|  |  |  |
| --- | --- | --- |
|  | **TABLE OF CONTENTS** |  |
|  |  |  |
|  | **ABSTRACT** | **ix** |
|  | **LIST OF FIGURES** | **x** |
|  | **ABBREVIATIONS** | **xi** |
|  | **1 INTRODUCTION** | **1** |
|  | 1.1  The Emerging Threat of DDoS Attacks for IoT Networks | 1 |
|  | 1.2  Impact of Real-time Detection and Mitigation | 1 |
|  | 1.3  Dynamic Fuzz Testing Framework with AI | 1 |
|  | 1.4  Contribution and Future Prospects | 2 |
|  | **2      LITERATURE SURVEY** | **3** |
|  | 2.1  IoT Security Challenges | 3 |
|  | 2.2  DDoS Attacks on IoT Networks | 3 |
|  | 2.3  AI-Based IoT Security Solutions | 4 |
|  | 2.4  Dynamic Fuzz Testing for IoT Security | 5 |
|  | 2.5  NS3 Simulations for IoT Security | 6 |
|  | **3      SYSTEM ARCHITECTURE AND DESIGN** | **7** |
|  | 3.1  Layered Architecture Overview | 7 |
|  | 3.1.1   Device Layer | 7 |
|  | 3.1.2  Security and Detection Layer | 7 |
|  | 3.1.3  Fuzz Testing Layer | 7 |
|  | 3.1.4   Mitigation Layer | 8 |
|  | 3.1.5   Edge Computing and Analytics Layer | 8 |
|  | 3.1.6   Cloud Processing Layer | 8 |
|  | 3.2  Core Components of FuzzAIoT | 8 |
|  | 3.2.1  Graph Neural Networks (GNNs) | 8 |
|  | 3.2.2  Fuzz Testing Engine | 9 |
|  | 3.2.3   Traffic Filtering Module | 9 |
|  | 3.2.4   Mitigation Engine | 9 |
|  | 3.3  Scalability and Performance Considerations | 9 |
|  | 3.3.1   Scalability | 9 |
|  | 3.3.2  Low Latency Operations | 9 |
|  | 3.3.3   Adaptability | 10 |
|  | 3.4  Integration with Existing Infrastructure | 10 |
|  | 3.4.1   Cloud and Edge Integration | 10 |
|  | 3.4.2   Legacy System Compatibility | 10 |
|  | 3.5  Summary of System Design | 10 |
|  | **4        METHODOLOGY** | **11** |
|  | 4.1  Addressing IoT Security using AI-Driven Techniques | 11 |
|  | 4.2  Dynamic Fuzz Testing | 11 |
|  | 4.2.1   Introduction to Dynamic Fuzz Testing | 11 |
|  | 4.2.2   Data Generation through Fuzz Testing | 13 |
|  | 4.3  Dataset Preprocessing | 13 |
|  | 4.3.1   XML to CSV from Simulation Data | 13 |
|  | 4.3.2   Data Labelling and Balancing | 13 |
|  | 4.4  Training AI Model | 14 |
|  | 4.4.1  Model Architecture | 14 |
|  | 4.4.2   Training Process and Hyperparameter Tuning | 15 |
|  | 4.4.3   Validation | 15 |
|  | 4.5  Simulation Environment | 15 |
|  | 4.5.1   Network Setup | 15 |
|  | 4.5.2   Traffic Simulation | 16 |
|  | 4.5.3   Visualization with NetAnim | 16 |
|  | 4.6  Mitigation Strategy | 16 |
|  | 4.6.1   Blocking of malicious IPs | 16 |
|  | 4.6.2   Evaluation and Refinement | 16 |
|  | **5        CODING AND TESTING** | **17** |
|  | 5.1  Simulation Setup in NS3 | 17 |
|  | 5.1.1   Network Topology Design | 17 |
|  | 5.1.2  Traffic Generation | 18 |
|  | 5.1.3   Mobility Model | 20 |
|  | 5.1.4   Setting up NetAnim | 20 |
|  | 5.2  Dataset Generation | 20 |
|  | 5.2.1   Traffic Data Collection | 20 |
|  | 5.2.2   Data Conversion into CSV Format | 21 |
|  | 5.2.3   Data Balancing | 24 |
|  | 5.3  AI Model Training | 24 |
|  | 5.3.1   Model Architecture | 25 |
|  | 5.3.2   Training Process | 26 |
|  | 5.3.3   Validation | 26 |
|  | 5.4  Mitigation in NS3 – Real Time detection and Action | 27 |
|  | 5.4.1   Detection of Malicious IP Address | 27 |
|  | 5.4.2  Packet Filtering | 27 |
|  | 5.5  Conclusion of Testing and Implementation | 28 |
|  | **6        RESULT AND DISCUSSIONS** | **29** |
|  | 6.1  Introduction | 29 |
|  | 6.2  Results | 29 |
|  | 6.2.1  Training and Accuracy of the GNN Model | 29 |
|  | 6.2.2   Results of Validation | 30 |
|  | 6.2.2.1  Validation Accuracy | 30 |
|  | 6.2.2.2  Confusion Matrix | 30 |
|  | 6.2.3   Mitigation Strategy | 31 |
|  | 6.2.3.1  Packet Dropping | 31 |
|  | 6.2.3.2  Performance on Network | 31 |
|  | 6.3  Visualization of Results | 31 |
|  | 6.3.1  Loss and Accuracy Graphs | 31 |
|  | 6.3.2   Confusion Matrix | 33 |
|  | 6.3.3   Network Animation | 34 |
|  | **7        CONCLUSION AND FUTURE ENHANCEMENT** | **35** |
|  | 7.1  Summary of Headings | 35 |
|  | 7.1.1  AI-Driven Dynamic Fuzz Testing | 35 |
|  | 7.1.2   NS3 Simulation and Visualization | 36 |
|  | 7.1.3  Counter Measures | 36 |
|  | 7.2  Future Improvements | 37 |
|  | 7.2.1   Real-Time Mitigation | 38 |
|  | 7.2.2  Elaborated Attack Scenarios | 38 |
|  | 7.2.3   Scalability Test | 38 |
|  | 7.2.4   Adaptive Learning | 38 |
|  | 7.2.5   Integration of Security Tool | 39 |
|  | 7.3  Conclusion | 39 |
|  | **REFERENCES** | **40** |
|  | **APPENDIX** |  |
|  | **A CONFERENCE PUBLICATION** | **43** |
|  | **B PLAGIARISM REPORT** | **44** |