



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

Module Code: ACFA7004

Module Title: Research Methods and Professional Development

**Assignment Title: Evaluating Large Language Models for
Encrypted C2 Malware Traffic Detection and Adversarial
Robustness**

Student Name: Md Arafat Rahman

Student ID: 2437469

Module Tutor: Dr. Suresh Shanmugasundaram

Word Count: 4167

Table of Contents

1. Introduction.....	3
1.1 Context and Background.....	3
1.2 Research Problem.....	4
1.3 Research Aim and Objectives	4
1.4 Significance of the Research	6
2. Literature Review.....	6
3. Research Design and Methodology	8
3.1 Research Design.....	8
3.2 Research Methods	9
3.3 Research Materials	10
4. Expected Outcome	10
5. Timeline	11
References.....	13

1. Introduction

1.1 Context and Background

You are here Cyber security is one of the most prominent fields on computing research, since as organisations become more dependent on networked infrastructure that never free from attack by greed driven criminals, ransomware gangs and state-sponsored actors. Every single step of a contemporary breach is subject to the requirement that the attacker's malware must connect out to some kind of remote command-and-control (C2) infrastructure to obtain guidance, retrieve payloads and exfiltrate data [1]; therefore analysis of network traffic is in the front line of defence.

The widespread implementation of TLS has been exploited by attackers. As WatchGuard pointed out in its Q1 2025 report, "71 percent of malware traverses encrypted inbound and outbound connections"[2], while less specific examinations suggest that approximately half of all malware encoded with SSL/TLS [3]. This "encryption dilemma" means that the same technologies designed to protect sensitive information simultaneously conceal C2 beacons, downloader traffic and exfiltration channels; no longer visible, the payload renders signature-based, deep-packet-inspection NIDS relatively impotent [4].

Three parsers define the state of the art. Machine-learning-based encrypted traffic analysis: These methods classify flows based on statistical side-channel features (e.g., packet size, timing and burst structure) rather than payload content [4]. Second, ET-BERT performs substantially better than hand-crafted-feature approaches via self-supervised representation learning on raw datagram sequences at the byte-level [5]. The third and, in this instance, most relevant area where LLMs have been adapted pertains to traffic: Traffic LLM [6] and MET-LLM [7], employ the inherent generalisation and instruction following capability of these models to classify malicious & encrypted flows bot DoS attacks (the primary deep-learning NIDS issue of limited cross-dataset generalisation) across novel attack types [8].

On the other hand, a much less mature body of literature has eso-fft bet that these same LLMs based classifiers are also susceptible to adversarial manipulation just like the deep-learning ones before them if an attacker knows how he or she can directly control flow characteristics, they could generate perturbations of legitimate traffic which avoid detection while preserving the malign functional behaviour called malware [9]. The project begins by taking a sobering look at the full ML cyber security landscape, narrow over one sub subfield encrypted traffic classification and then in equivocally concentrates on one very specific but, to our knowledge, never critically isolated intersection: LLM-based detectors reap massive generalisation while also facing an obstacle of how attack-resistant these detectors actually are under realistic threat models as defined in the C2 malware problem space.

1.2 Research Problem

Although C2 evasion detection techniques are based on LLMs and recent work [6], [7] and malicious traffic [6], to the best of our knowledge, nearly none evaluate how an Adversary will hone in fields about which they can flexibly change the attributes of their network connections to obfuscate them from this kind of model. The Challenge: Many evaluations of LLM-based encrypted C2 detectors are conducted on static and non-adversarial test data; hence accuracy is often reported without attesting performance against adaptive adversary who may modify specific features (timing, size, or burst) of malicious flows. These tools need to be characterised better in regard to their behaviour under adversarial pressure and organisations may therefore blindly trust them.

