

# Car Hacking for Ethical Hackers

Dr. Bryson Payne, GPEN, CEH, CISSP



UNG Center for Cyber Operations  
(CAE-CD) 2016-2021

Languages ★ Leadership ★ Cyber



**UNG**  
UNIVERSITY of  
NORTH GEORGIA™

# Why Car Hacking?

- Internet-connected and self-driving cars have become more commonplace – “datacenters on wheels”
- Highly publicized hacks against production cars in the news
- Securing smart cars is matter of public and individual safety
- Integrates well into an ethical hacking/reverse engineering course or program of study, across all 7 NICE CWF categories



SECURELY  
PROVISION



OPERATE &  
MAINTAIN



OVERSEE &  
GOVERN



PROTECT &  
DEFEND



ANALYZE



COLLECT &  
OPERATE



INVESTIGATE

UNG

NICE  
NATIONAL INITIATIVE FOR  
CYBERSECURITY EDUCATION

[nist.gov/nice](http://nist.gov/nice)

NIST  
National Institute of  
Standards and Technology  
U. S. Department of Commerce

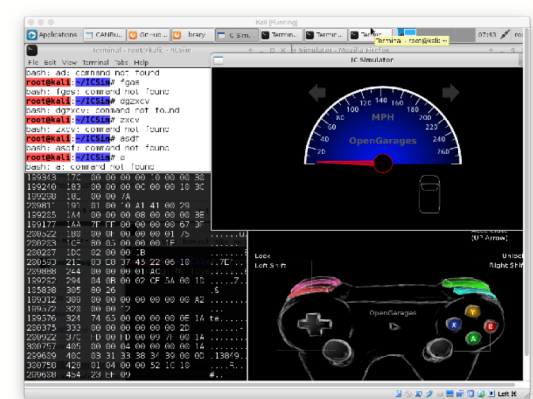
# Introduction



- Self-driving cars have logged millions of miles with significantly fewer accidents than human drivers
- Rapid adoption of driver-assist, semi-autonomous, and internet-connected features makes Car Hacking timely topic
- Automobile networks increasingly complex, 10's of millions of lines of code, decades-old protocols with little security
- Tools needed to access Controller Area Networks (CAN) range from under \$20 to \$80 USD, plus open-source utils

# Goals

- Describe implementation of hands-on car-hacking module in an ethical hacking computer security course
- Detailed setup of free, open-source car-hacking tools
- Demonstration of a replay attack on a virtual CAN network
- Show low-cost tools needed to test vehicle security in real automobiles
- Using Kali Linux, can-utils, ICSim, scantool, Wireshark, tcpdump -> crossover with pentesting, NetSec, IoTSec



# Background



- Automobiles increasingly sophisticated – but CAN bus is largely unchanged, unauthenticated UDP network since 1991
- 2016 Ford F150 unveiled at CES: 150 million lines of code?!?!
- Broad attack surfaces: Bluetooth, Wi-Fi, 4G LTE, USB, OBD-II
- Car hacking shares similarities with hacking other networked devices: network sniffer, open-source tools, reverse engineer
- Good tie-in to ethical hacking/RevEng/NetSec courses

# Intro to the CAN Bus



- CAN (controller area network) bus enables communication between the vehicle's sensors and its various electronic control units (ECUs)
- Modern production cars can have 70 or more ECUs: engine, airbags, anti-lock brakes, tail lights, entertainment system,...
- Message-based protocol standardized in 1991 by Bosch
- UDP – fewer comm delays, broadcast over fewer wires
- 8-16 bytes, no addresses, just priority value/ID

