AES

Advance Encryption Standard (AES) is a symmetric key algorithm used for encrypted data transmission. As any symmetric key algorithm, it relies on both the sending node and receiving node both have the same, secret key to encrypt and decrypt. It is a block cipher algorithm, meaning the data is encrypted and decrypted in a set size of data. Here’s a good explanation of the history and operation of the algorithm presented in an easy and interesting format:

[Stick Figure Guide to AES](http://www.moserware.com/2009/09/stick-figure-guide-to-advanced.html)

**Lab Exercise**

In this lab simply run the solution code and look at the output. To run, you will need two Teensy Loggers connected on a CAN bus. When both sketches are running, type any string into the serial input on the serial monitor. This message will be encrypted and sent across the CAN bus where it is decrypted. Both nodes can Send and Receive (Note: The first sent messages sometime fail, send from both nodes first and then subsequent inputs are successfully sent and received)

Attempt to understand the Code and answer consider the following questions:

* What difficulties resulted from only being able to send 8 bytes in one message?
* How many messages does it take to send a small message, such as “Hi”?
* Why might AES be a poor solution over CAN buses?
* Where is the secure key located? Is this key secure?
* What algorithm/process could help establish a more secure Key before sending a message?
* Is this code well documented? Organized? How could it be improved for easier reading and understanding?

**NOTE**: In general, it is typically a poor security decision to implement cryptographic algorithms that are not widely reviewed and evaluated. The libraries used in this lab have little verification and are not endorsed by reputable cryptographic organizations. This is for demonstration and learning purposes only.