A number of studies show this gap, yet a critical reading of each reveal's critical weaknesses. Qing et al. demonstrate that encrypted malicious-traffic detectors trained on standard datasets see significant degradation once the quality of the training-data shifts or a data-distribution shift occurs[10], but their evaluation is not applied to LLM-based architectures, so their finding we cannot trivially assume transfers. In a more general sense, it has been shown that deep-learning classifiers are vulnerable to targeted perturbation of flow statistics [9], but this line of adversarial assessment have not been consistently conducted on attention-based LLM classifiers [8], [9], nor extended to unlabelled traffic-generation settings [11]. However, the work of Li et al. is a partial exception, proposing adversarial examples for LLM-based classifiers on general intrusion-detection benchmark datasets, and one of only several works to consider attacks against this emerging class of model and a serious limitation being using general-purpose NIDS benchmarks rather than encrypted C2 traffic specifically [12]. Zhan, Zhou and Haddadi also confirm the adversarial robustness of NIDS classifiers via similar adversarial attack methods but a further critical limitation is that they treat an LLM as an auxiliary agent working alongside a non-LLM classifier rather than evaluate the LLM directly as the end-to-end network [11].

Despite the wide availability and use of C2 frameworks, such as Cobalt Strike, Mythic and Brute Ratel that are designed to avoid detection with adjustable jitter, padding and traffic shaping [13], an adversary with sufficient funding will be able to slip past this layer of defence if security teams allow this problem to go uncorrected. As wider surveys on LLMs in cyber security agree the emphasis on detection applications have outnumbered adversarial evaluation [14], [15] hence the combination addressed here is as an un-studied inter-connected problem.

1.3 Research Aim and Objectives

Abstract: Purpose: To analyse the performance and robustness of classifiers based on large language models to identify encrypted C2 malware traffic in real-world settings as a network intrusion detection system (NIDS).

Objectives:

Objectives: To provide a systematic review of Encrypted traffic classification, LLM-based network intrusion detection, and the field of adversarial machine learning in cyber security in order to formulate a comprehensive view on state-of-the-art practices as well as validate the found research gap.

Design, design and pipeline a minimum of two LLM-based architectures (an open-weight general-purpose LLM which traffic-domain prompting/fine-tuning is performed for benign versus C2 encrypted traffic flow classification from labelled public datasets), Potentially including, traffic-specialised transformer e.g. such as ET-BERT

We are not simply treating pretrained models as a component in a pipeline: we want to design and experiment with adversarial perturbation methods that adhere to flow attributes an attacker could feasibly manipulate (e.g. packet inter-arrival time, padding, or burst segmentation) while measuring how the accuracy, F1-score, and false-negative rate of each model changes with increasing perturbation budgets.

A task objective to evaluate the detection ability and adversarial robustness for LLM-based classifiers against bias-authored conventional machine learning baselines (e.g. Random Forest or XGBoost), by training both under the same dataset and attack scenarios.

Accordingly, the current study provides empirical guidelines for effectively and safely deploying LLM-based detectors in comparison with encrypted C2 detection and recommendations on how to mitigate any robustness vulnerabilities we observe.

Objective form SMART objectives or All these SMART objectives is aligned with the criteria of smart objective. Specific (this is about an impressive literature review, a model that you build and adversarial validation / robustness benchmark/recommendation); Measurable with outputs to include: a search-and-selection record; implementation of the pipeline; robustness metrics for a given perturbation budget; Written Model Comparison against multiple others.

All of these with an MSc researcher budget are Attainable through: Open-source Datasets (Section 3.3), and couple them with open-weight LLMs that also have API access + university (or cloud) GPU compute and parameter-efficient fine-tuning will be briefly covered in batch-mode for this series. They are Relevant, which you trail with gap > artefact > interpretation pathway; and they are Time-boxed against the built-in timeline for Section 5. The gapless chain: The problem mediates a utility (O2) that without adversarial stress-testing (O3) and a baseline for comparison (O4), on top of the recommendations conducting the end-goal in O5, would not be plausible at all.

1.4 Significance of the Research

This research is of interest to many groups. Instead of vendor-reported, non-adversarial accuracy SOC analysts and blue teams will receive evidence-based information on whether or not LLM-based detectors can be reliably trusted in operations against an adaptive adversary. A reusable adversarial-robustness evaluation methodology for both vendor and researchers. Enabling stand-offs at the boundaries of their native language trying to provide novel perspectives for researchers in the overlapping domains of network security and adversarial machine learning. The practical outputs include a robust-testing pipeline with documentation, adversarially perturbed C2 samples in the hope that they will be useful for future benchmarking and knowledge of which architectural options provide the most resistance to evasion.

