B.M.S. COLLEGE OF ENGINEERING

Department of Computer Science and Engineering

Major Project Phase 1

(22CS7PWMP1)

Synopsis Report

On

HoneyGAN Pots: A Deep Learning Approach for Generating Honeypots

Submitted by:

Aastha Priya	**Aneesh K P**	**Anup Vaidya**	**Araga Laxman Anirudhadithya**
1BM22CS003	1BM22CS040	1BM22CS047	1BM22CS050

Under the Guidance of:

Guide Name: Dr. Asha G.R

Designation: Associate Professor

INRODUCTION

Cybersecurity requires constant innovation to combat evolving threats. Honeypots, which are decoy systems used to detect and analyse cyberattacks, play a critical role in threat intelligence. However, traditional honeypots rely on static configurations that lack adaptability, making them less effective against sophisticated attackers. This project introduces an approach by leveraging **Generative Adversarial Networks (GANs)** to dynamically generate realistic honeypot configurations that mimic real-world systems, enhancing deception and threat analysis.

Feasibility:

GANs provide a scalable and automated method for generating diverse honeypot configurations, reducing manual effort and improving adaptability. With the growing computational power available, deploying the GANs can be done on cloud services such as AWS or Azure. Honeypots can be deployed in the most vulnerable devices or spots of the network so that it can easily communicate with the existing firewalls regarding the attacks.

Relevance:

By continuously learning and improving based on real-time attacker data, GAN-enhanced honeypots provide a significant advantage in identifying and mitigating advanced threats. This approach keeps us one step ahead in network security, making it highly applicable across industries such as finance, healthcare, and critical infrastructure, and also governments where robust cybersecurity measures are essential.

Contribution to SDGs:

By enhancing cyber defences, this project contributes to **SDG 9 (Industry, Innovation, and Infrastructure)** and **SDG 16 (Peace, Justice, and Strong Institutions)** by ensuring secure digital ecosystems and promoting resilient infrastructure.

There are few research papers that leverages Generative Adversarial Networks (GANs), trained on real-world network device configurations, to enhance attack detection. Building on this foundation, we aim to use honeypots to gather real-time attacker data and feed it into the GAN, enabling continuous learning and adaptation to evolving cyber threats, thereby improving attack detection overtime.

REQUIREMENTS

The primary objective of this project is to develop a Generative Adversarial Network (GANs) to effectively detect and mitigate various types of cyberattacks, including Distributed Denial-of-Service (DDoS) attacks, Man-in-the-Middle (MITM) attacks, and IP spoofing attempts that computer networks are vulnerable to. The project aims to:

Simulate and Detect Modern Cyber Threats: Deploy honeypots in a network to simulate and analyze malicious activity, capturing and categorizing various types of attacks for model training.

Enhance Threat Detection using GANs: Utilize network device configurations dataset from various sources like Shadon Search Engine and KDD99, enhancing the diversity of data to improve the IDS's capability in identifying new and unknown attack patterns.

Automate and Optimize Honeypot Deployment: Integrate DevOps pipelines to enable seamless and automated deployment of GANs across cloud environments, ensuring dynamic and adaptive responses to potential attacks.

Evaluate System Effectiveness and Efficiency: Measure the system's performance in terms of detection accuracy, false-positive rate, and response time under various simulated attack scenarios.

Methodological Boundaries:
Conceptual Boundary: Focus on preventing IP spoofing, DDoS, and DNS attacks using intelligent honeypots and machine learning techniques.
Analytical Boundary: Application of GANs to create adversarial attack datasets and improve IDS performance.
Experimental Boundary: Simulated attack scenarios and deployment in cloud environments such as AWS or Azure to validate the system.
Time-bound Scope: The project is planned to be executed over 8-9 months, with milestones covering design, development, testing, and evaluation phases.

REQUIREMENTS

Hardware Requirements:

Local (On-Premise) Setup

- CPU: Intel i7/i9 or AMD Ryzen 7/9
- RAM: 16GB or more
- Storage: 250GB SSD
- GPU: NVIDIA RTX 3060+

A. Cloud (AWS/GCP/Azure) Setup

- Compute Instance: 4+ vCPUs, 16GB RAM
- GPU for ML Training: NVIDIA T4 / V100
- Storage: 500GB SSD
- Security: Cloud Firewall, AWS Shield, Cloudflare WAF

Software Requirements

B. Honeypot Deployment

- Cowrie – SSH/Telnet Honeypot
- Dionaea – Malware & Exploit Detection
- T-Pot – Multi-honeypot Framework

C. Machine Learning & GANs

- Python (3.8+)
- TensorFlow / PyTorch
- Scikit-learn, Pandas, NumPy
- Jupyter Notebook
- Matplotlib, Seaborn (for visualization)

D. Security & Network Monitoring

- Suricata / Snort – Intrusion Detection System
- Wireshark / Tcpdump – Packet Capture

E. Cloud Services & Deployment

- AWS EC2 / GCP Compute – Cloud Instance for Honeypot
- AWS S3 / GCS – Log Storage
- Kubernetes / Docker – Containerized Deployment

COST ESTIMATE

The cost estimation for the project has been defined in the table below with both the minimal cost required and the maximal cost it may potentially go up to for each category specific to it.

Component	Minimum Cost	Maximum Cost
AWS Cloud Services	₹2,000	₹7,000
Storage	₹1,500	₹5,000
AWS Cloud Computing for GAN	₹1,000	₹2,000
Total	₹4,500	₹14,000

WORKPLAN

The workplan to complete the project has been divided into a total of 7 phases, with the total duration of the project surmounting to an approximate of 8-9 months.

The following phases are mentioned below:

Phase 1: Project Initiation and Planning

Duration: 2 weeks

Objective: Establish the project foundation and gather resources quickly.

Phase 2: Data Collection and Preparation

Duration: 4 weeks

Objective: Collect and preprocess data efficiently.

Phase 3: GAN Architecture Design and Development

Duration: 5 weeks

Objective: Design and implement GAN models rapidly.

Phase 4: Training and Optimization

Duration: 8 weeks

Objective: Train GANs efficiently with real data.

Phase 5: Evaluation and Validation

Duration: 6 weeks

Objective: Assess GAN performance and honeypot quality quickly.

Phase 6: Documentation and Conclusion

Duration: 5 weeks

Objective: Compile results and paper efficiently.

Phase 7: Presentation and Dissemination

Duration: 3 weeks

Objective: Share findings with minimal delay.

REFERENCES

[1] Gabrys, R., Silva, D., & Bilinski, M. (2024b, July 10). HoneyGAN Pots: a deep learning approach for generating honeypots. arXiv.org. https://arxiv.org/abs/2407.07292

[2] M Karthigha, L Latha, Sripriyan K et al. Intelligent Honeypot-based IDS for Cyber Attack Detection using Generative Adversarial Networks, 10 June 2024, PREPRINT (Version 1) available at Research Square [https://doi.org/10.21203/rs.3.rs-4370529/v1]