An Intelligent Honeypot System for Proactive Threat Detection and Engagement

Sangati Ganga Mahija
Dept. of Computer Science
BNM Institute of Technology
Bengaluru, India
gangamahijasangati@gmail.com

***Abstract*—The increasing sophistication of cyber threats made traditional intrusion observation and defense systems inadequate for modern attack approaches. Honeypot is a cybersecurity technique where a decoy system is created to set up which looks like a real computer, network, or service, helps to attract, detect, and study attackers. The paper hands an intelligent honeypot system augmented with adaptive artificial intelligence to preparedly find, categorize, and involve cyber adversaries. The considered system integrates real-time traffic monitoring, machine learning-based aims segmentation and flexible artifice methods that dynamically transform based on attacker's interaction. Key advancements integrate context-aware record and document creation, sandboxed malware accomplishment for behavioral evaluation, and structured trap services that change to various attack vectors. Moreover, the framework embeds anomaly detection, attacker profiling, and automated reporting to elevate situational awareness for network administrators. By growing attacker attention span and truthfulness, the system obtains deeper threat assessment while residual resilient to elusion. Detailed modeling and evaluation demonstrate that the suggested system decreases false positives, enhances threat classification authenticity, and assists proactive countermeasures. The adaptive honeypot not only notices and inspects attacks but also supplies to predictive cybersecurity defense by schooling from evolving attack patterns. The conclusions emphasize the potential of AI-driven deception systems to transform honeypots from passive traps into active, intelligent cybersecurity mechanisms capable of addressing to rising threats in real time.**

***Index* Terms— Cybersecurity, Honeypot, Adaptive Deception, Threat Detection, Intrusion Analysis, Machine Learning, Artificial Intelligence, Network Security, Sandboxing, Cyber Threat Intelligence**

I. INTRODUCTION

Cybersecurity threats are becoming steadily sophisticated, aiming at crucial infrastructure, enterprise networks, and individualized devices. Conventional defense approaches such as firewalls, antivirus software, and signature-based intrusion detection systems are regularly lacking against modern threats, with advanced persistent threats, zero-day exploits, ransomware, and polymorphic malware. The growing threat landscape has created the need for proactive, intelligent systems capable not only of discovering attacks but also engaging adversaries to gather actionable threat intelligence.

Honeypots, which are decoy systems designed to attract attackers while is detaching them from real production environments, have long offered useful knowledge into threat strategies, malware behavior, and system vulnerabilities. However, conventional honeypots are typically static, predictable, and low-interaction, narrowing their utility in opposition to talented adversaries. Latest advances in artificial intelligence and machine learning now facilitate the formation of adaptive honeypots that responsively react to attacker behavior. These intelligent systems can review network activity in real time, group attacker intent, produces context-aware logs, and modify simulated services or files to prolong engagement, while the integration of sandboxed malware execution and mechanized reporting further improves the quality and extent of acquired threat intelligence.

The intelligent honeypot system proposed here adheres to a structured pipeline designed to facilitate proactive detection and engagement. Stage one, traffic monitoring and capture, continuously observes network traffic as well as new incoming connections towards the honeypot, capturing IP addresses, port scanning, protocol use, as well as session activity in real time. Stage two, preprocessing and feature extraction, cleans the extracted traffic and selects useful features such as command patterns, payload signatures, as well as request frequencies for deeper inspection. In stage three, AI-driven attack classification, machine models such as Random Forests, LSTMs, or Transformer-based classifiers process these features so as to group attacks as reconnaissance, brute-force, SQL injection, malware uploading, or advanced persistent threats.

After classification, adaptive deception and engagement stage creates realistic system responses upon detection of an attack type such as fake logs, system files, credentials, or service behavior, while sandbox execution of malicious payloads ensures safe behavior data collection thus extending attacker engagement. In the last stage, the threat analysis and reporting stage processes the gathered data so as to create actionable insights where attack patterns, attacker profiles, as well as system vulnerabilities are visualized for SOC teams, with feedback from this stage refining the AI models so as to facilitate continuous knowledge as well as adaptive defense.

