourse and transmitted data sessions, the encodings as well as the extracting of the relevant features from PCAP files.

Acknowledgements

I would like to thank the following people for their contribution to this research:

- Professor Eloff, for his guidance, expertise, patience, and time.
- My wife and children, for their love and support, that helped me to make this dream a reality.

References

Caminero, G., Lopez-Martin, M. and Carro, B., 2019. Adversarial environment reinforcement learning algorithm for intrusion detection. *Computer Networks, 159*, pp. 96-109.

Du, J., Raza, S.H., Ahmad, M., Alam, I., Dar, S.H. and Habib, M.A., 2022. Digital forensics as advanced ransomware pre-attack detection algorithm for endpoint data protection. *Security and Communication Networks*, *2022*.

Du, Z., Ma, L., Li, H., Li, Q., Sun, G., & Liu, Z., 2018. Network traffic anomaly detection based on wavelet analysis. In *2018 IEEE 16th International Conference on Software Engineering Research, Management and Applications (SERA)* (pp. 94-101). IEEE.

Eschenbach, Ted., 2006. Technical note: constructing tornado diagrams with spreadsheets. *The Engineering Economist, 51*(2), 195-204.

Goodman, E. L., Zimmerman, C., & Hudson, C., 2020. Packet2vec: utilizing word2vec for feature extraction in packet data. *arXiv preprint arXiv:2004.*14477.

Indusface, 15 Malware statistics to take seriously in 2022. Indusface,. https://www.indusface.com/blog/15-malware-statistics-to-take-seriously-in-2022/ Accessed 25 October.

Ingham, K., & Forrest, S., 2002. A history and survey of network firewalls. *University of New Mexico, Tech. Rep*.

Kaloudi, N., & Li, J., 2020. The ai-based cyber threat landscape: a survey. *ACM Computing Surveys (CSUR)*, *53*(1), 1-34.

Kaspersky, What is a Zero-day atack? - Definition and explanation. Kaspersky. https://www.kaspersky.co.za/ resource-center/definitions/zero-day-exploit, Accessed 29 June.

Keim, D.A., 1996. Pixel-oriented database visualizations. *ACM Sigmod Record, 25*(4), pp. 35-39.

Lopez-Martin, M., Carro, B. and Sanchez-Esguevillas, A., 2020. Application of deep reinforcement learning to intrusion detection for supervised problems. *Expert Systems with Applications, 141*, p. 112963.

Malialis, K., 2014. Distributed reinforcement learning for network intrusion response *(Doctoral dissertation, University of York.)*

Moustafa, N., & Slay, J., 2015. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (Milcis)* (pp. 1-6). IEEE.

Nari, S., & Ghorbani, A. A., 2013. Automated malware classification based on network behavior. In *2013 International Conference on Computing, Networking and Communications (ICNC)* (pp. 642-647). IEEE.

Sobers, R., 2022. 89 Must-know data breach statistics [2022]. Varonis, May 2022. https://www.varonis.com/ blog/cybersecurity-statistics, Accessed 29 June. 2022.

Nguyen, T. T., & Reddi, V. J., 2019. Deep reinforcement learning for cyber security. *IEEE Transactions on Neural Networks and Learning Systems*.

Sun, P., Liu, P., Li, Q., Liu, C., Lu, X., Hao, R., & Chen, J., 2020. DL-IDS: extracting features using CNN-LSTM hybrid network for intrusion detection system. *Security and Communication Networks*, *2020.*

Sun, Y., Ochiai, H., & Esaki, H., 2020. Intrusion detection with segmented federated learning for large-scale multiple lans. In *2020 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

Velea, R., Ciobanu, C., Gurzau, F., & Patriciu, V. V., 2017. Feature extraction and visualization for network PCAPNG traces. In *2017 21st International Conference on Control Systems and Computer Science (CSCS)* (pp. 311-316). IEEE.

Wu, A., Wang, Y., Shu, X., Moritz, D., Cui, W., Zhang, H., Zhang, D. and Qu, H., 2021. Ai4vis: survey on artificial intelligence approaches for data visualisation. *IEEE Transactions on Visualization and Computer Graphics.*

Zhou, J., Niu, W., Zhang, X., Peng, Y., Wu, H., & Hu, T., 2020. Android Malware Classification approach based on host-level encrypted traffic shaping. In *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)* (pp. 246-249). IEEE.