**SinkHole Attack in LEACH**

**Software Recommended:** NetSim standard v13.2, Visual Studio 2022

**Project Download Link:**

Follow the instructions specified in the following link to download and setup the Project in NetSim:

[https://support.tetcos.com/en/support/solutions/articles/14000128666-downloading-and-setting-upnetsim-file-exchange-projects](https://support.tetcos.com/support/solutions/articles/14000128666-downloading-and-setting-up-netsim-file-exchange-projects)

**Low – Energy Adaptive Clustering hierarchy(“LEACH”):**

Leach is a MAC protocol which is integrated with clustering and a simple routing protocol in wireless sensor networks (WSN). The goal of LEACH is to lower the energy consumption required to create and maintain clusters in order to improve the lifetime of a wireless sensor network.

This Cross Layer Protocol is implemented in NetSim in MAC layer which involves ZigBee Protocol and Network layer which involves DSR protocol. The clustering of sensors happens in the Network Layer and the cluster head election involves interacting with the MAC layer to obtain the remaining power of the sensors.

A **LEACH.c** file is added to the DSR project.

1. For this implementation of LEACH, the number of Clusters is fixed as 4 and all the 4 clusters are equal. If the user wants to change it, then he/she must also change the static routing for the Cluster Heads and the ClusterElement array accordingly.

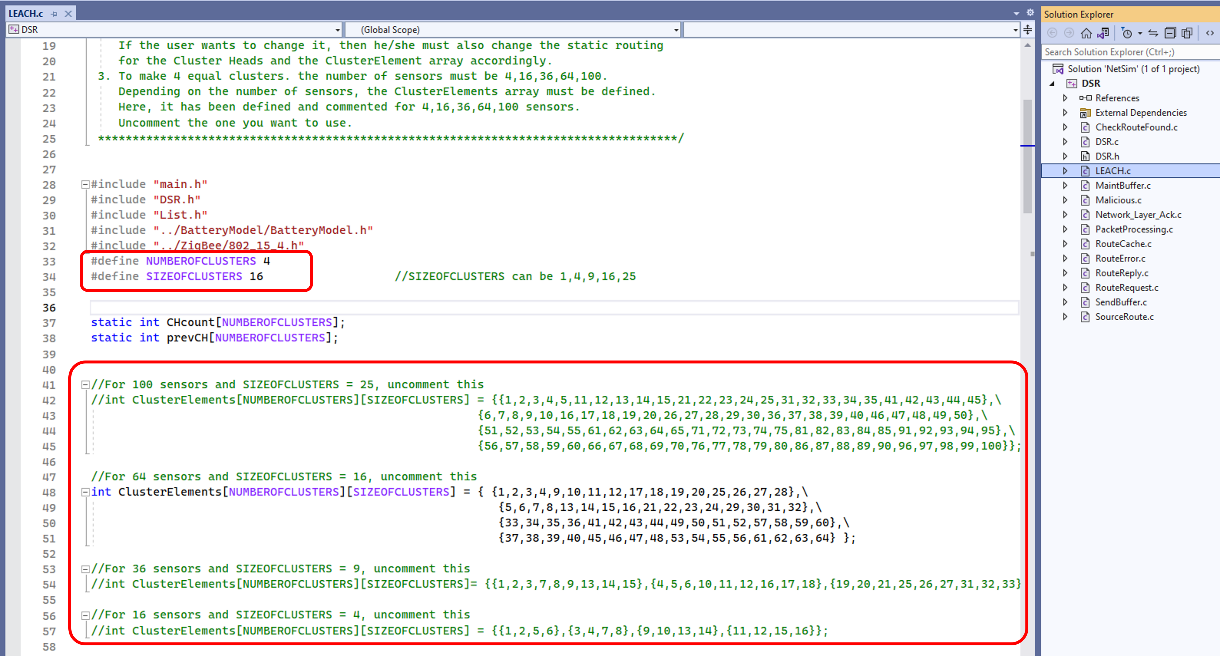


Figure :Leach.c file in source code

1. To make 4 equal clusters the number of sensors must be 4,16,36,64,100. Depending on the number of sensors, the ClusterElements array must be defined. Here, it has been defined and commented for 4,16,36,64,100 sensors. Uncomment the one you want to use.

