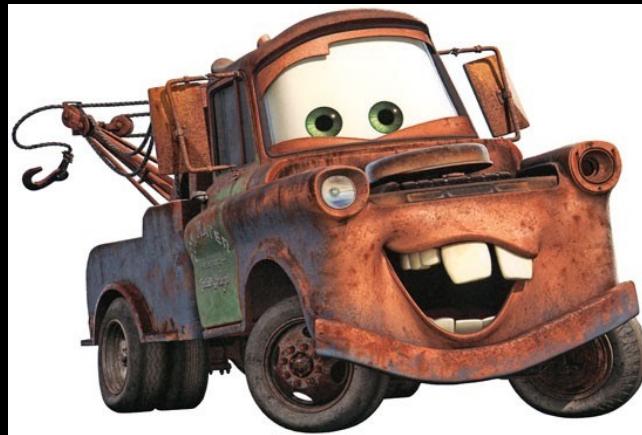


Hackeando o seu carro com GNU/Linux, OBD2 e python

Sergio Cioban Filho - <https://br.linkedin.com/in/cioban>
cioban@virtmasters.com - twitter: @cioban



Agenda

- cat /home/cioban/ABOUTME
- O OBD-II
- O básico sobre Linux embarcado
- Mas, python!?!?
- O que eu consigo coletar?
- Juntando tudo...
- O obdpy
- Resultados e possibilidades
- Conclusão
- Referências

cat /home/cioban/ABOUTME

- Certificado RHCE, RHCVA, LPIC3
- Pós-graduando em Redes e Segurança de sistemas pela PUC-PR
- Técnico em eletrônica pelo CEFET-SC
- Trabalha com linux desde 2003
- Um misto de Sysadmin e Developer
 - Tá bom vai, pode me chamar de nerd... :)
- Tentando alavancar a [@virtmasters ~] #
- Mais info em: <https://br.linkedin.com/in/cioban>

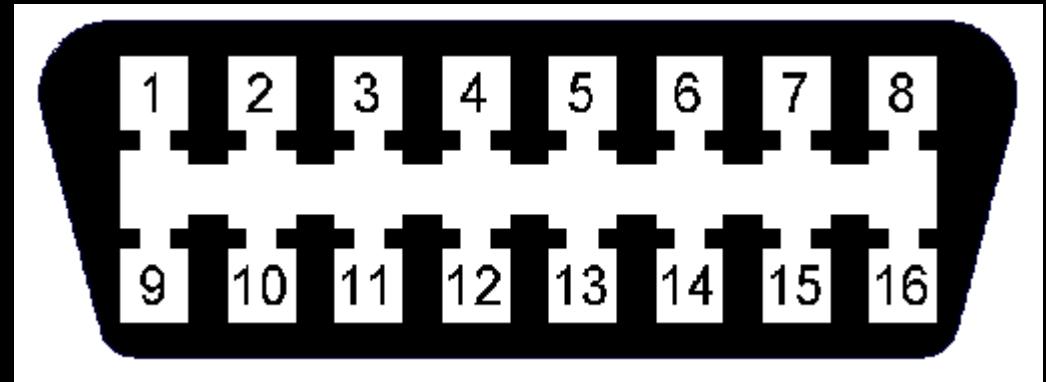
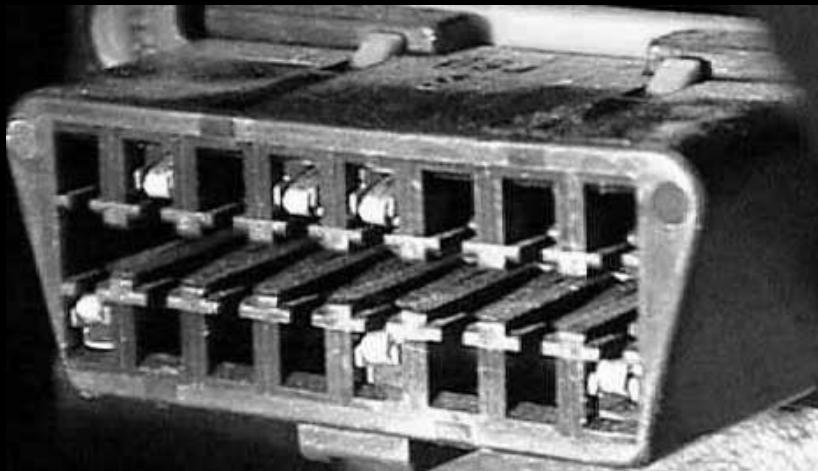
import obd2

O OBD-2

- Do inglês On-Board Diagnostic
- Sistema padronizado de diagnóstico de veículos
 - Conector padronizado
 - Protocolos abertos e padronizados (nem tanto)
- Coleta informações da ECU / PCM
 - Engine control unit / powertrain control module
- Depende dos sensores disponíveis no carro
- Desde 1996 é obrigatório nos EUA
- Desde 2001 é obrigatório na Europa
- Desde 2010 é obrigatório no Brasil

O OBD-2

- Diversos protocolos documentados
- Se divide em modos de operação e PIDs
 - <MODO><PID> - ex.: 010D - Speed
- Pode ler informações da ECU
- Pode alterar(escrever) informações da ECU
- Fabricantes limitam o acesso para não danificar o motor
- Fabricantes alteram constantemente PIDs e informações.



import linux

O básico sobre Linux embarcado

- Propósito específico
- Restrição de tamanho físico
- Restrição de consumo de energia
- Restrição a utilização de RAM, ROM, FLASH
- Bootloader, kernel, rootfs
 - Ex.: celulares, routers, tablets, etc.
 - Ex.: Computador de bordo esquisito do palestrante nerd... :)

import python

Mas, python!?!?