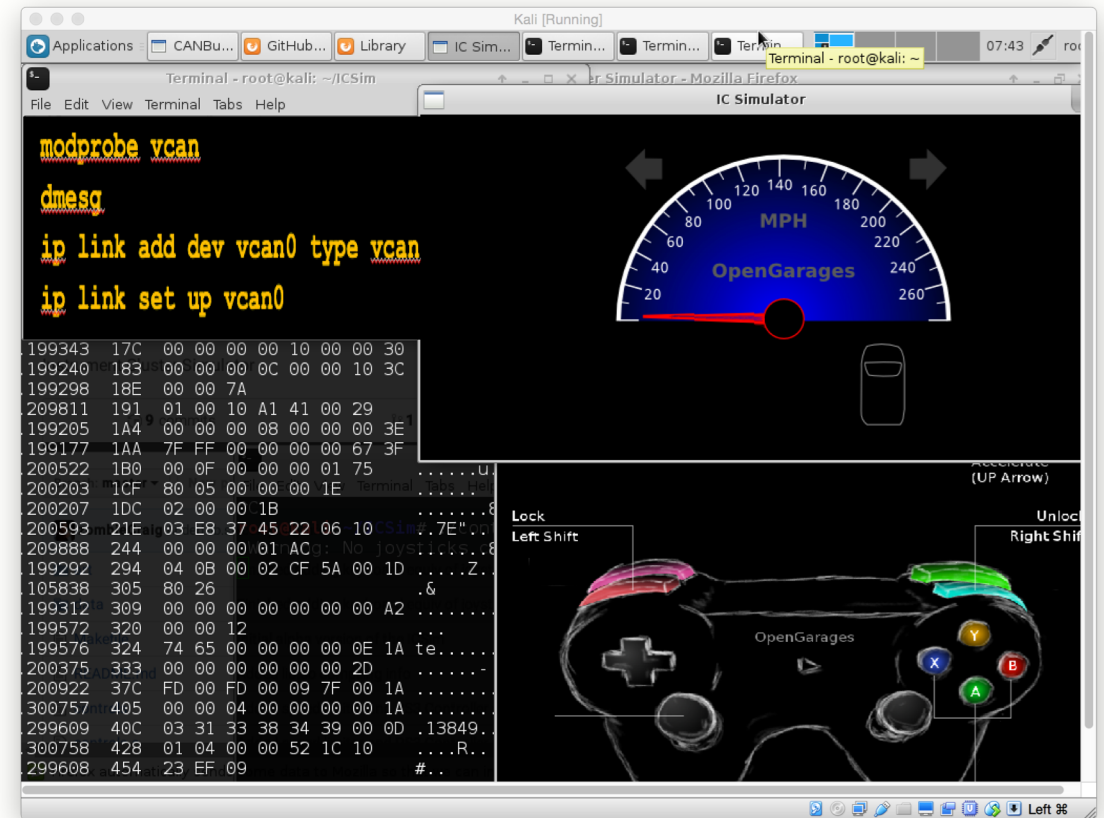
# Brief History of Car-Hacking



- 2011 – UCSD (Checkoway et al.) hack 2011 Chevy Malibu – lock up brakes while driving w/ two different remote attacks
- 2015 – Miller and Valasek remotely controlled steering, braking, acceleration, A/C, stereo, etc. in 2015 Jeep Cherokee
- Researchers recommended TLS encryption – were shocked to learn CAN would need to implement TCP first...
- 2016 Tesla Model S, 2018 BMW i3 by Tencent's Keen Security Lab

# Open-Source Toolkits for Car Hacking

- CAN Utilities (can-utils) included in some Linux distros, most package installer repositories
- Instrument Cluster Simulator (ICSim) from OpenGarages.org
- Scantool, Wireshark, tcpdump
- Easy to set up on Kali Linux
- Other favorites?





# Implementation



- Virtual machine running Kali Linux (VBox, VMware)
- Dependencies:

```
sudo apt-get update
```

```
sudo apt-get install libsdl2-dev libsdl2-image-dev
```

