Intrusion detection system on LEACH

Software Recommended: NetSim Standard v11.1 32-bit/ 64-bit, Visual Studio 2017/2019

Follow the instructions specified in the following link to clone/download the project folder from GitHub using Visual Studio:
https://tetcos.freshdesk.com/support/solutions/articles/14000099351-how-to-clone-netsim-file-exchange-project-repositories-from-github-

Other tools such as GitHub Desktop, SVN Client, Sourcetree, Git from the command line, or any client you like to clone the Git repository.

Note: It is recommended not to download the project as an archive (compressed zip) to avoid incompatibility while importing workspaces into NetSim.

Secure URL for the GitHub repository:

https://github.com/NetSim-TETCOS/IDS_in_LEACH_v11.1.git

The following steps show how a user can run the IDS in NetSim to detect a malicious node, and then setup a new route to the destination avoiding the malicious node

- Creating Malicious nodes for a particular network scenario is explained in Malicious.c file
- Clustering and cluster head election is explained in LEACH.c file
- To detect the intruder and to send data via a new route, the following files are added in DSR and Zigbee:
  - **Pathrater.c** :

    This file contains code for avoiding the malicious node and finding a new route (once the IDS detects the malicious node) in networks running DSR in Layer 3. Note that this system would work only for UDP and not for TCP, since TCP involves receiving ack's from the destination

    If _NETSIM_PATHRATER_ is defined, the code is used to validate routes. When the Node is a Malicious Node and a Route Reply is processed, the Function verifies the route reply in the route cache and checks for the black listed node i.e.,malicious node. When a malicious node is found, that route entry is deleted from
    the cache.

- **Watchdog.c**

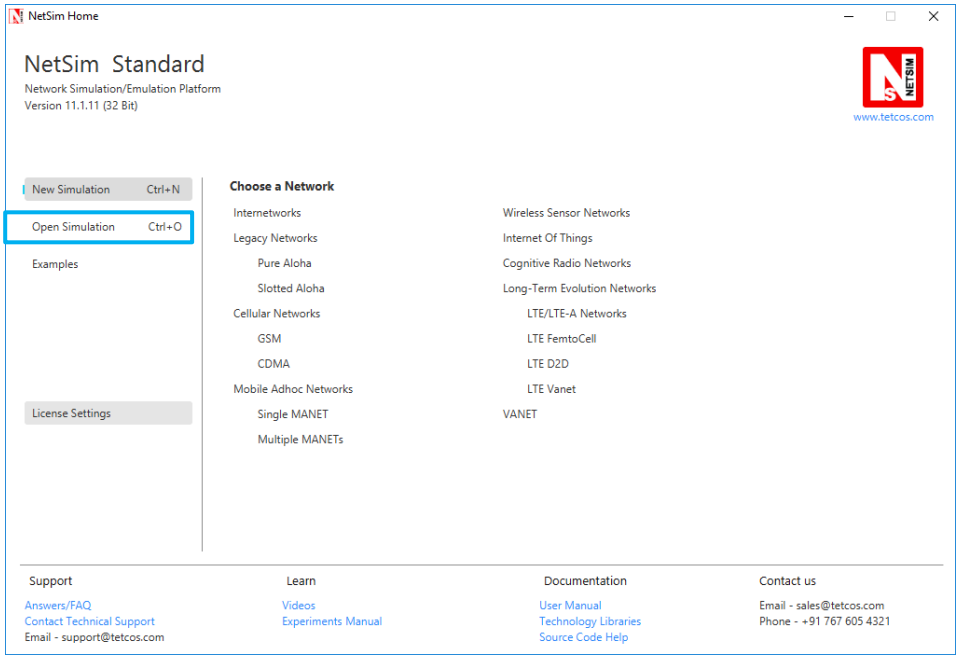
  This file contains code for the IDS and is added in Zigbee operating in Layer 3.

  If _NETSIM_WATCHDOG_ is defined, a watchdog timer starts the moment a packet is sent. Once a packet is forwarded to next hop node, the current node checks for watchdog timer duration if the packet is getting forwarded further on to destination node or not.

