TPMS

SS

3/16/2017

Odyssey

From http://opengarages.org/handbook/ebook/, TPMS data can be exploted in the following ways:

• Send an impossible condition to the engine control unit (ECU), causing a fault that could then be exploited • Trick the ECU into overcorrecting for spoofed road conditions • Put the TPMS receiver or the ECU into an unrecoverable state that might cause a driver to pull over to check for a reported flat or that might even shut down the vehicle • Track a vehicle based on the TPMS unique IDs • Spoof the TPMS signal to set off internal alarms

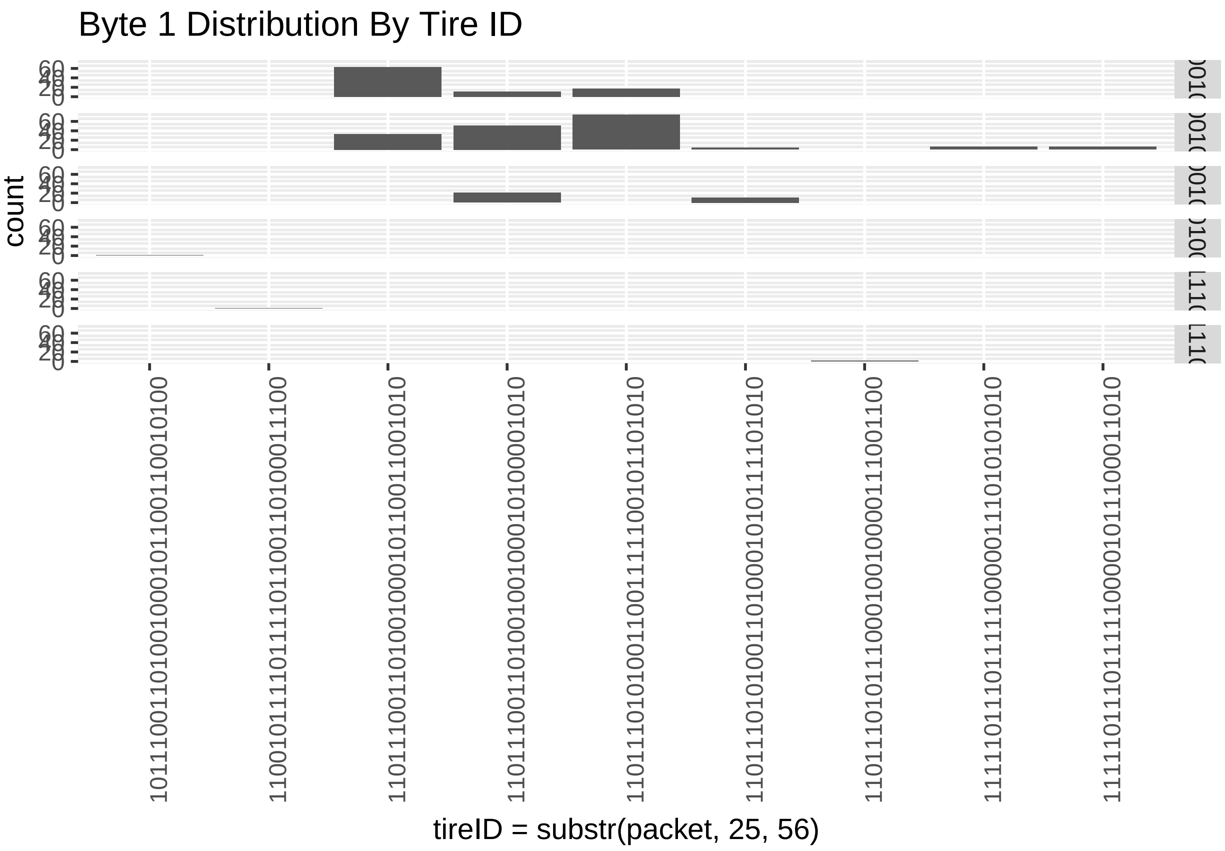
This project uses the code at https://github.com/jboone/gr-tpms to capture TPMS data. The author's talk, using an earlier version of the code, can be found here: http://www.youtube.com/watch?v=bKqiq2Y43Wg. Previous research on the topic can be found at https://web.wpi.edu/Pubs/E-project/Available/E-project-091115-154458/unrestricted/MQP_piscitelli_arnold_2015.pdf, and security vulnerabilities discussed in more depth at http://www.winlab.rutgers.edu/~gruteser/papers/xu_tpms10.pdf/.