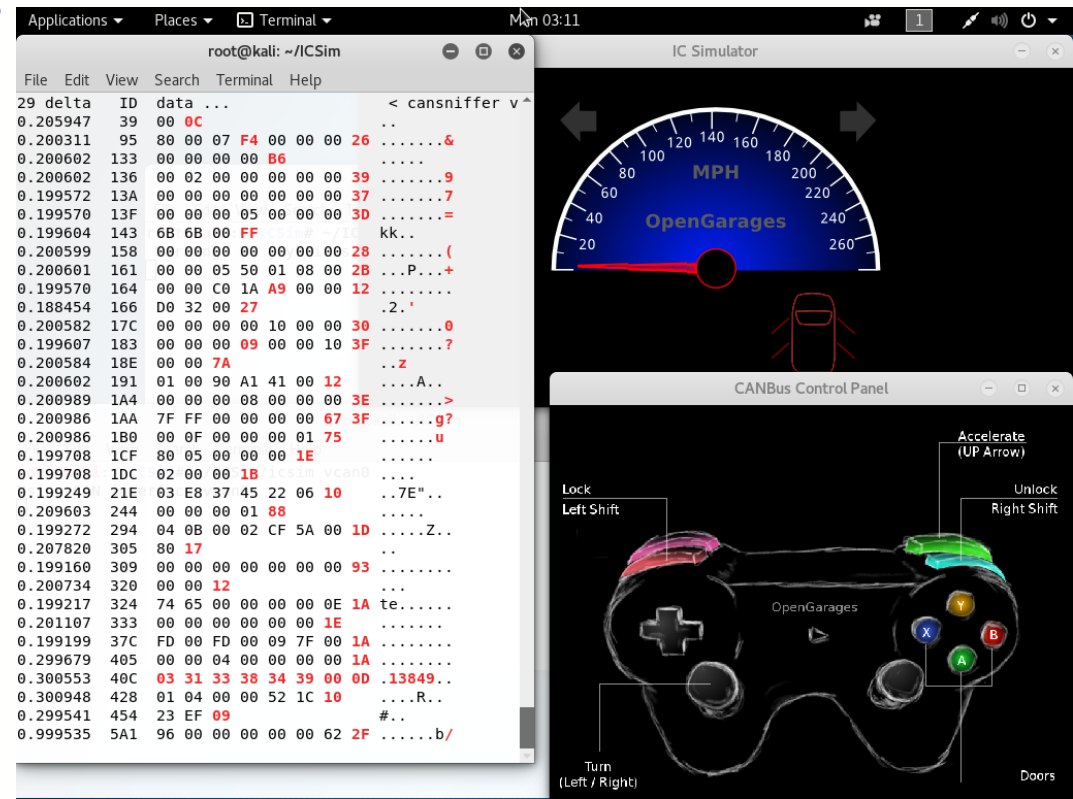
```
sudo apt-get install can-utils
```

- Install ICSim:

```
git clone https://github.com/zombieCraig/ICSim.git
```

# Implementation (cont)

- Prepare Virtual CAN Network:  
`sh ICSim/setup_vcan.sh`
- Verify vcan0 network link is active:  
`ifconfig`
- Run ICSim in **three** terminal windows:
  - ① `~/ICSim/icsim vcan0`
  - ② `~/ICSim/controls vcan0`
  - ③ `cansniffer -c vcan0`



# DEMO: Replay Attack

- Replay attack is classic, works on many IoT and some ICS systems
- Capture CAN bus packets:

```
candump -l vcan0
```