The File contains the following functions:

**fn\_NetSim\_LEACH\_CheckDestination() //** to check whether the current device is the destination or not.

**fn\_NetSim\_LEACH\_GetNextHop() //** For getting the next hop device id.

**fn\_NetSim\_LEACH\_AssignClusterHead() //** For electing the Cluster head based on Remaining energy.

**fn\_NetSim\_LEACH\_IdentifyCluster() //** To determine the cluster to which a sensor belongs.

**Sinkhole Attack on LEACH:**

In this project we are implementing sinkhole attack on top of the LEACH project where a malicious node advertises false battery information to become a cluster head. Upon being elected as a cluster head, it attracts network traffic from all its cluster members and destroys the packets without forwarding them to the sink/base station.

**Implementation:**

A file **malicious.c** is added to the DSR project which contains the following functions:

* **fn\_NetSim\_DSR\_MaliciousNode()** This function is used to identify whether a current device is malicious or not in-order to establish malicious behavior.
* **fn\_NetSim\_DSR\_MaliciousProcessSourceRouteOption()** This function is used to drop the received packets if the device is malicious, instead of forwarding the packet to the next hop.

You can set any device as malicious and you can have more than one malicious node in a scenario. Device id’s of malicious nodes can be set inside the **fn\_NetSim\_DSR\_MaliciousNode()** function.

**Steps:**

1. To open-source code open **NetSim** > **Your Work** > **Source Code** > **Open code**
2. In **LEACH.c** present inside **DSR** Project, Number of clusters and Size of clusters is decided, and in **Malicious.c** the malicious node is set You can set them to any value and node

**Note:** By default, Malicious Node is set to 4 and NUMBER OF CLUSTERS – 4, SIZE OF CLUSTERS – 16, If changed Rebuild the Solution as shown below:

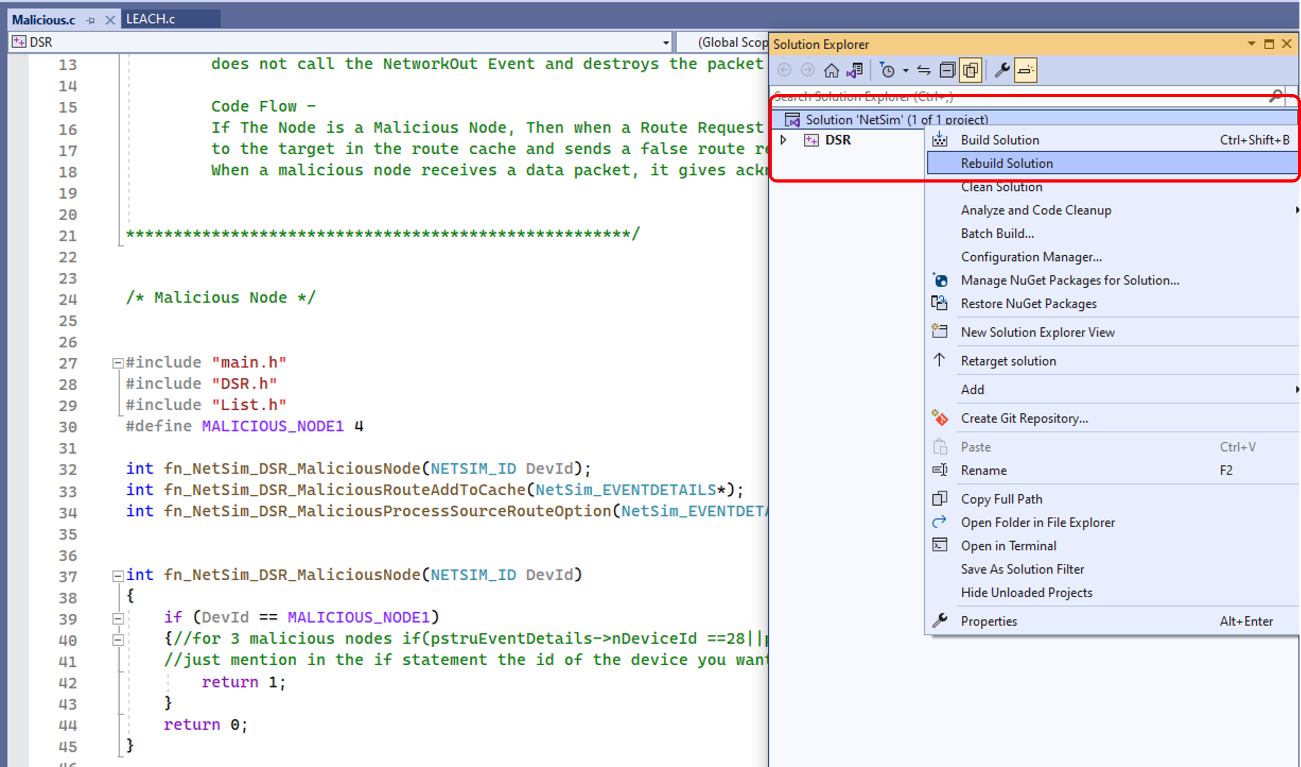


Figure : solution Rebuild in source code

1. Rebuild the solution by right clicking on solution > Rebuild solution.
2. The **Workspace\_Sinkhole\_in\_LEACH** comes with an inbuilt network scenario example, Open the scenario
3. Run the simulation