2. Literature Review

Introduction WE critically analyze the literature to find patterns, contributions and gaps in current research in the following three fields: encrypted traffic classification, LLM-based network intrusion detection systems and adversarial ML in cyber security which guides the present research. **Literature** The literature considers peer-reviewed articles and preprints and primarily from 2021 to 2026, but includes important preprint papers when needed (particularly, in terms of transformers architecture). **Search strategy.** We conducted our search using IEEE Xplore, ACM Digital Library, ScienceDirect and arXiv (cs.CR/cs. And we were able to conduct it thanks to gathering information up to October 2023 And that becomes possible due to variety of sources both direct and indirect (Google Scholar, Nestations: LG), plus industry reports (WatchGuard, Netscope, StationX). Our search strings are built upon groups of core concepts with the use of Boolean operators (e.g. ("large language model" OR "LLM") AND ("encrypted traffic" OR "network intrusion detection") AND ("adversarial" OR "robustness" OR "evasion")) and command and control keywords (e.g. ("command and control" OR "C2" OR "beaconing") AND ("machine learning" OR "deep learning")).

The time range considered for the literature search is 2021-2026 although seminal studies before this period have been included as they form the background literature. The inclusion criteria mentioned peer-reviewed publications and highly reputable conference proceedings or preprints on target themes; industry reports were considered for quantitative data. No technical discussion was present in the abstract and there were no duplicates. About 45 candidate studies were found out of which 25 were considered for in depth synthesis.

In place of the natural chronological organization, we consider the thematic arrangement of the literature as these span multiple, mostly parallel, methodologies (statistical traffic classification; representation learning using transformer models; application of large language models to security problems; testing adversarial robustness).

SYNOPSIS of theme 1: Encrypted traffic classification without decryption. Shen et al. Survey of machine learning based encrypted traffic analysis approaches and the development of their methodology from payload analysis to statistical properties, including sequence of packet sizes and timings with TLS handshake

metadata [4]. Benchmark datasets with standardized, labeled attack traffic such as CIC-IDS2017 have formed the backbone of many papers that have followed [16], Sharafaldin, Lashkari and Ghorbani. One acknowledged limitation is poor generalisation across datasets: Models learned in one environment usually do not transfer to another [4], [8].

Theme 2: Traffic-Aware Pretrained-Transformer Representations. Taking a similar turn, ET-BERT re-conceptualised encrypted-traffic classification as language-modelling with self-supervised learning on byte-wise "datagram" tokens -achieving huge 5-task result F1 gains over hand-crafted-feature methods [5]. This framework is the basis for all later LLM-for-traffic literature treating traffic bytes as natural-language tokens; however, a major limitation even of this central paper is that it was not trained nor evaluated on adversarially perturbed traffic, meaning there can be no adversarial robustness evaluation.

Lastly, Highlight 3a: The Role of LLMs in Cyber Security more Broadly Systematic Reviews by Hasanov et al. and Zhang et al. So, the use of these systems has been increasing for survey studies addressing phishing detection, log analysis, vulnerability discovery and threat intelligence, but only a few in traffic analysis [14], [15]. Traffic analysis – In their article, Liu, Xie, Zhang and Cui define traffic analysis as the novel use of LLMs but emphasize the existence of the so-called "modality gap" between natural language and structured network data [8]. Ferrag et al. and Seyyar, Yavuz and Ünver explain light-weight BERT-based threat-detection models and show their effectiveness for transformers [17], [18].

Traffic LLM for Traffic Domain similarly provides tokenisation for traffic-domain open-source LLMs, benchmarking on eight datasets to provide a benchmark of malware, botnet and encrypted VPN/Tor detection showcasing generalisability to unseen scenarios [6]. MET-LLM: A dynamic tuning Adaptor integrated with domain-specific tokenization and retraining cost reduction [7]. Guastella et al. Qing et al. Detecting DDoS attacks using deep learning based LLMs a try of robustness on corrupted between shifted managing data[19], [10]. The LLM-based detection it is getting quite mature, but nearly all the reported results are from evaluations over clean non-adversarial environments.