*{-l is lowercase "L" for 'log'}*

- Replay CAN bus packets:

```
canplayer -I candump-2018-07-23_083845.log
```

- Turn off controller window, ICSim will run from log data

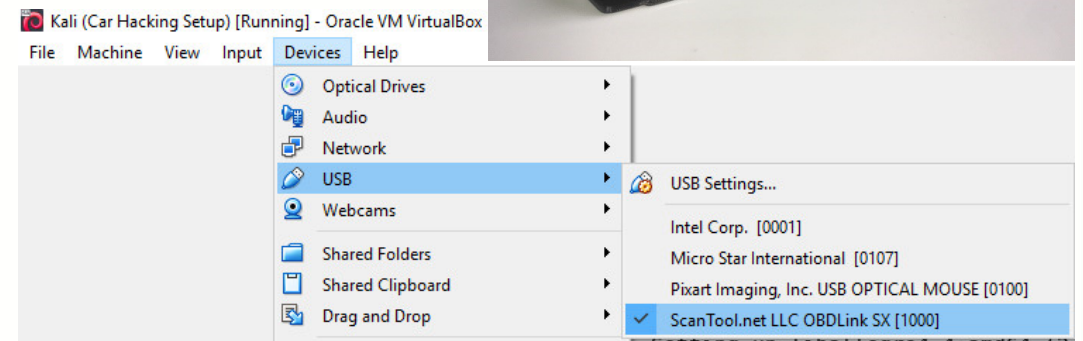




# Extending to Real Life Automobiles

- Easy first step is just displaying OBD-II (on-board diagnostic port) data on PC/Mac/Linux
- ScanTool (free, open-source) and an OBDLink cable (\$29) give you full OBD access
- ScanTool:  

```
sudo apt-get install scantool  
scantool
```
- Connect OBDLink to your Kali VM  
**Devices > USB > ScanTool OBDLink**



root@kali: ~

File Edit View Search Terminal Help

Setting up liballegro4.4:amd64 (2:4.4.2-12) ...

Setting up scantool (1.21+dfsg-6) ...

Processing triggers for libc-bin (2.27-5) ...

root@kali:~# scantool

Mon Jul 23 05:48:56 2018

Version: 1.21 for DOS

Initializing All Modules...

Initializing Allegro...

Installing Timers... OK

Installing Keyboard... OK

Installing Mouse... OK

Loading Preferences... OK

Trying Windowed Graphics

Loading Data File... OK

Initializing Serial Modul

Opening COM0... Dzcomm: U

r

Displaying Main Menu...

-----

opened /dev/ttyUSB0

IC Simulator

ScanTool.net 1.21

**ScanTool.net**

*Inexpensive Alternatives  
for the Do-It-Yourself  
Auto Mechanic*

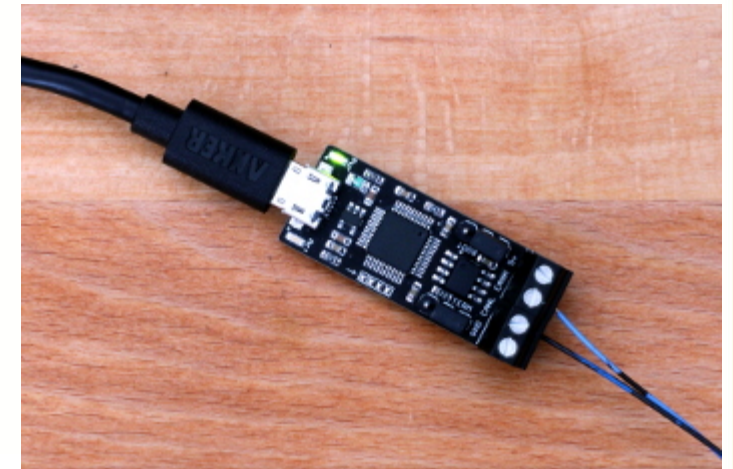


Read codes and their definitions, turn off MIL and erase diagnostic test data.

**Read Codes****Sensor Data****Freeze Frame****Tests****Options****About****Exit**

# Car Hacking on a Real Automobile

- OBDLink may be readable on `ttyUSB/usbmonX` as serial data, but unreliable in practice
- Need true CAN to USB connection
- Cheapest: **CANable \$29.95** – shown here-> from [canable.io](http://canable.io) – direct wiring to CAN pins
- Less MacGyver-ish and more durable: **CANtact (\$65)** plus OBD-CAN cable (\$10) shown here ->



# Further Extension: Hack the Car Hacking SW

- ICSim is open-source, as are can-utils, scantool, etc.
- Fun extension: hack the car-hacking tools!
- Change the max speed of the ICSim dashboard speedometer:
- In controls.c, change

```
#define MAX_SPEED 90.0
```
- to

```
#define MAX_SPEED 300.0
```
- Then, `make` and run

# Conclusion

- You can set up free, open-source car-hacking software for your classes and for your own automotive security research
- Go to **BrysonPayne.com** for a shortened/condensed version of these instructions
- JCERP publication forthcoming with full, step-by-step instructions, all commands, references, resources







UNG Center for Cyber Operations Education

NSA/DHS National Center of Academic  
Excellence in Cyber Defense  
(CAE-CD) 2016-2021

Languages ★ Leadership ★ Cyber  
<http://www.ung.edu/cyber>