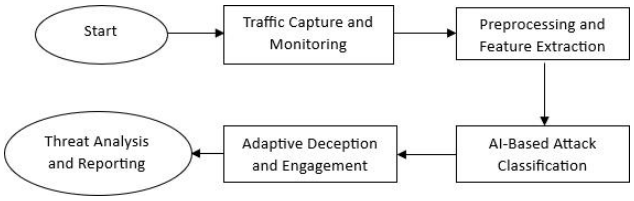

Fig. 1. Intelligent Honeypot processing Pipeline.

The intelligent honeypot uses a pipeline to monitor traffic, classify attacks with AI, and adapt responses to engage attackers. This enables continuous learning and transforms it into an active, adaptive defense system.

II. THEORETICAL FRAMEWORK

The intelligent honeypot system represents the integration of cybersecurity, artificial intelligence (AI), and adaptive systems theory for the proactive discovery, evaluation, and mitigation of cyber threats. Differing fundamentally from traditional static honeypots that lack predictability, intelligent honeypots interact actively with adversaries, collect entire threat intelligence, and adapt their own behavior in real time [1],[2],[3],[4],[5]. This integration turns honeypots into proactive defense tools that increase situational awareness as well as strengthen predictive cybersecurity methods.

- **Node Tracking and Data Gathering:** The system undertakes persistent tracking of network traffic, user action, and system events. It captures vital information such as IP addresses, port scanning behavior, protocols used, session length, payloads, and suspicious patterns. Advanced techniques like deep packet inspection, logging against anomaly detection, and behavior observation are used to increase the accuracy and granularity of the gathered data [6],[7],[8]. Robust data gathering ensures consistent inputs for further AI-based analyses and alleviates blind spots in threat discovery [9].
- **Feature Extraction and Preprocessing:** Raw data from the honeypot often contains noise, incomplete packets, or irrelevant signals. Preprocessing techniques including filtering, normalization, aggregation, and feature extraction—transform this raw data into actionable insights. Key features may include unusual command patterns, frequency anomalies, payload characteristics, and attacker interaction sequences. Effective preprocessing improves AI model accuracy, reduces false positives, and ensures robust classification [10],[11],[12],[13].
- **Attack Classification by AI:** Machine learning (ML) and deep learning algorithms like Random Forests, Support Vector Machines (SVM), LSTM networks, and Transformer models evaluate extracted features in real-time for classifying attacks. Types of classifications can be reconnaissance, brute-force attacks, uploading malware, SQL injection, or advanced persistent threats (APTs) [14],[15],[16]. Knowing the attacker intent as well as behavior helps a system prioritize threats, distribute resources optimally, and facilitate proactive defense action.
- **Adaptive Deception and Engagement:** When an attack is sensed, the honeypot auto-generates realistic decoy responses such as fake files, logs, credentials, or service behavior. Sandboxing malware enables safe viewing of attacker tactics as well as malware behavior. The system can further adjust its response based on attacker tactics observed, extending engagement as far as maximizing the gathering of threat intelligence. Adaptive deception ensures that attackers remain diverted by real assets while delivering usable data back to defenders [17],[18],[19].
- **Threat Analysis, Reporting, and Feedback Loop:** The data that is gathered undergoes thorough analysis to generate practical insights, including the visualization of attack patterns, the identification of attacker profiles, and the exposure of system vulnerabilities. This crucial information is disseminated to security teams and incorporated into security operation centers (SOC). Furthermore, the established feedback loop enables AI models to undergo retraining and adjust in response to new data, thereby enhancing the honeypot system's intelligence, resilience, and ability to anticipate emerging threats [20],[21],[22][23].
- **Integrations with the Cybersecurity Ecosystem:** Smarter honeypots can integrate seamlessly with intrusion detection systems (IDS), firewalls, Security Information and Event Management (SIEM) tools, and threat intelligence systems. By sharing knowledge across the entire cybersecurity ecosystem, this system greatly enhances automated incident responses, enables coordinated threat neutralizations, and supports predictive defense approaches [24].
- **Ethical, Legal, and Privacy Considerations:** Deployment of intelligent honeypots must address privacy and ethical concerns. Attacker data must be anonymized and stored securely, and system design should comply with organizational policies and legal regulations. Ensuring safe use of honeypots prevents misuse and maintains trust in AI-driven cybersecurity solutions [25].