- Rápido desenvolvimento
- Rápida prototipação
- Módulos
- Em C teria que compilar toda hora, usando um compilador na placa ou crosscompile
- Em C teríamos que se preocupar com, malloc, memória, ponteiros, etc, etc.
- E o overhead?
- Ainda pode se usar C como módulo

O que eu consigo coletar?

- Resposta rápida: Depende
- As informações disponíveis via OBD-2, variam de acordo com o fabricante e modelo.
- Basicamente, a ECU precisa de um conjunto mínimo de sensores para o correto funcionamento do carro (speed, RPM, MAF, etc).
 - A ECU gasta a maior parte do seu processamento regulando a mistura correta entre ar e combustível 14.7:1
- Existe uma série de informações que os fabricantes não são obrigados a disponibilizar.
- Mas se você pagar, eles disponibilizam.

Juntando tudo... - Premissas

- Conseguir coletar informações do carro
- Calcular métricas com os dados obtidos
- Mostrar informações em tempo real
- Armazenar as informações para relatórios
- Integrar com outras tecnologias (TODO)
 - Sensores, GPS, 3G
- Informações ao mundo externo (TODO)
 - Displays, etc

Juntando tudo... - Materiais

- Placa com Linux Embarcado (Utilizei uma Raspberry Pi)
- Dongle USB Bluetooth (\$1.58 SKU: 11866)
- OBD-2 interface – bluetooth (\$8.86 SKU: 142679)
- LCD pcd8544 - padrão Nokia 5110 - (\$3.60 SKU: 145860)
- Teclado de matriz (\$2.19 SKU: 117718)
- RTC i2c
- Caixa plástica, cabos, fonte, placa perfurada, etc
- Python
- obdpy

Juntando tudo... - Materiais



Juntando tudo... - Como coletar?

- O ELM327 disponibiliza um set de comandos AT
- Para informações iniciais, não precisamos aprender nenhum protocolo
- Basta se conectar via serial bluetooth e trocar comandos AT
- Coletar as respostas via serial e calcular os valores
- O ELM327 não conseguiu fazer comunicação serial assíncrona

SEND: 0 1 0 0

RECV: ['0100', '41 00 BE 1F B8 13 ']

O obdpy

- Para colocar tudo pra funcionar, criei o projeto obdpy
- Escrito em python
- “Drivers”: LCD, keypad, obd, menu
- Coleta dados via serial
- Trata os dados e calcula os valores
- Mostra informações no Display
- Grava no SD (Utilizando o pickle)
- Gera relatórios dos dados coletados

