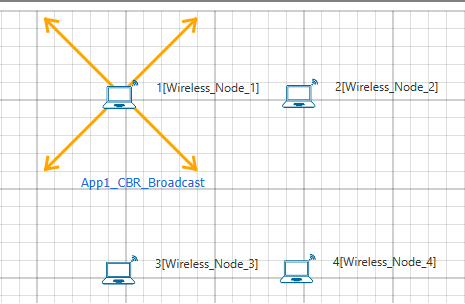
**Rebroadcasting packet in NetSim MANET\VANETs**

**Software Used:** NetSim Standard v11.0 (32bit), Microsoft Visual Studio 2017

**Broadcasting:**

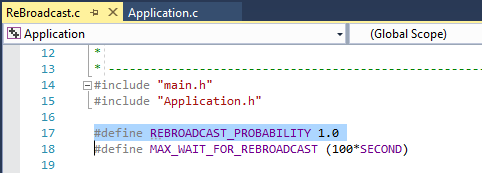
Broadcasting is the process of sending a message from one node to all other nodes in an ad-hoc network. It is a fundamental operation for communication in ad-hoc networks as it allows for the update of network information and route discovery at every node.

**Rebroadcasting:**



Wireless Node 1 initiates a broadcast message and the message is received by nodes 2, 3 and 4. 2, 3 and 4 rebroadcast the message if they have not broadcasted that before. Furthermore, this implementation involves a Rebroadcast\_Probability based on which the nodes resend the packets.

**Probability-based rebroadcasting** - The decision of rebroadcasting is based upon a random probability. This probability may be as simple as flipping a coin or it may be very complex involving probabilities which include parameters such as node density, duplicate packets received, battery power or a particular nodes participation within the network etc. Users can change the Rebroadcast\_Probability macros present in Rebroadcast.c file as shown below:



**Rebroadcasting in NetSim:**

To implement this project in NetSim, we have created an additional Rebroadcast.c file inside Application project. The file contains the following functions:

* void rebroadcast\_packet();

This function is used to rebroadcast the packet.

* static bool isRebroadcastAllowed();

This function is used to check whether rebroadcasting is allowed or not.

* void rebroadcast\_add\_packet\_to\_info();

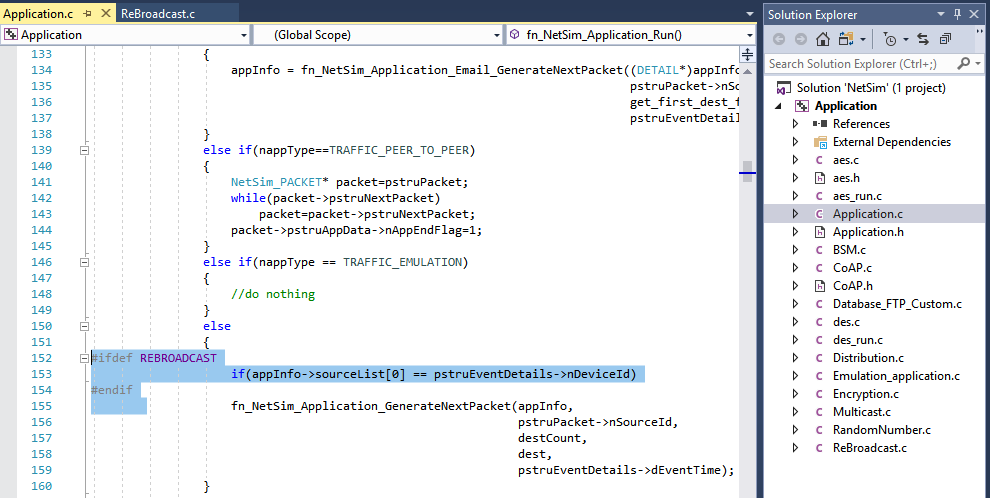
This function is used to add the packet to rebroadcast list.