5th Theme: Command-and-control and beaconing detection The content examination based workload on C2 detection has inspired from prominent work from industry and academia relying on statistical beacon-interval analysis [19] and TLS fingerprinting [20], rather than content examination. In a systematic review APT beaconing detection, it was found that behaviour- and network-based methods, including machine-learning approaches now far out together for beaconing detection compared to solely signature-based detectors; adversaries tool increasingly randomises timing to circumvent these detectors [1]. This theme sets the stage for both how operationally relevant it is to detect C2, and what an arms race any LLM-based detector will enter.

Theme 6: Intrusion classifiers' robustness to adversarial attacks. Adversarial attacks on deep-learning NIDS have been widely studied and found, as any NIDS: our first seed paper is dedicated to critical examination of the vulnerability of traffic classifiers to adversarial perturbations [9], there are also papers devoted to critical investigation of the vulnerability of

intrusion classifiers to adversarial attacks, which include approaches of defending from such attacks using trained opponents or ensembles[21], [22]. The first study to impose bounds on attacker-controlled, realistic perturbations to flow features while attacking LLM-based classifiers and discovering “substantial but dataset- and comparator-dependent” vulnerability in seven classifier types is that of Li [12]. Zhan Zhou Haddadi [11] strengthen the robustness of the GNN-based NIDS with the help of LLM agents without testing the LLM itself as the classifier and go on to explore how different the performance of the LLM is with regard to task granularity [23].

Literature underpinned by frameworks. Most studies applying LLM to traffic classification build upon the classical transformer architecture and self-attention mechanism the pretraining via masked language modeling objective as introduced by BERT [25], and the capability of large-scale generative models to learn in context [26]. The adversarial literature is built from gradient-based and black-box adversarial examples (such as the PGD and NES-type queries attacks used in image classifiers) [9] and generalized to network-flow data [12].

Agreement, disagreement, and synthesis. There is consensus that LLM-based techniques are superior to hand-crafted features when applied to clean and in-distribution data [5], [6], [7]. These works may disagree in the robustness context because whereas one finds that LLMs generalize better to novel attacks than traditional machine learning techniques [6], [8], the field of adversarial robustness reports that this generalization creates model-dependent vulnerabilities that do not occur in classical approaches [12], [23]. The reason for this discrepancy is perhaps the difference in experimental conditions, with high-performance works being assessed on static data, and only few adversarial experiments having been conducted on general-purpose IDS benchmarks instead of encrypted C2 traffic. In this paper, we provide a crucial critique of this claim, by evaluating the performance of the same detectors on anonymized C2/beaconing traffic under clean and adversarial setups. The contribution of this work lies in the combination of three factors which is achieved uniquely in none other.

From the literature review presented above, it can be concluded that: (i) the encrypted traffic comprises an increasing percentage of malicious C2 traffic that can reach its destination[2], [3] (ii) transformer-based and large language model (LLM) based classifiers are an emerging technology to detect the malicious C2 traffic without needing decryption [5], [6], [7]; and (iii) adversarial robustness, a property well studied over traditional deep learning intrusion detection systems (IDSs) [8] has recently been considered for LLM-based classifiers but not under the realistic conditions of attackers and not even in the case of encrypted C2 traffic[9], [11], [12].

3. Research Design and Methodology

3.1 Research Design

The adopted ontology for this research is objectivist/realist, meaning that accuracy of detection, false negative rates, and attack success rates are objective measurable and observable features of the system being tested and are not constructs influenced by personal perception. This ontology fits well for a technical experiment in the field of cyber security where the underlying events, whether or not the classifier can identify the traffic, whether an attack is detected, are verifiable objectively with a particular set of data and protocol.

As a result of that, the epistemology of the research is positivist and knowledge is acquired by experimentation and measurements rather than interpretation. Such epistemology is chosen since the purpose of the research implies comparison of objective measurable parameters (accuracy, F1 score, attack success rate), which requires positivism for hypothesis testing[9], [12].