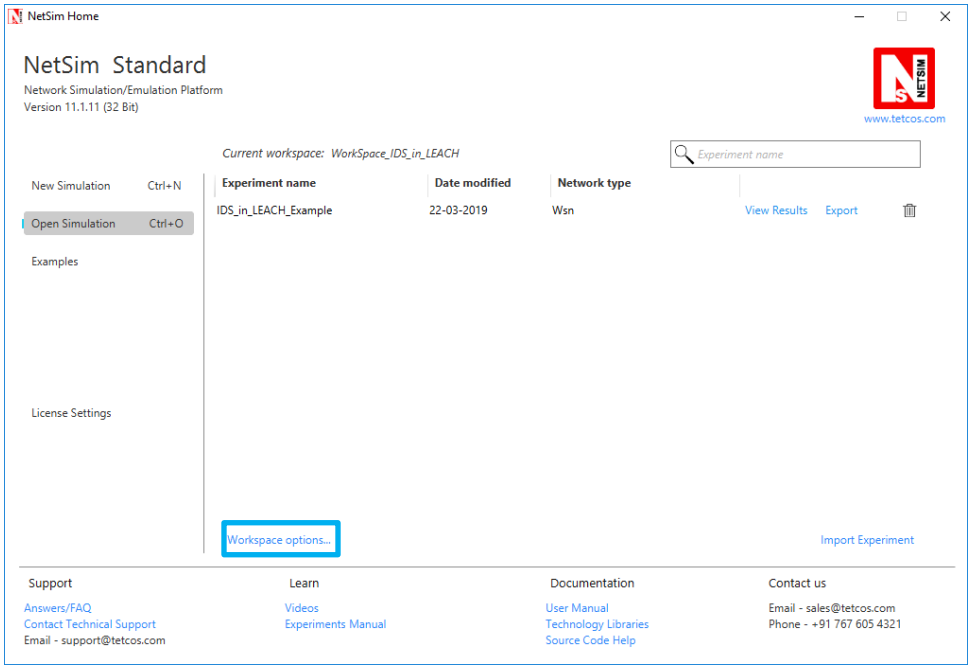
  The malicious node doesn't forward packets that it receives. The watchdog timer in the node (which forwarded the packet to the malicious node) expires. A counter is present which measures the number of times the watchdog timer expires (in other words the number of packets sent out but not forwarded by the next hop node). Once this counter's value reaches the failure threshold the next hope is marked by the current node as a malicious node.

Steps:

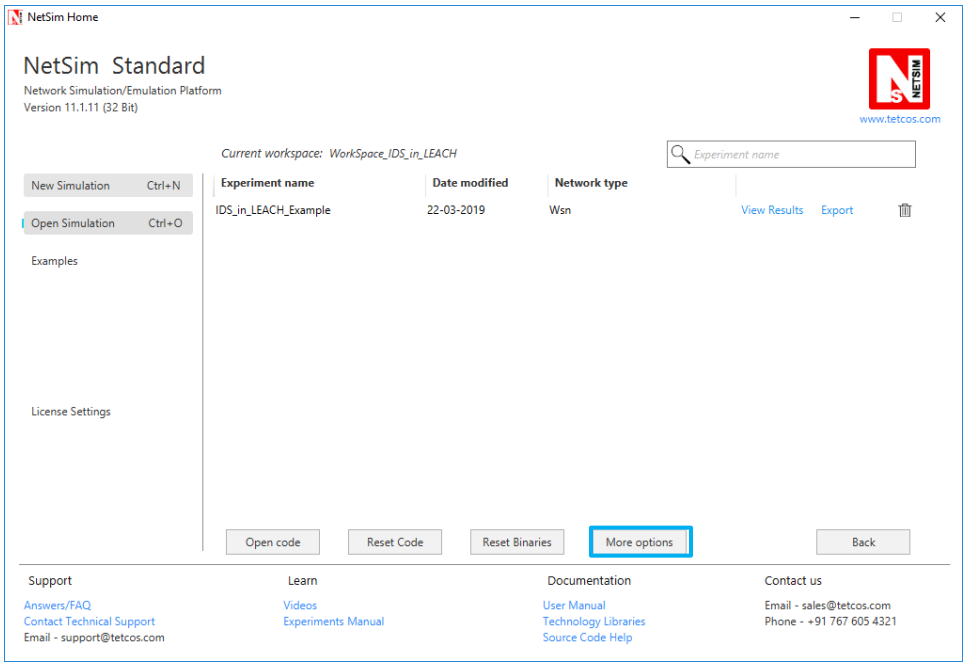
- After downloading the project folder using the GitHub URL, Open NetSim Home Page click on **Open Simulation** option,