* static void cleanup\_broadcast\_info();

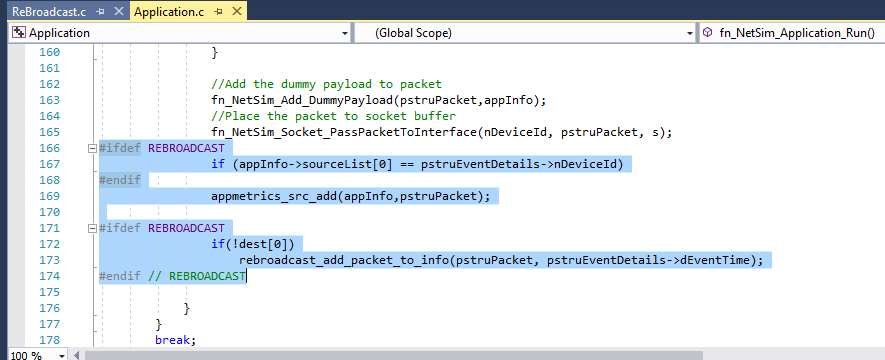
This function is used to clean the broadcast information.

**Code modifications done in NetSim:**

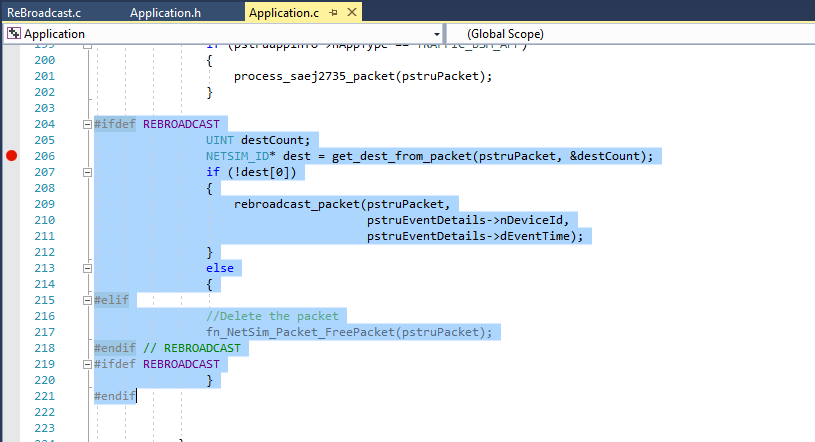
1. We have added the following lines of code in fn\_NetSim\_Application\_Run() function in the APPLICATION\_OUT\_EVENT present in Application.c file inside Application project. This is used to generate next broadcast packet if the current device is present in the source list.

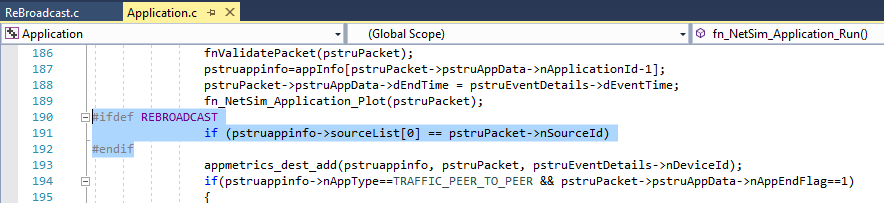


1. The following lines of code are added in the same fn\_NetSim\_Application\_Run() function in the APPLICATION\_OUT\_EVENT present in Application.c file inside Application project. The code checks if the destination is ‘0’ i.e., Broadcast packet, then it adds the packet to rebroadcast list.

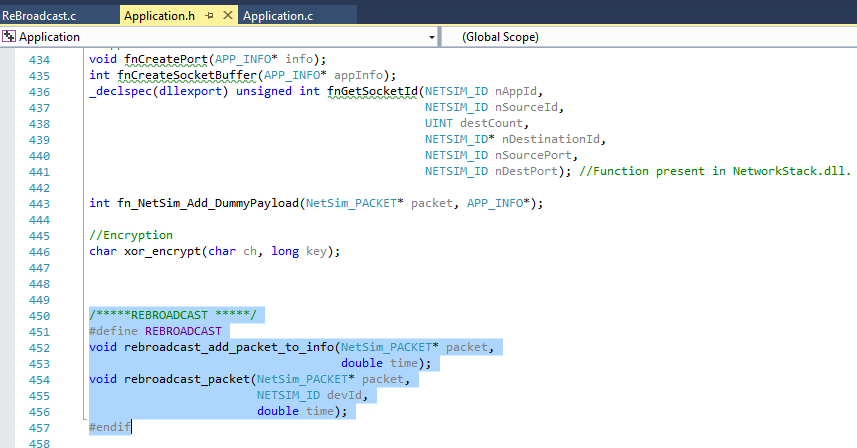


1. Now add the following code in fn\_NetSim\_Application\_Run() function in APPLICATION\_IN\_EVENT present in Application.c file inside Application project. It checks whether the destination is ‘0’ or not. If it is ‘0’, then it rebroadcasts the packet or else deletes the packet.