III. HONEYPOT TYPES: A COMPARISON OF LOW-INTERACTION, HIGH-INTERACTION, AND INTELLIGENT HONEYPOTS

Low- and high-interaction honeypots exhibit significant differences in their complexity, levels of interaction, and the nature of the data they gather. Low-interaction honeypots replicate a constrained array of services, are relatively simple to implement, and demand minimal resources; however, they only capture rudimentary attack information and can be more readily identified by advanced attackers. In contrast, high-interaction honeypots offer comprehensive operating environments, permitting attackers to engage extensively with the system, which facilitates the collection of intricate behavioral data, albeit

at the cost of increased maintenance requirements and the need for robust isolation protocols. Intelligent honeypots integrate high-interaction functionalities with AI and machine learning-based analysis, enabling real-time classification of attacks, adaptive responses, and anticipatory threat intelligence, thereby representing the most sophisticated solution available for contemporary cybersecurity [1],[2],[5].

Feature / Type	Low-Interaction	High-Interaction	Intelligent (AI-Driven)
Interaction Level	Low	High	High + Adaptive
Data Collected	Basic attack patterns	Detailed attack behavior	Behavioral + predictive insights
Deployment Complexity	Low	High	Moderate-High
Resource Requirement	Low	High	Moderate-High
Detectability by Attackers	Easily detected	Harder to detect	Hard to detect; adaptive
Typical Applications	Reconnaissance, scanning detection	Malware analysis, attack behavior	Proactive defense, threat intelligence, predictive cybersecurity
Advantage	Easy to deploy, safe	Rich behavioral data	Adaptive, self-learning, real-time defense

Table 1 Key Differences Between Honeypot Types

IV. SURVEY OF EXISTING WORK

A. Rule-Based Systems/Classical Honeypots

The traditional honeypots utilize static settings and known protocols in order to record malicious activity. These are effective against simple, automated attacks but struggle when up against more transient or adaptable malicious activity.

- Low-Interaction Honeypots (LIH): These simulate low network services like (SSH, HTTP, FTP) and store superficial attacker activities. Examples are Kippo and Honeyd [1], [2].
  - Strengths: Simple to deploy, lightweight, utilize few resources, and provide low operational risk as the attacker can't totally subvert them.
  - Limitations: Shallow engagement; they can be fingerprinted easily by adversaries, diminishing long-term success.
- Signature and Rule-Based Detection Honeypots: Honeypots like Gasthof [3], [4] use predefined attack signatures or heuristics guidelines for determining malicious activity. These honeypots prove very effective against known vulnerabilities as well as exploit kits.
  - Strengths: Simple deployment, actionable alerts, can be used for training or small installations.
  - Limitations: Not effective against zero-days, polymorph viruses/malware, or adaptive adversaries. Signature library upgrades are reactive rather than proactive and consume vast system resources.

B. Machine Learning-Based Honeypots

In order to break the static nature of traditional honeypots, scientists incorporated machine learning (ML) for enhanced adaptability as well as detection capability.

- **Shallow Machine Learning Models:** Traditional machine learning approaches, such as Support Vector Machines (SVMs), Random Forests (RF), Decision Trees, and k-Nearest Neighbours (k-NN), have been employed for the investigation on honeypot logs [5], [6]. Extracted features often include packet numbers, connection lifetimes, inter-request intervals, and byte-level statistics.
  - Strengths: Provide higher detection accuracy than static rules, are relatively interpretable, and work well on small- to medium-sized datasets.
  - Limitations include the necessity for comprehensive feature engineering, insufficient scalability when handling high-dimensional data, and a deficiency in adaptability to swiftly changing attack strategies.
- **Time-Series / Sequential Models:** Those advanced deep learning models such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Temporal Convolutional Networks (TCNs) can identify sequential attacker behaviour patterns [7], [8]. These can detect multi-step intrusions as well as brute-force attacks and extended lateral movement activities.
  - **Strengths:** Extract temporal relationships in attack behaviour, so they can be useful against stealthy or stage-structured intrusions.
  - **Limitations:** Requires large labelled datasets, computationally expensive, and susceptible to concept drift when operating in real-world dynamic threat landscapes.

C. Artificial Intelligence-Enhanced Honeypots and Deep Learning

Latest studies focus on applying deep learning as well as reinforcement learning in developing intelligent adaptive honeypots that keep on learning by studying attacker behavior.