In terms of positionality, the researcher conducts this project from the computing/cyber-security position and with no professional access to production C2 or SOC data, using public, pre-labeled datasets for this work, thus eliminating one potential source of access bias. The remaining bias may be that of confirmation, assuming poor performance of the LLM under adversarial conditions, countered by pre-registration of the evaluation criteria and perturbation budget and reporting of all results, including instances of LLMs exceeding expectations.

The proposed methodology is the combination of the Design Science Research and experimental/simulation research. DSR is applicable in this case because one of the products expected as a result is the evaluation artefact, the adversarial testing pipeline, that is not available at the moment; the experimental part is also relevant due to the need to measure and compare classifiers' performance under certain manipulation. This aligns with the dual character of the objectives of Section 1.3, where Objectives 2 and 3 are addressed through DSR, while Objectives 3 and 4 are addressed through experimental research.

3.2 Research Methods

This is a mostly quantitative study in that its objectives require numerical performance metric comparisons against multiple model types; no qualitative approach needed to address the research question justifies this study as needing more than a quantitative approach for Objectives 2-4.

Immediate research methodology is controlled experimentation and software development: the researcher constructs a classification and adversarial evaluation pipeline (software development), then executes controlled experiments where independent variables (architecture, perturbation budget, and perturbation type) are varied systematically while dependent variables (detection metrics) are observed. Randomized experiments allow causation attribution based on relatively isolated input changes, something that wouldn't be possible through a purely observational or survey-oriented research design.

Proposals Architecture In regard to the architecture there are five steps. There is a step for data preprocessing where CIC FlowMeter is used to convert packet captures into labeled flow-level feature vectors, plus byte sequence representations for the transformer-based models. The Baseline Classifier step trains a traditional ML benchmark (e.g., Random Forest and/or XGBoost) as a non-LLM comparison point. [4], [12]. LLM classification stage: At least two LLM-based systems, namely, an open-weight general-purpose LLM (prompted/fine-tuned) and a traffic-specific transformer such as ET-BERT [5]. Then, the

controllability-aware adversarial perturbation stage, depending on the achievable flow features (timing/jitter, padding, burst segmentation) of the attacker [12], which affects only these features. Finally, a test procedure where all models undergo a test on their clean and perturbed performance (test set) with respect to the perturbation budget.

The usual classification evaluation metrics (accuracy, precision, recall and F1-score of targets) will be applied, which are relevant for binary/misbalanced malicious-vs-benign classification and are consistent with the metrics used in the existing literature [5], [6], [9]. Moreover, the robustness will be quantified as an attack success rate (ASR), which is the ratio of successful adversarial examples in evading detection for a certain perturbation budget and according to the same protocol as in the most relevant work [12]. Paired significance tests (e.g. McNemar's test) will be used to assess the statistical significance of comparisons between SOTA and each of our Boosting runs.

3.3 Research Materials

No new live traffic will be captured; instead, the main data sources will be pre-labeled datasets of network traffic publicly available. Other sources of traffic data include: normal traffic and attacks traffic (CIC-IDS2017/CSE-CIC-IDS2018) [16] and malicious malware specific encrypted traffic dataset like USTC-TFC2016 and CTU-13 that have been used in past studies on LLM-traffic[6]. Peer reviewed datasets will help to establish comparability with the literature review presented in Section 2 and also help to solve any ethical and legal challenges that may arise when capturing live malicious traffic.

Thus, the method of collecting data is reusing the already available publicly available datasets (instead of capturing the live traffic on the network); the features at the flow level will be derived from the PCAPs using CIC Flowmeter and if needed features at the byte level will be derived through the use of Python packet parsers (like Scapy or pyshark/tshark). This is suitable because the research questions are focused on the classifier behavior in laboratory settings (for which the benchmark datasets are available) and generating new live traffic would be too risky.

There are several ethical concerns related to this project. The datasets under consideration have been made publicly available and contain no personally identifiable information. Benign traffic was generated inside testbeds for research purposes and therefore, GDPR processing should not come into question here. Moreover, the suggested approach will not involve live malware and real C2 communication, but only static capture analysis in isolation inside a VM. Another problem is that the output will have a dual-use component – a method of adversarial perturbation can potentially be employed by an attacker. By limiting the output to robustness values alone, and making it clear that these tools are to be regarded as defensive methods in line with responsible-disclosure guidelines. See the form provided on ethics below.