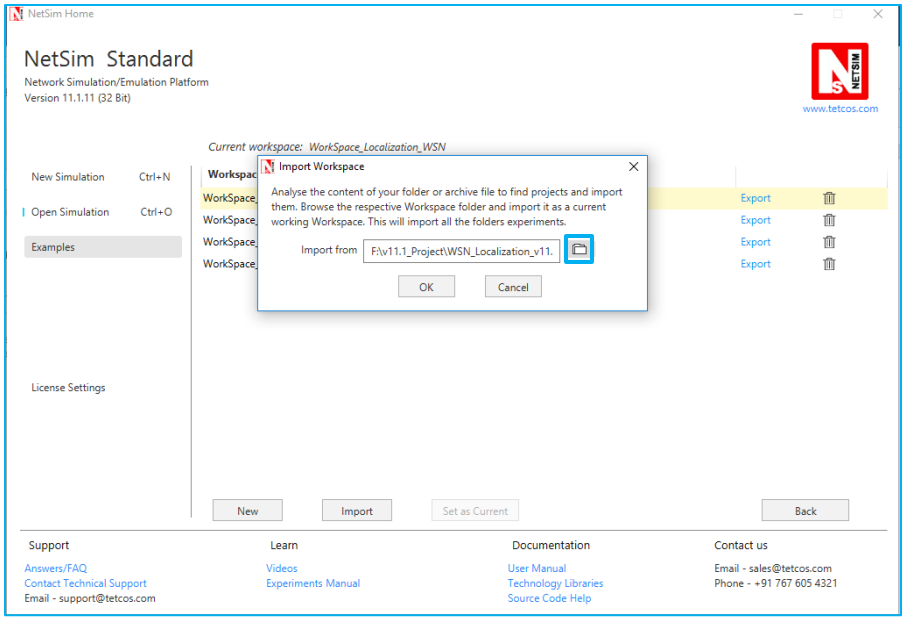
- Click on **Workspace options**



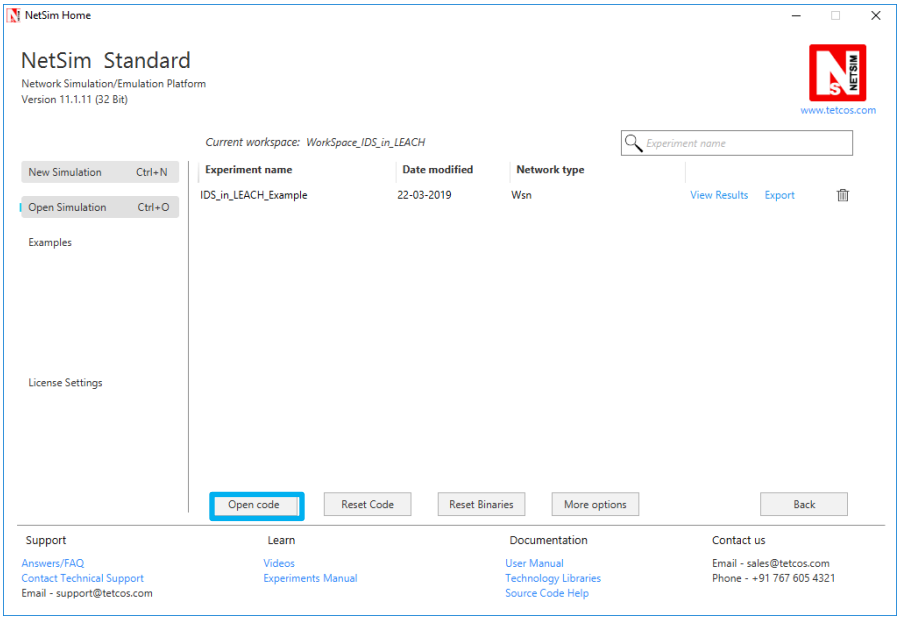
- Click on **More Options,**



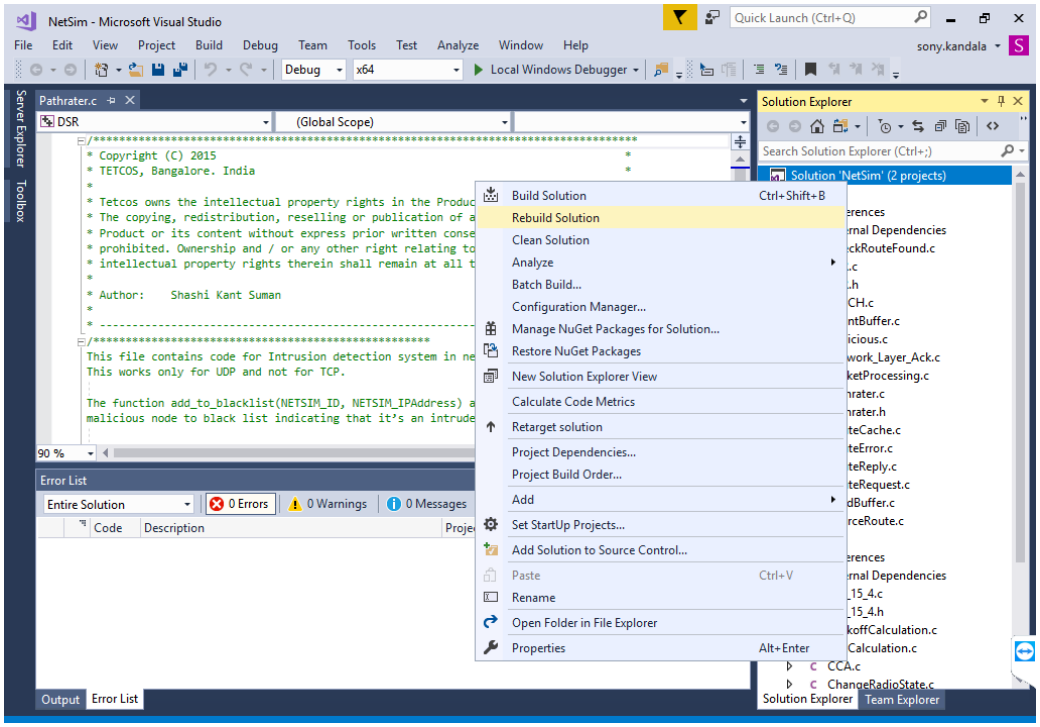
- Click on **Import**, browse the extracted folder path, and go into the