- **Deep Neural Networks (DNNs):** Used on massive honeypot datasets for automatic feature extraction as well as attack classification [9].
- **Convolutional Neural Networks (CNNs):** Ideal for malicious payload detection when considering packet-level or byte-level representation [10].
- **Reinforcement Learning (RL):** Utilized for creating adaptive deception policies where the honeypot learns the best responses for maximizing attacker interest along with harvesting intelligence of a high-value intelligence [11].
- **Generative Adversarial Networks (GANs):** can create realistic traffic as well as fake responses in order to bolster the credibility of honeypots [12].
  - **Strengths:** Excellent adaptability, ability to counter complex attack vectors, and diminished manual rule update reliance.
  - **Limitations:** Supervised training necessitates big, nicely balanced datasets; susceptible to adversarial ML assaults; heavy resource usage.

Honeypot Type	Best Used For	Strengths	Limitations	Typical Applications
Low-Interaction	Reconnaissance, automated attacks	Lightweight, easy to deploy	Limited realism; easily fingerprinted	Perimeter monitoring, worm detection
High-Interaction	Deep malware analysis, forensic investigation	Captures rich attacker behavior	High resource cost; risk of compromise	Malware labs, advanced attack study
AI/ML-Driven	Real-time adaptive deception	Intelligent adaptation, automated classification	Requires large datasets, high compute	SOCs, enterprise defense
IoT Honeypots	Smart device botnets & malware	Protocol-specific, lightweight	Limited realism, niche-focused	Smart home, healthcare IoT
Cloud Honeypots	Virtualized cloud attack detection	Scalable, low hardware cost	Susceptible to fingerprinting if misconfigured	Cloud security, hybrid infrastructures
ICS/SCADA Honeypots	Industrial protocol monitoring	Domain-specific deception, critical protection	Complex configuration, high-stakes environments	Power grids, industrial plants, transport
Wireless/Mobile	Wi-Fi, GSM, Bluetooth attack capture	Captures unique mobile/wireless threats	Limited maturity, early-stage research	Telecom, wireless networks
Distributed Honeynets	Shared multi-org defense	Collaborative intelligence, large-scale visibility	Coordination challenges, latency	Threat intelligence sharing, ISACs

Table II Comparison of Honeypot Architectures

D. Highly Specialized Honeypots (IoT, Cloud, ICS/SCADA)

When the aggressors began targeting IoT, cloud, and industrial systems, scientists created domain-specific honeypots.

- **IoT Honeypots:** These mimic IoT devices (home assistants, routers, smart cameras) and are highly effective against botnets such as **Mirai** [13].
- **Cloud Honeypots:** Used on AWS, Azure, or OpenStack infrastructures for identifying cloud-native attacks, misconfigurations, as well as privilege escalation [14].
- **ICS/SCADA Honeypots:** Mimic protocols like Modbus, DNP3, and IEC 104, allowing tracking of attacks against the vital infrastructure [15].
  - **Strengths:** Offer domain-specific knowledge, appeal to actual attackers who prey on niche ecosystems.
  - **Limitations:** Costliness in deployment and maintenance, chances of disrupting sensitive infrastructures.

E. Distributed & Collaborative Honeypots

Distributed honeypot networks (honeynets) allow large-scale deployment and **collaborative intelligence sharing**.

- **Distributed Honeynets:** Deployed over several organizations, allowing for attack global campaigns as well as botnet propagation detection [16].
- **Federated Honeypots:** Collect together AI-driven honeypots in separate territories where models get trained on-site but share aggregated intelligence [17].
  - **Strengths:** Overall big-picture visibility, faster recognition of new adversaries and weaknesses, reduced dataset bias.
  - **Limitations:** Data-sharing introduces **privacy and security concerns**; coordination across entities can be challenging.

F. Consideration on Privacy, Ethics, and Security

Deploying honeypots raises ethical and legal questions. Research highlights issues of **data privacy, consent, and liability** in real-world deployments [18], [19]. While honeypots are invaluable for defense, improperly secured ones may serve as **launchpads for attacker pivoting**, potentially endangering legitimate systems. To mitigate risks, best practices include **controlled isolation, anonymization of captured data, and encrypted storage** [20].

V. DISCUSSION, CHALLENGES, AND FUTURE DIRECTIONS

A. Discussion

Honeypots once were just trivial, rule-based decoys but now are advanced, adaptive deception tools that incorporate machine learning, deep learning, and domain-specific constructions. Old-style low-interaction honeypots such as Honeyd offered minimal attack visibility but new methods use AI-driven adaptive behavior, cloud-native deployment, and collaborative architectures.