Software Requirements:

Python 3- LLM fine-tuning & inference (Hugging Face Transformers, PyTorch)

Scikit-Learn & XGBoost - to build the baseline classifier

CIC Flowmeter -flow-feature extraction tool

Wireshark/tshark - packet inspection

Hardware Required: GPU-accelerated compute (through-University HPC/GPU resources)

Or when these are unavailable, cloud GPU instances such as those provided by Google Colab Pro. Where full fine-tuning is not feasible, go for parameter-efficient fine-tuning (LoRA), or prompting using APIs. [7].

4. Expected Outcome

This work expects that LLM-based classifiers with rich-scale settings will behave equally well or even better in the case of detecting clean, non-adversarial encrypted C2 traffic when compared to the state-of-the-art baseline in[6], [7]. Introducing adversarial attacks, which are controllability-aware in nature, leads to degradation of the detection performance of those classifiers, thus making an explicit "accuracy robustness gap" evident, confirming initial observations of the effectiveness of LLM-based classifiers in the holistic intrusion-detection setting[12]. Traffic-specialized pre-trained transformer models (as ET-BERT) might be more robust in terms of the OOB test than prompting-based general-purpose LLMs since those are trained on traffic-distribution data.

From the theoretical side of things, the research extends the knowledge of adversarial machine learning to the domain of detection of encrypted C2 communication with LLM-based classifier framework; practical side of things is represented by the fact that those insights can help SOC teams and vendors evaluate whether or not LLM-based detector should be deployed.

Limitations and possible risks will occur. Using publicly available datasets (which is methodologically obligatory, however) but, possibly, creating a bias in the dataset because of labelled datasets such as CIC-IDS2017 and USTC-TFC2016 may not reflect the most up-to-date C2 toolkits behavior. The small amount of LLM architecture examined owing to the limitations on computation and accessing APIs for the duration of the project. This, of course, implies that being an MSc project with only one researcher, the scope of my research will be smaller compared to the bigger one.

5. Timeline

The plan is for this project to span around eighteen weeks starting from project commencement to be completed through seven phases. This would ensure that enough time is allocated for iteration of write-ups and experimentations prior to the submission deadline. We shall identify some of the major elements involved below.

Phase	Weeks	Key Milestone
1. Ethics approval & systematic literature review	Weeks 1–3	Ethics form approved; SLR search-and-selection record and thematic synthesis complete
2. Environment & dataset preparation	Weeks 4–5	Datasets acquired; flow-extraction and preprocessing pipeline operational

3. Baseline ML classifier development	Weeks 6–7	Random Forest/XGBoost baseline trained and validated on clean data
4. LLM-based classifier development	Weeks 8–11	At least two LLM-based classifiers fine-tuned/prompted and validated on clean data
5. Adversarial pipeline & experiments	Weeks 12–14	Controllability-aware perturbation generator built; adversarial evaluation runs completed
6. Analysis & comparative evaluation	Weeks 15–16	Statistical comparison of all models across clean and adversarial conditions complete
7. Write-up, review & submission	Weeks 17–18	Full proposal/report drafted, proofread, referenced, and submitted