WorkSpace_IDS_in_LEACH directory, click on Select folder button and then on OK button.



- Go to home page, Click on **Open Simulation → Workspace options → Open code**



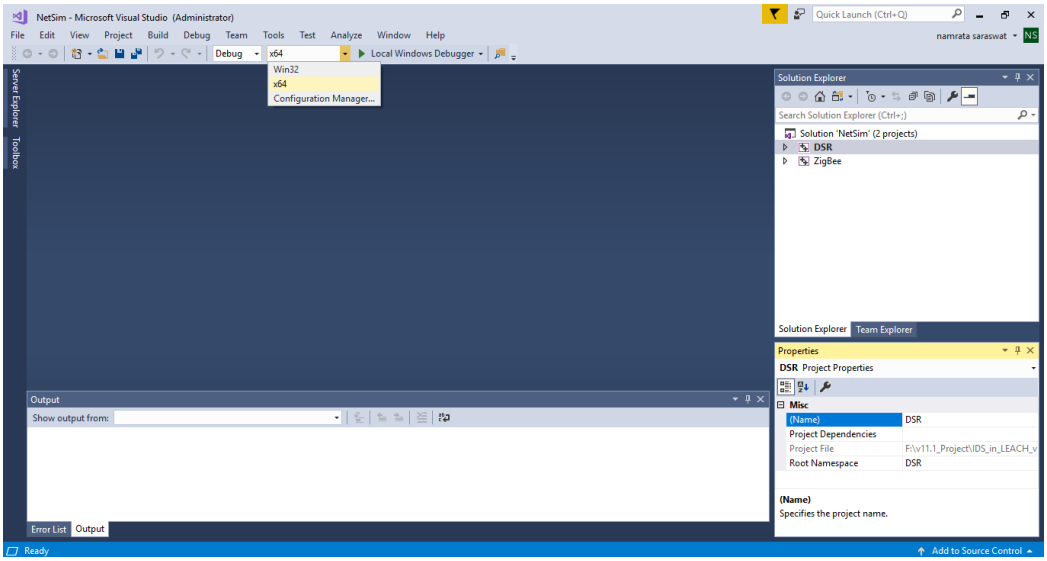
- Right click on the solution and select rebuild.