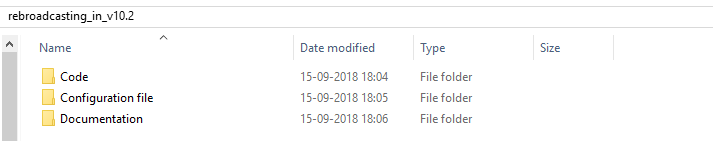


1. We have added the following function declarations in Application.h file.

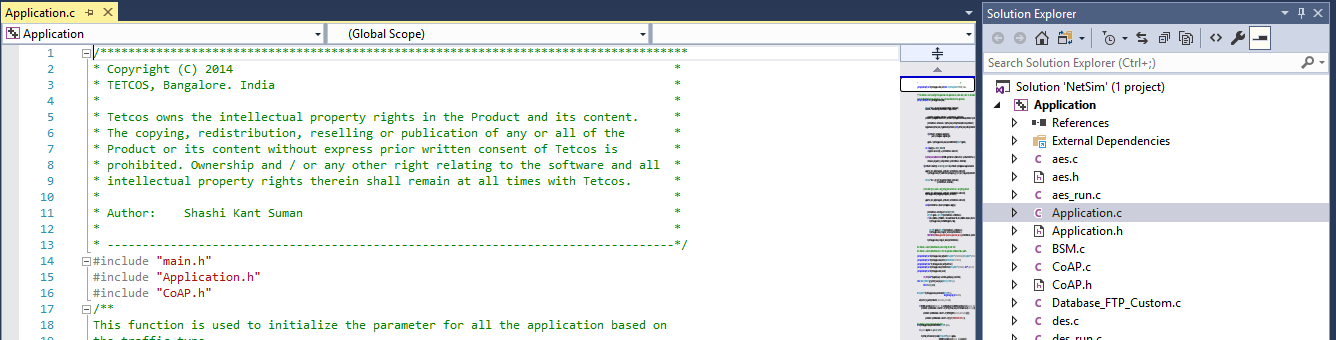


**Steps:**

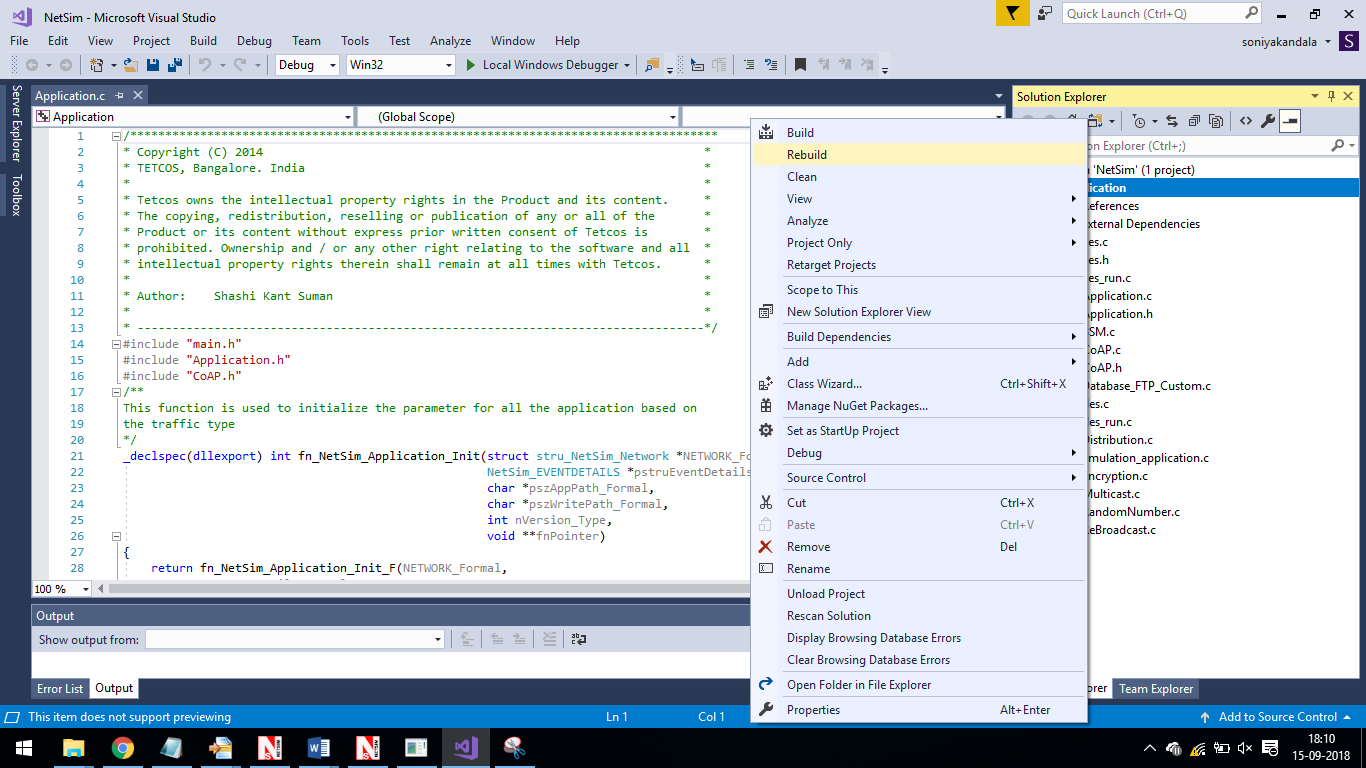
* After you unzip the file, the folder would look like:



* Open Code folder and double click on the NetSim.sln file present to open the project in Visual Studio 2017.

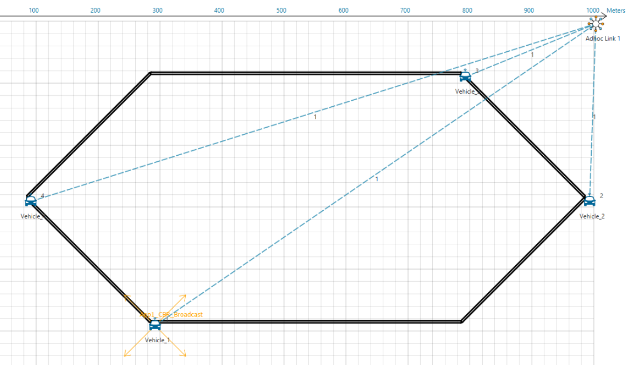


* Right click on Solution in Solution Explorer and select ‘Rebuild solution’.

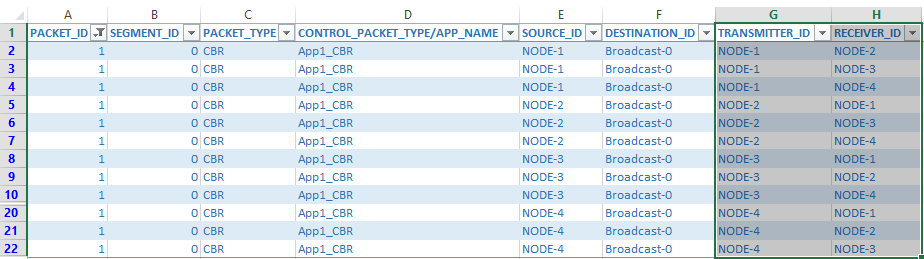


* Upon rebuilding, **libApplication.dll** will get created in the DLL folder.
* Now copy the **libApplication.dll** from the DLL folder and paste it in NetSim bin folder present in the NetSim installation directory. The NetSim install directory would look something like < C:\Program Files\NetSim Standard\bin>.
* Note that there already exists **libApplication.dll** in this bin folder. This is the default file being shipped with NetSim. The user has to replace this file with his newly built file.
* Therefore, take care to rename the original **libApplication.dll** file before replacing it, so that it is backed up. For example, you may rename it as **libApplication\_default.dll**.
* Run NetSim and open **Configuration.netsim** file present inside the Configuration file/**VANET or MANET** folder and run the simulation for 100 seconds.

**VANET SCENARIO:**



* In the above scenario, Vehice-1 is broadcasting the packet and it is received by the Vehicles 2, 3 and 4. Then Vehices 2, 3, and 4 will rebroadcast the same packet based on the probability value in Rebroadcast.c file.
* After simulation, open Packet Trace and filter Packet\_Id to ‘1’ or any other id and observe that the nodes other than source are rebroadcasting the same packet.



* Note that Users SHOULD NOT use the performance metrics provided at the end of simulation but should rather calculate the network performance metrics from the packet trace.
* Users can also create their own network scenarios in **Single MANET/VANET** and run the simulation.