Project Timeline (Gantt Chart) — 18-Week Schedule

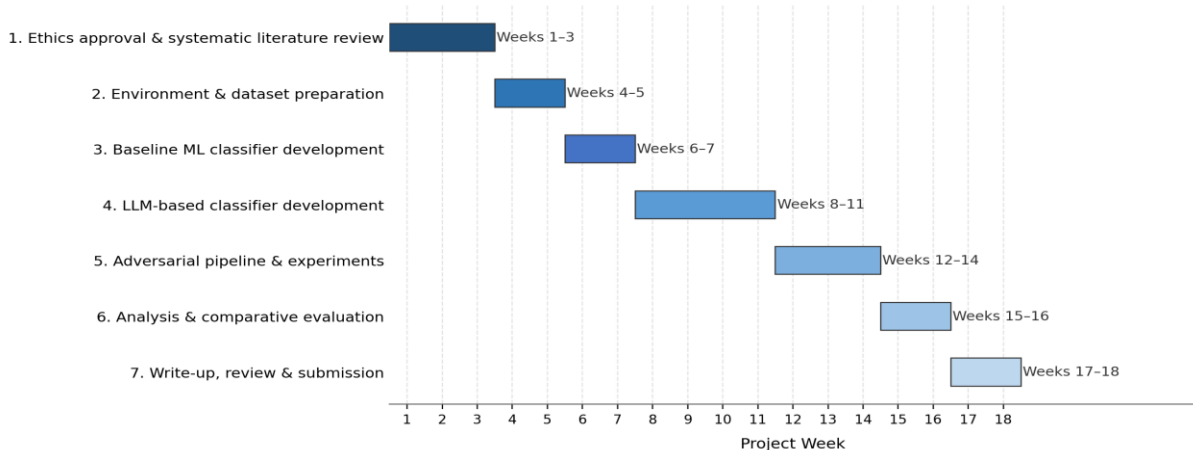


Figure 1: Gantt chart summarising the seven project phases across an 18-week schedule.

The two major risks include, firstly, the dependency on GPU computation to fine-tune the LLMs in Phase 4, and we will mitigate this risk using early contingency by moving to parameter efficient fine tuning or API hosted prompting in case the GPU access is delayed. Secondly, adversarial perturbations do not transfer well to the hosted/parameterless models (API only) in Phase 5, and we will mitigate this risk by using open weight models mainly in our experiments and API-based models for comparison. One more risk includes obtaining ethics approval (Phase 1), and we will mitigate this risk by submitting an Ethics Form as early as possible, most likely in Week 1.

References

- [1] M. Abu Talib, Q. Nasir, A. Bou Nassif, T. Mokhamed, N. Ahmed, and B. Mahfood, "APT beaconing detection: A systematic review," *Computers & Security*, vol. 122, 102875, 2022. Available: <https://doi.org/10.1016/j.cose.2022.102875>
- [2] WatchGuard Technologies, "Internet Security Report Q1 2025," 2025. Available: <https://www.watchguard.com/wgrd-news/blog/watchguard-report-malware-explodes-across-network-and-endpoints-0>
- [3] StationX, "Malware statistics 2026: Facts on threats and trends," StationX, 2026. Available: <https://app.stationx.net/articles/malware-statistics>
- [4] M. Shen, K. Ye, X. Liu, L. Zhu, J. Kang, S. Yu, Q. Li, and K. Xu, "Machine learning-powered encrypted network traffic analysis: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 791–824, 2023. Available: <https://doi.org/10.1109/COMST.2022.3208196>
- [5] X. Lin, G. Xiong, G. Gou, Z. Li, J. Shi, and J. Yu, "ET-BERT: A contextualized datagram representation with pre-training transformers for encrypted traffic classification," in *Proc. ACM Web Conf. (WWW)*, 2022. Available: <https://doi.org/10.1145/3485447.3512217>
- [6] T. Cui, X. Lin, S. Li, M. Chen, Q. Yin, Q. Li, and K. Xu, "TrafficLLM: Enhancing large language models for network traffic analysis with generic traffic representation," *arXiv preprint arXiv:2504.04222*, 2025. Available: <https://arxiv.org/abs/2504.04222>
- [7] Y. Huang et al., "MET-LLM: Enhancing large language models for malicious encrypted traffic detection," *Expert Systems with Applications (in press)*, 2025. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0957417425042368>
- [8] C. Liu, X. Xie, X. Zhang, and Y. Cui, "Large language models for networking: Workflow, advances and challenges," *IEEE Network*, 2024. Available: <https://doi.org/10.1109/MNET.2024.3510936>
- [9] A. M. Sadeghzadeh, S. Shiravi, and R. Jalili, "Adversarial network traffic: Towards evaluating the robustness of deep-learning-based network traffic classification," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1962–1976, 2021. Available: <https://doi.org/10.1109/TNSM.2021.3052888>
- [10] Y. Qing, Q. Yin, X. Deng, Y. Chen, Z. Liu, K. Sun, K. Xu, J. Zhang, and Q. Li, "Low-quality training data only? A robust framework for detecting encrypted malicious network traffic," in *Proc. 31st Network and Distributed System Security Symp. (NDSS)*, 2024. Available: <https://doi.org/10.14722/ndss.2024.23081>
- [11] Z. Zhan, H. Zhou, and H. Haddadi, "Poster: Enhancing GNN robustness for network intrusion detection via agent-based analysis," *arXiv preprint arXiv:2506.20806*, 2025. Available: <https://arxiv.org/abs/2506.20806>
- [12] Z. Li, "Controllability-aware adversarial examples against LLM-based network traffic classifiers," *arXiv preprint arXiv:2607.07739*, 2026. Available: <https://arxiv.org/abs/2607.07739>

