***NCY2***

***PROJECT REPORT***

***WORMHOLE DETECTION SYSTEM***

***BY k213602***

***HUZAIFA GHORI***

**Wormhole Detection in Wireless Sensor Networks Using Spanning Trees**

**Introduction**

Wireless Sensor Networks (WSNs) are widely used in various applications such as environmental monitoring, healthcare, and surveillance. However, they are vulnerable to security threats, including wormhole attacks. A wormhole attack involves malicious nodes colluding to create a shortcut in the network, compromising its security and integrity. In this project, we propose a wormhole detection system based on spanning trees and Euclidean distance analysis to identify and mitigate such attacks.

**Objective of this system**

The primary objective of this project is to develop a wormhole detection system capable of detecting and mitigating wormhole attacks in Wireless Sensor Networks. The system utilizes spanning trees and Euclidean distance analysis to identify affected sensors and visualize the network topology.

**Methodology**

The project follows these key steps:

**WSN Generation**

The system generates a random WSN on a 2D plane to simulate sensor nodes deployed in an area.

This is achieved by generating random coordinates for each sensor node within the specified area size.

**Distance Calculation**

After generating the sensor nodes, the system calculates the distances between all pairs of sensor nodes.

This distance calculation is based on the Euclidean distance formula, which computes the straight-line distance between two points in a plane.

**Wormhole Insertion**

The system identifies potential wormholes in the network based on predefined criteria.

A potential wormhole is identified if a group of sensor nodes is within a certain distance threshold of each other. In this implementation, if at least 6 nodes are within this threshold, they are considered potential wormholes.

These potential wormholes are stored as a list of wormhole nodes.

**Affected Sensor Detection**

Using spanning trees and Euclidean distance analysis, the system detects affected sensors that are likely to be impacted by the wormholes.

It constructs a spanning tree of the network using networkx library in Python.

Then, it analyzes the connectivity of each potential wormhole to other sensor nodes in the network.

Affected sensors are detected based on their connectivity to potential wormholes through the spanning tree.

**User Authentication**

Before accessing the network and viewing the results, users are prompted to authenticate themselves with a username and password.

Default username and password are provided for ease of demonstration.

**Logging**

Viewer information, including user details and network characteristics, is logged into a file for monitoring and analysis.

This information includes the user's name, the number of devices in the network, the number of wormholes detected, and the details of each wormhole (including its distance to other devices).

**Visualization**

Finally, the system visualizes the network topology using matplotlib library in Python.

It generates two figures: one showing only the wormhole detection and the other showing the complete network.

Detected wormholes are highlighted in red, affected sensors in green, and normal sensors in blue.

The network topology is displayed with labels indicating the device number and distances between devices.

**SECURITY ASPECTS**

**Data Confidentiality**

Ensuring that sensitive data transmitted within the network remains confidential and cannot be intercepted or compromised by unauthorized entities, including potential wormhole attackers.

**Data Integrity**

Verifying that the data collected from sensor nodes is not altered or tampered with during transmission. This prevents attackers from injecting false data or modifying legitimate data packets.

**Authentication**

Verifying the identity of sensor nodes and ensuring that only authorized nodes have access to the network resources. Authentication mechanisms help prevent unauthorized nodes, including potential wormhole attackers, from joining the network

**Secure Communication**

Implementing secure communication protocols to encrypt data transmission between sensor nodes. This prevents eavesdropping and ensures that data exchanged within the network is protected from unauthorized access.

**Network Resilience**

Designing the network architecture to be resilient against various security threats, including wormhole attacks. This may involve implementing redundant communicationpaths, dynamic routing protocols, and intrusion detection systems to detect and mitigate attacks in real-time.

**Attack Detection and Mitigation**

Developing algorithms and techniques to detect and mitigate potential wormhole attacks in the network. This includes analyzing network topology, traffic patterns, and cryptographic signatures to identify anomalies indicative of wormhole attacks and take appropriate countermeasures.

**Physical Security**

Protecting the physical infrastructure of the network, including sensor nodes and communication channels, from unauthorized access and tampering. Physical security measures such as tamper-evident seals, secure enclosures, and surveillance systems help prevent physical attacks on the network.



**FIG 1 : AUTHENTICATION**

**Monitoring and Logging**

Implementing monitoring and logging mechanisms to track network activity, detect security incidents, and record relevant information for forensic analysis. This enables network administrators to identify security breaches, investigate the root cause of attacks, and take remedial actions to prevent future incidents.



**FIG 1.2: AUTO CREATION OF FILE FIG 2: VERIFICATION**



**FIG 3: AUTHENTICITY**

**IMPLEMENTATION**



**FIG 4 : LOG VIEWER’S NAME**

The implementation of the wormhole detection system provides a comprehensive solution for identifying and mitigating wormhole attacks in Wireless Sensor Networks. By utilizing spanning trees and Euclidean distance analysis, the system effectively detects potential security threats and provides valuable insights into the network topology. This implementation serves as a useful tool for network administrators to monitor and secure WSNs against malicious attacks.

THEREFORE, the project is implemented in Python. It utilizes libraries such as numpy, matplotlib, and networkx for network generation, visualization, and analysis. The main functionalities include network generation, wormhole detection, user authentication, logging, and visualization.

**Results**

The system successfully detects potential wormholes and affected sensors in the network. It provides visualizations of the network topology, highlighting detected wormholes and affected sensors. Viewer information, including user details and network characteristics, is logged into a file for further analysis.



**FIG 5: WARMHOLE DETECTION AND COMPLETE NETWORK**



**FIG 6: DISTANCE SHOWN BETWEEN DEVICES IN A MATRIX**



**FIG 7: DATA IN THE FILE**

**Conclusion**

The project demonstrates an effective wormhole detection system for Wireless Sensor Networks. By utilizing spanning trees and Euclidean distance analysis, it identifies potential security threats and mitigates wormhole attacks. The system provides a valuable tool for network administrators to monitor and secure WSNs against malicious attacks.

**References**

Y. S. Kim, B. K. Park, and D. Y. Lee, "Wormhole Detection in Wireless Sensor Networks Based on Packet Delivery Rate," *2007 IEEE International Conference on Communications*, pp. 4874-4879, 2007. (DOI: 10.1109/ICC.2007.571)

Y. N. Reddy, P. C. Reddy, and K. K. Reddy, "A Novel Approach for Wormhole Attack Detection in Wireless Sensor Networks," *2020 IEEE 8th International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 134-139, 2020. (DOI: 10.1109/WiSPNET48685.2020.9154482)