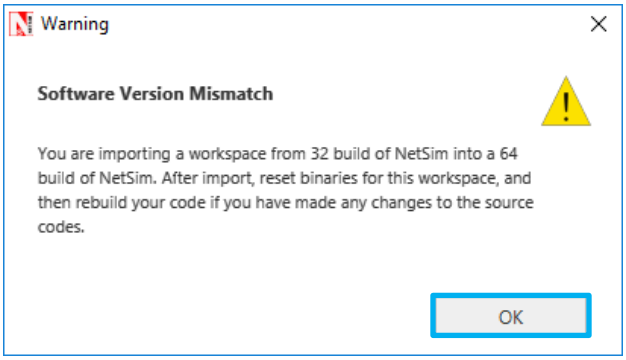
- Upon rebuilding, **libZigbee.dll** and **libDSR.dll** will automatically get updated in the respective binary folder of the current workspace.

Note:

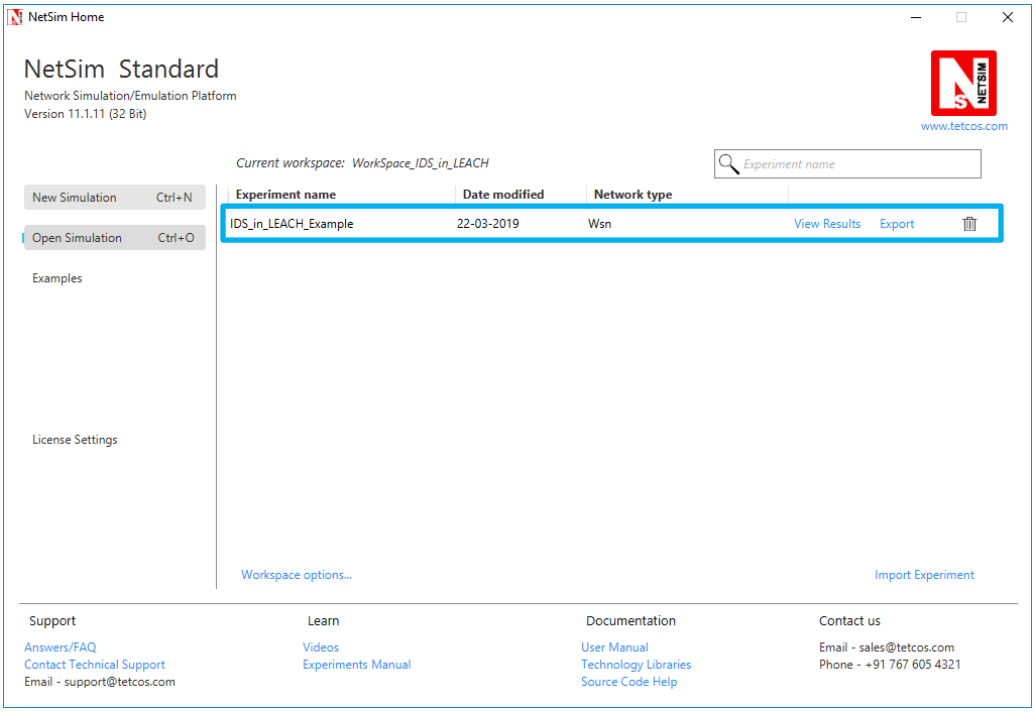
1. Based on whether you are using NetSim 32 bit or 64 bit setup you can configure Visual studio to build 32 bit or 64 bit Dll files respectively as shown below:

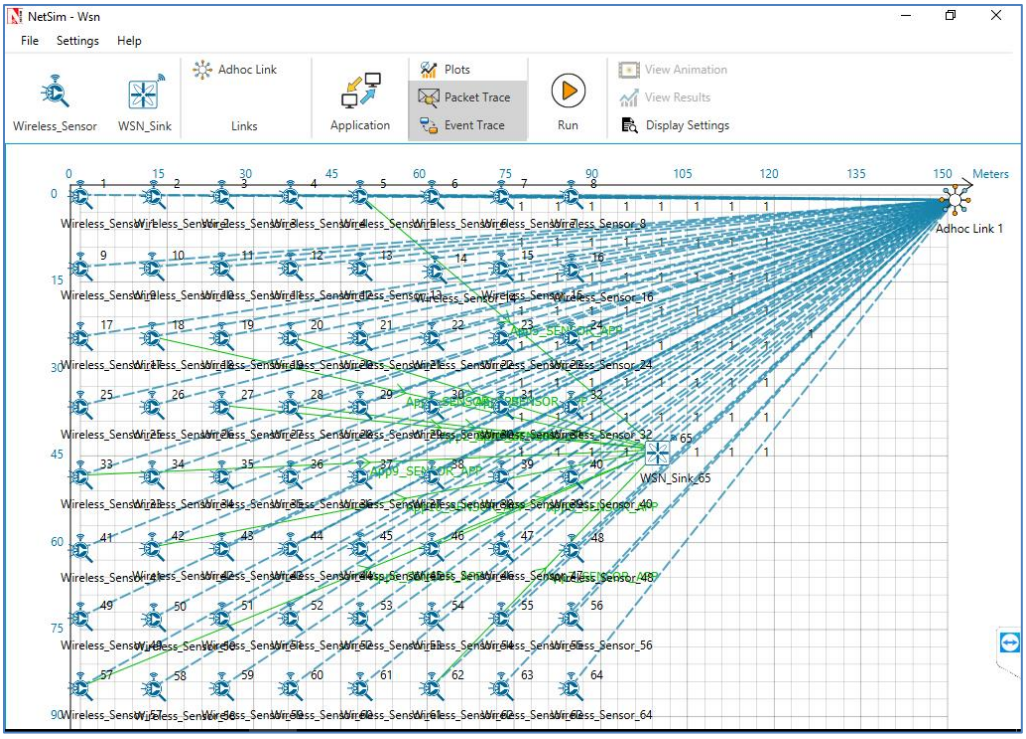


2. While importing the workspace, if the following warning message indicating Software Version Mismatch is displayed, you can ignore it and proceed.



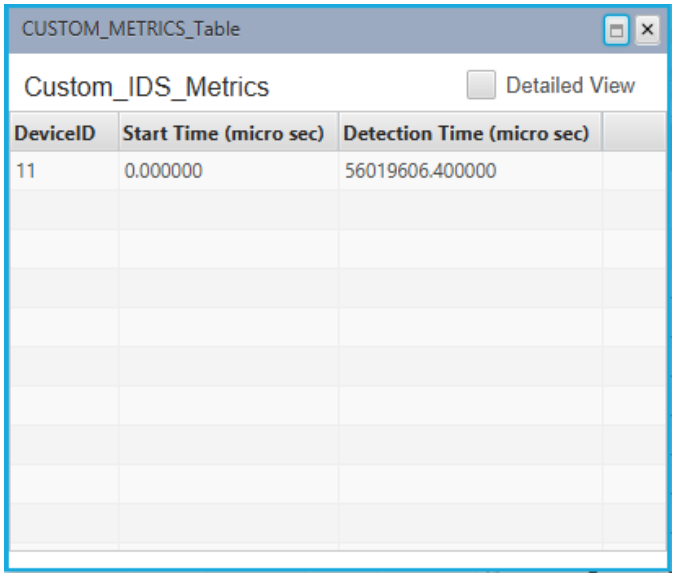
- Go to NetSim home page, click on **Open Simulation**, Click on **IDS_in_LEACH_Example**.





- This Network is created in WSN Network as per the Number of clusters and size of clusters that are set in the LEACH code. By default the code runs for a scenario with 64 sensors uniformly placed, with the SINKNODE placed as per the screenshot above
- Channel Characteristics is set to Pathloss only with LOG_DISTANCE as the path loss model. Path loss exponent is set to a high value 4
- Run the simulation
- 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, initially the cluster members transmits packets to malicious node (device id 11) since it advertises false battery information to become a cluster head. Per the original code setting the Watchdog timer is set to 2 seconds and the failure threshold is set to 20 packets. So you would notice that around 55 seconds, the malicious node is detected and then cluster head is elected dynamically based on the remaining energy of the sensor

- This can be observed in Packet trace by applying filters to Source_ID column by selecting only Sensor-18, 20, 27 and 28. You will be able to see that the receiver id is sensor-11 till 55s of simulation time and then it is changed
- Now undo filter in Source_Id column and apply filter to transmitter_Id column by selecting only Sensor-11. You will be able to see that no data packets are forwarded by the malicious node
- 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, 2, 3 and 4 are less since the source ids belongs to cluster1 having malicious node (device id 11).
- The time at which a malicious node is detected can be obtained from the CUSTOM METRICS in the results window where the start time - time from which a node becomes malicious, detection time - time at which the node was added to blacklist can be obtained.



Dedicated Metrics for IDS