The incorporation of ML/DL methods greatly enhanced detection rates and versatility so that now honeypots can discover zero-day attacks, polymorphic viruses, as well as advanced APTs. Domain-specific honeypots for IoT devices, cloud infrastructures, as well as ICS/SCADA systems show that increasing customization of deception tools across various fields is required. Distributed honeynets, meanwhile, facilitate mass scale knowledge gathering as well as worldwide situational consciousness.

B. Challenges

Despite advances, honeypot research and deployment face several persistent challenges:

- **Fingerprinting and Detection:** Highly advanced adversaries employ honeypot fingerprinting techniques that allow them to evade and identify decoys, thereby reducing their effectiveness.
- **Labelling and Quality of Data:** Machine-learning-oriented honeypots need big labelled datasets, yet attack traffic frequently happens to be imbalanced, noisy, or malicious in its nature**.**
- **Scalability in Cloud/IoT:** Deploying honeypots at scale in **cloud-native and IoT ecosystems** requires significant resources and seamless integration with production systems.
- **Legal and Ethical Considerations:** Honeypots can raise entrapment concerns, violation of privacy, as well as liability if the attacker interactions are published or leaked.
- **Security Risks:** Weekly isolated honeypots can then become an attack vector exploited by hackers as a springboard for further attacks on genuine systems.
- **Adaptive Adversaries:** As With growing usage Evasion methods, then, must evolve as well faster to remain effective.

C. Future Directions

To address these challenges, several promising research directions emerge:

- **AI-Enhanced Deception:** Combining reinforcement learning (RL) and generative adversarial networks (GANs) for the creation of self- adaptive honeypots capable of adapting in real-time against adversaries.
- **Hybrid Honeypots:** Combining high- and low-interaction honeypots with layered architectures, enhanced scalability as well as deepening engagement.
- **Edge and IoT Honeypots:** Lightweight IoT device, smart home, and industrial sensor honeypots that can defend against botnets as equally as ICS-targeted malware.
- **Cloud-Native Honeypots:** Deception tools designed keeping Kubernetes, containerized workloads, and serverless deployments in consideration that can discover cloud.
- **Federated Honeypots:** Cooperative networks where various organizations securely share attack intelligence collaboratively, increasing early discovery of global campaigns without sacrificing privacy.
- **Explainable AI for Honeypots:** Utilizing XAI methods for enhanced interpretability of ML-based honeypot detection so as to reduce the automation confidence-analyst confidence gap.
- **Security and Privacy by Design:** Integrating legal, ethical, and security safeguards in honeypot deployments in an effort to mitigate liability and grow credibility.
- **Integrations with Threat Intelligence Platforms:** Seamless interconnection with SIEMs, SOAR tools, and CTI feeds for providing actionable intelligence for pre-emptive defence.

IV. CONCLUSION

Honeypots evolved from rudimentary, rule-based decoys to advanced, intelligent systems that can identify and dissect sophisticated cyber threats. Initial implementations were mainly proof-of-concept tools for gathering limited attack evidence, yet recent innovations in machine learning, deep learning, and automation have made them an integral part of next-generation cybersecurity defense. By offering contained environments that entice attackers without risking production systems, honeypots will remain a significant asset for examining adverse tactics as well as creating threat intelligence that can be acted upon.

Not with standing these advances, numerous significant challenges persist such as adversarial fingerprinting, scalability in IoT as well as cloud settings, as well as the legal/ethical implications of deception-oriented security.

Attackers increasingly use AI-motivated evasion methods, pushing honeypots to increase more than static signatures and manually created rules. Additionally, maintaining data quality, system segregation assurance, as well as preventing misuse of gathered data, continue to be unending issues that hamper mass deployment. Overcoming these challenges will help maintain the reliability and credibility of honeypot-driven systems.

The future thus for honeypot research is in the creation of adaptive deception systems that scale securely and interoperable across larger cybersecurity ecosystems and that are artificial intelligence-driven. Prospects such as federated honeypots, cloud-native systems, and IoT/ICS deception systems portend an integrated and intelligent future direction.