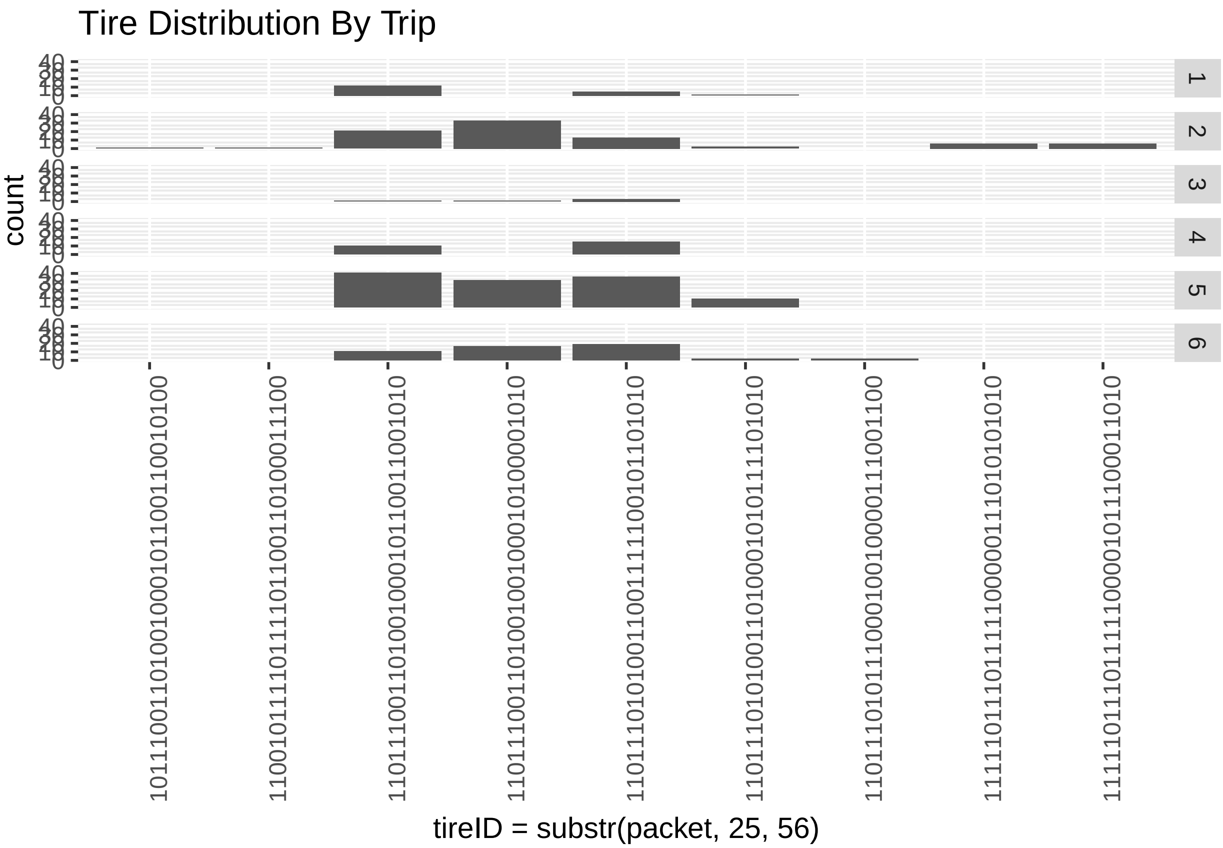
Distribution of tire IDs

```
##
## 10111001101001000101100110010100 11001011110111101100110100011100
##                                1                                1
## 11011100110100100010110011001010 11011100110100100100010100001010
##                               96                               83
## 11011101010011001111100101101010 11011101010011010001010111101010
##                               91                               15
## 11011101011100010010000111001100 11111011101111110000011101010101010
##                                2                                6
## 11111011101111110000101110001101 0
##                                6
```