**Results and discussion:**

* View the packet animation. You will note that the sensors directly start transmitting packets without route establishment since the routes are statically defined in LEACH. You will also note that the cluster heads keep changing dynamically in Clusters 2, 3 and 4. In cluster1, the cluster members transmits packets to malicious node (device id 4) since it advertises false battery information to become a cluster head.
* This can be observed in Packet trace by applying filters to Source\_ID column by selecting only Sensor-1, 12, 20, 28. You will be able to see that the receiver id is sensor-4 throughout the simulation. All the nodes in the Cluster1 are sending data packets to malicious node (Sensor-4) since it is the Cluster Head

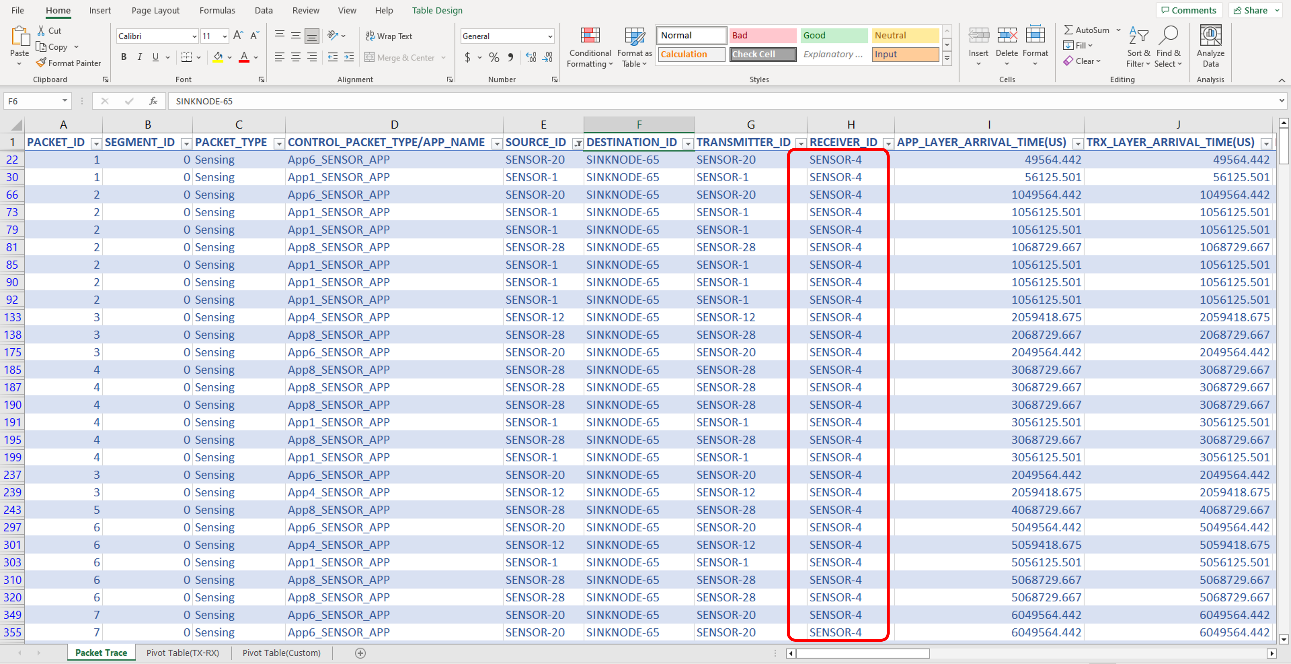
****

Figure : Packet trace file referring to malicious node reception of transmitted packets

* This will have a direct impact on the Application Throughput which can be observed in the Application Metrics table present in NetSim Simulation Results window. The throughput for applications 1, 4, 6 and 8 are zero since the source ids belongs to cluster1 having malicious node (device id 4)

**Table

Description automatically generated**

Figure : Simulation results window