Resultados e possibilidades



Resultados e possibilidades



Resultados e possibilidades

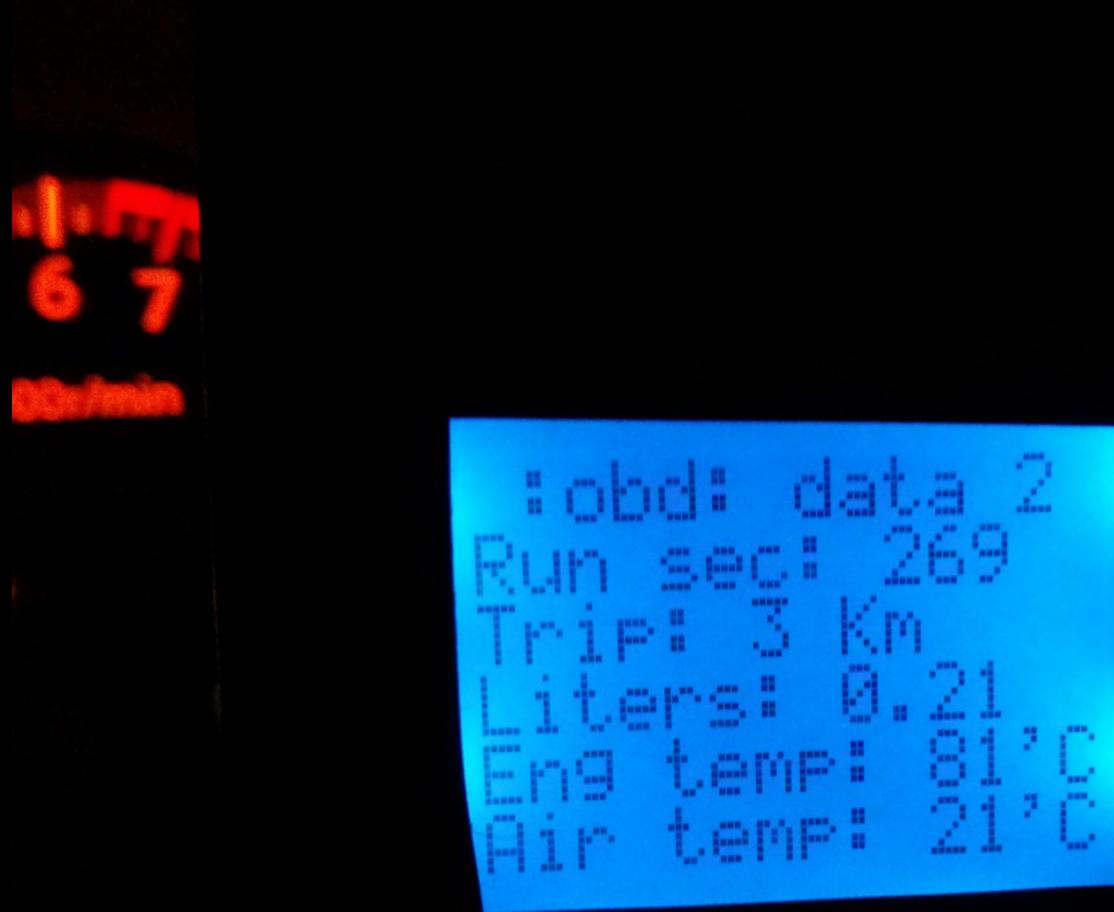


Resultados e possibilidades



```
:obd: data 1
Spd: 61 Km/h
RPM: 1598 25%
2 44.21 Km/L <
Avg 12.19Km/L
*Avg 11.85Km/L
```

Resultados e possibilidades



Resultados e possibilidades



Resultados e possibilidades



Resultados e possibilidades



Resultados e possibilidades



Resultados e possibilidades

```
cioban@asus:~/Dropbox/sandbox/python/obd/obdpy
root@mx01:~ x cioban@asus:/i x root@mx01:/e x cioban@asus:/ x cioban@asus:~ x cioban@asus:~ x
cioban@asus:~/Dropbox/sandbox/python/obd/obdpy 77x35

#####
# OBDPY DATA INFO
#####

    ==> PID DATA
[0x04] Calculated engine load value
    * MIN[10 %] MAX[100 %] AVG[53.870 %]
[0x05] Engine coolant temperature
    * MIN[80 °C] MAX[93 °C] AVG[83.531 °C]
[0x06] Short term fuel % trim-Bank 1
    * MIN[-15 %] MAX[13 %] AVG[-0.797 %]
[0x07] Long term fuel % trim-Bank 1
    * MIN[-21 %] MAX[-19 %] AVG[-20.162 %]
[0x0C] Engine RPM
    * MIN[532 RPM] MAX[4886 RPM] AVG[2448.705 RPM]
[0x0D] Vehicle speed
    * MIN[0 Km/h] MAX[154 Km/h] AVG[90.930 Km/h]
[0x10] MAF air flow rate
    * MIN[1 grams/sec] MAX[82 grams/sec] AVG[21.080 grams/sec]
[0x11] Throttle position
    * MIN[10 %] MAX[79 %] AVG[21.140 %]
[0x31] Distance traveled since codes cleared
    * MIN[4603 Km] MAX[4788 Km] AVG[4688.541 Km]
[0x46] Ambient air temperature
    * MIN[23 °C] MAX[32 °C] AVG[25.389 °C]
[0x1F] Run time since engine start
    * MIN[53 sec] MAX[7352 sec] AVG[3540.669 sec]

    ==> INFO
Fuel consumption average (using MAF and Speed): 12.506 Km/L
Distance traveled: 185 Km
Fuel consumption : 12.387 Liters
Fuel consumption average (using distance and liters): 14.935 Km/L
```

Resultados e possibilidades

- Um computador de bordo “bombado”
- Data logger
- Telemetria
- Controle de frotas
- Informação de velocidade no transporte público
- Perfis de forma de dirigir
- etc

Referências

- <https://github.com/cioban/obdpy>
- http://en.wikipedia.org/wiki/OBD-II_PIDs
- http://en.wikipedia.org/wiki/On-board_diagnostics
- <https://github.com/oesmith/obdgpslogger/blob/master/doc/mpg-calculation>
- <http://www.mp3car.com/engine-management-obd-ii-engine-diagnostics-etc/75138-calculating-mpg-from-vss-and-maf-from-obd2.html>
- <http://en.wikipedia.org/wiki/Gasoline>
- <http://www.investidorpetrobras.com.br/pt/servicos/formulas-de-conversao/detalhe-formulas-de-conversao/densidade-e-poderes-calorificos-superiores.htm>

Sergio Cioban Filho
cioban@virtmasters.com

twitter: @cioban

<https://br.linkedin.com/in/cioban>
<https://github.com/cioban>

Obrigado...