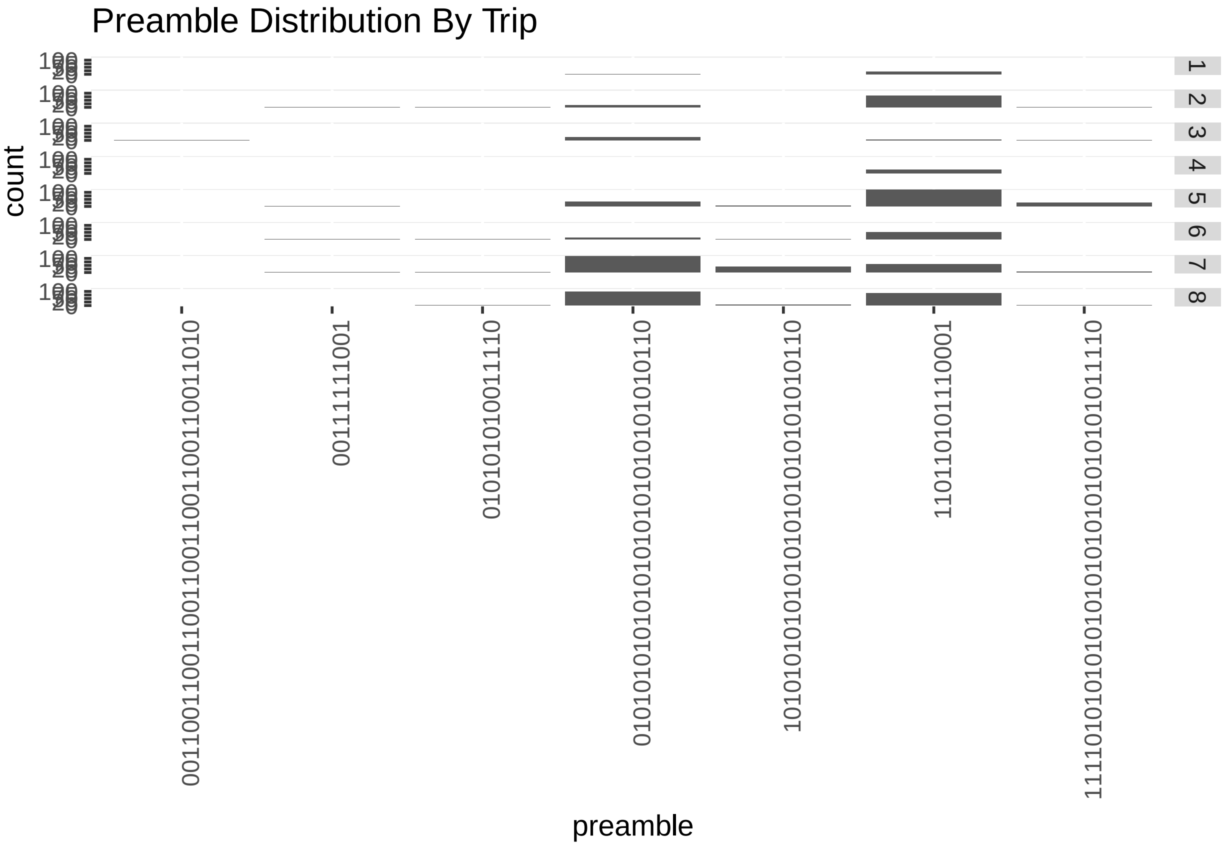
First 3 bytes, statistical distribution

```
## Byte 1:
## byte
## 00001001 00001010 00001011 01010011 11111010 11111011
##       91      174       32        1        1        2
## Byte 2:
## byte
## 00000011 00000100 00010011 00010100 00100010 00100011 00110011 00110100
##        8       19       14        2        2        6       22        1
## 01000010 01000011 01010010 01010011 01010100 01100011 01100100 01100101
##        5        4        4       25        1       18        7        1
## 01110010 01110011 01110100 10000010 10000011 10010011 10100010 10100011
##        4        6        1        5        7       25        2       15
## 10110010 10110011 11000010 11000011 11000100 11010011 11100010 11100011
##        3       11        4       18        1       15        1       28
## 11110010 11110011
##       10        6
## Byte 3:
## byte
## 00001000 00011000 00011100 00101000 00101101 00111000 01001000 01011000
##       28       24        1       19        2       19        6       14
## 01011101 01101000 01111000 10001000 10011000 10101000 10101101 10110001
##        6        9       12       14        9       31        6        1
## 10111000 11001000 11011000 11101000 11111000
##       43       26       15        4       12
```