Through the integration with explainable artificial intelligence, threat intelligence sharing, as well as ethical safe-guards, honeypots can transition from being passive traps towards being proactive, resilient tools for cyber defence that can no longer be avoided.

REFERENCES

[1] **Kubba, A., Nasir, Q., Elmutasim, O., & Abu Talib, M.** (2025). A systematic review of honeypot data collection, threat intelligence platforms, and AI/ML techniques.
[2] **Lanz, S.** (2025). Optimizing Internet of Things honeypots with machine learning. *MDPI Electronics*, 15(10), 5251.
[3] **Alatawi, E.** (2025). Honeypot-driven intrusion detection systems. *Mathematics*, 17(5), 628.
[4] **Iyer, K. I.** (2021). Adaptive honeypots: Dynamic deception tactics in modern cyber defence. *International Journal of Science and Research Archive*, 4(1), 340-351.
[5] **Ebunoluwa, A.** (2025). AI-powered honeypots: Enhancing deception technologies for cyber defence.
[6] **Smith, J. A., Johnson, E. R., Brown, M. T., Davis, L. K., & Castro, H.** (2025). AI-driven honeypot architectures for next-generation intrusion detection and prevention.
[7] **Bouarfa, A.** (2025). Intelligent honeypot-based IDS for cyber-attack detection.
[8] **Morić, Z.** (2025). Advancing cybersecurity with honeypots and deception technologies. *MDPI Electronics*, 12(1), 14.
[9] **Morozov, D. S.** (2024). The sweet taste of IoT deception: An adaptive honeypot framework for IoT environments. *JEC Journal of Engineering and Computing*, 9(2), 607.
[10] **Panda, S., Rass, S., Moschoyiannis, S., Liang, K., Loukas, G., & Panaousis, E.** (2021). HoneyCar: A framework to configure honeypot vulnerabilities on the Internet of Vehicles.
[11] **Srinivasa, S., Pedersen, J. M., & Vasilomanolakis, E.** (2021). Gotta catch 'em all: A multistage framework for honeypot fingerprinting.
[12] **Crespi, V., Hardaker, W., Abu-El-Haija, S., & Galstyan, A.** (2021). Identifying botnet IP address clusters using natural language processing techniques on honeypot command logs.
[13] **Castro, H., & Brown, M. T.** (2025). AI-driven honeypot architectures for next-generation intrusion detection and prevention.

[14] **Bouarfa, A.** (2025). Intelligent honeypot-based IDS for cyber-attack detection.

[15] **Morić, Z.** (2025). Advancing cybersecurity with honeypots and deception technologies. *MDPI Electronics*, 12(1), 14.

[16] **Morozov, D. S.** (2024). The sweet taste of IoT deception: An adaptive honeypot framework for IoT environments. *JEC Journal of Engineering and Computing*, 9(2), 607.

[17] **Panda, S., Rass, S., Moschoyiannis, S., Liang, K., Loukas, G., & Panaousis, E.** (2021). HoneyCar: A framework to configure honeypot vulnerabilities on the Internet of Vehicles.

[18] **Srinivasa, S., Pedersen, J. M., & Vasilomanolakis, E.** (2021). Gotta catch 'em all: A multistage framework for honeypot fingerprinting.

[19] **Crespi, V., Hardaker, W., Abu-El-Haija, S., & Galstyan, A.** (2021). Identifying botnet IP address clusters using natural language processing techniques on honeypot command logs.

[20] **Castro, H., & Brown, M. T.** (2025). AI-driven honeypot architectures for next-generation intrusion detection and prevention.

[21] **Bouarfa, A.** (2025). Intelligent honeypot-based IDS for cyber attack detection.

[22] **Morić, Z.** (2025). Advancing cybersecurity with honeypots and deception technologies. *MDPI Electronics*, 12(1), 14.

[23] **Morozov, D. S.** (2024). The sweet taste of IoT deception: An adaptive honeypot framework for IoT environments. *JEC Journal of Engineering and Computing*, 9(2), 607.

[24] **Panda, S., Rass, S., Moschoyiannis, S., Liang, K., Loukas, G., & Panaousis, E.** (2021). HoneyCar: A framework to configure honeypot vulnerabilities on the Internet of Vehicles.

[25] **Srinivasa, S., Pedersen, J. M., & Vasilomanolakis, E.** (2021). Gotta catch 'em all: A multistage framework for honeypot fingerprinting.