- [13] Netskope, "Effective C2 beaconing detection," White paper, Netskope, 2025. Available: <https://www.netskope.com/resources/white-papers/effective-c2-beaconing-detection-white-paper>
- [14] I. Hasanov, S. Virtanen, A. Hakkala, and J. Isoaho, "Application of large language models in cybersecurity: A systematic literature review," IEEE Access, 2024. Available: <https://doi.org/10.1109/ACCESS.2024.3505983>
- [15] J. Zhang, H. Bu, H. Wen, Y. Liu, H. Fei, R. Xi, L. Li, Y. Yang, H. Zhu, and D. Meng, "When LLMs meet cybersecurity: A systematic literature review," Cybersecurity, vol. 8, no. 1, pp. 1–41, 2025. Available: <https://doi.org/10.1186/s42400-025-00361-w>
- [16] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. 4th Int. Conf. Information Systems Security and Privacy (ICISSP), 2018, pp. 108–116. Available: <https://doi.org/10.5220/0006639801080116>
- [17] M. A. Ferrag, M. Ndhlovu, N. Tihanyi, L. C. Cordeiro, M. Debbah, T. Lestable, and N. S. Thandi, "Revolutionizing cyber threat detection with large language models: A privacy-preserving BERT-based lightweight model for IoT/IIoT devices," IEEE Access, 2024. Available: <https://doi.org/10.1109/ACCESS.2024.3363469>
- [18] Y. E. Seyyar, A. G. Yavuz, and H. M. Ünver, "An attack detection framework based on BERT and deep learning," IEEE Access, vol. 10, pp. 68633–68644, 2022. Available: https://www.researchgate.net/publication/361503157_An_Attack_Detection_Framework_Based_on_BERT_and_Deep_Learning
- [19] M. Guastalla, Y. Li, A. Hekmati, and B. Krishnamachari, "Application of large language models to DDoS attack detection," in Int. Conf. Security and Privacy in Cyber-Physical Systems and Smart Vehicles, Springer, 2023, pp. 83–99. Available: https://doi.org/10.1007/978-3-031-51630-6_6
- [20] Sophos, "Nearly half of malware now use TLS to conceal communications," Sophos News, 2021. Available: <https://www.sophos.com/en-us/blog/nearly-half-of-malware-now-use-tls-to-conceal-communications>
- [21] X. Yuan, S. Han, W. Huang, H. Ye, X. Kong, and F. Zhang, "A simple framework to enhance the adversarial robustness of deep learning-based intrusion detection system," Computers & Security, 2023. Available: <https://arxiv.org/abs/2312.03245>
- [22] "An enhanced ensemble defense framework for boosting adversarial robustness of intrusion detection systems," Scientific Reports, 2025. Available: <https://doi.org/10.1038/s41598-025-94023-z>
- [23] "Mind the gap: Evaluating LLMs for high-level malicious package detection vs. fine-grained indicator identification," arXiv preprint arXiv:2602.16304, 2025. Available: <https://arxiv.org/abs/2602.16304>
- [24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017. Available: <https://arxiv.org/abs/1706.03762>

- [25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018. Available: <https://arxiv.org/abs/1810.04805>
- [26] T. Brown et al., "Language models are few-shot learners," in Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901, 2020. Available: <https://arxiv.org/abs/2005.14165>