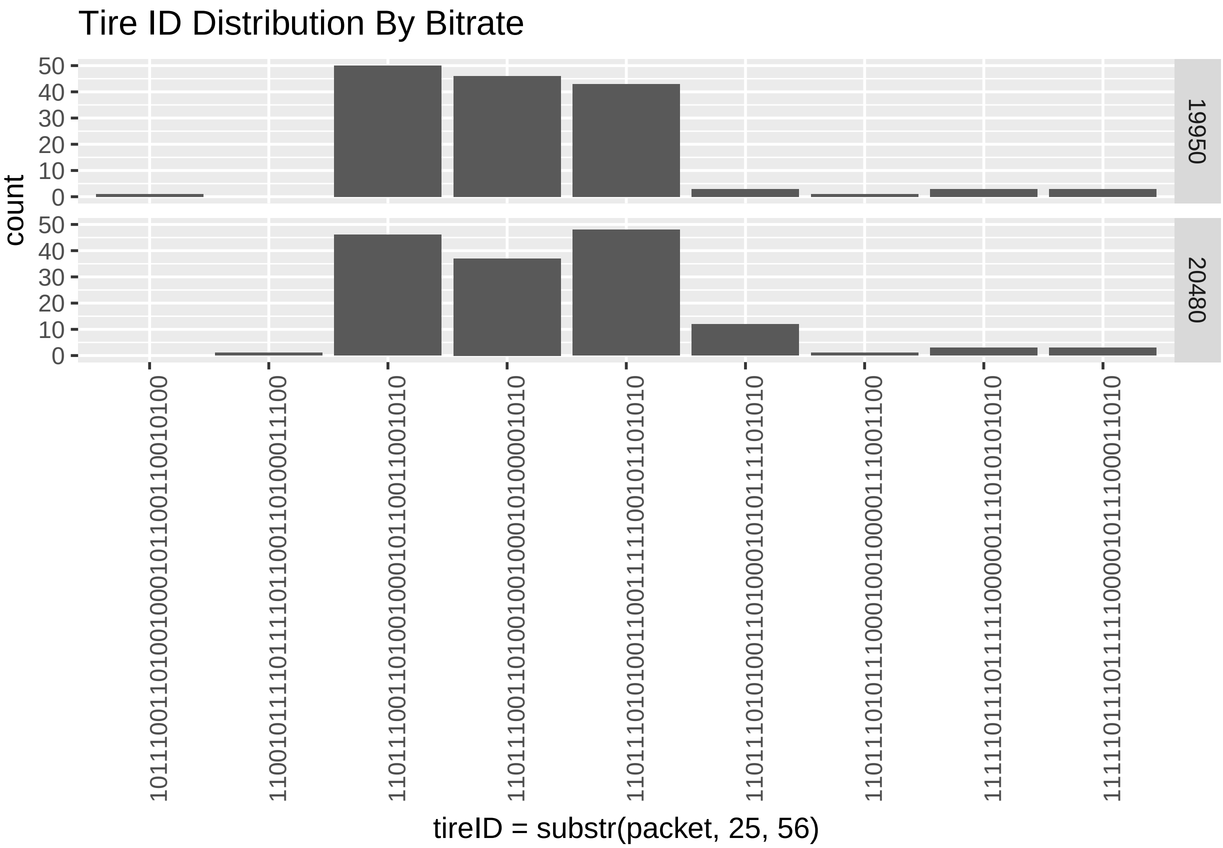




Tire Distribution By Trip



Preamble Distribution By Trip


Tire ID Distribution By Bitrate
count
0
10
20
30
40
50
19950
20480
10111001101001000101100110010100
11001011110111101100110100011100
11011100110100100010110011001010
11011100110100100100010100001010
11011101010011001111100101101010
11011101010011010001010111101010
11011101011100010010000111001100
11111011101111100000111010101010
11111011101111100001011100011010
tireID = substr(packet, 25, 56)