Hacking (autonomous) Vehicles

A brief survey of recent work

SparkFun AVC

Updated: 17 September 2016
Overview

1. Introduction

2. HOPE XI Talks

3. Blackhat 2016 Talks

4. DEF CON 2016 Tracks

5. DEF CON 2016 Car Hacking Village

6. Conclusion
INTRODUCTION
Hi, I’m Josh!

- Owner of Cryptotronix
- SparkFun Hacker-in-Residence 2014
- CryptoCape & CryptoShield
- DEF CON 22: NSA Playset: CHUCKWAGON
- BSides Portland: Cryptowarez
Why this talk?!

Why this talk?
Why this talk?!

1. SparkFun is awesome
Why this talk?

1. SparkFun is awesome
2. Wanted to share the latest research
Why this talk?

1. SparkFun is awesome
2. Wanted to share the latest research
3. I hope to see others giving security talks at SparkFun AVC!

**Goal**

Increase security awareness among hackers, makers, builders, and consumers!
Cory Doctorow

A car is a computer that drives.

Sandy Clark

1. Everything is made of software.
2. Software is insecure.
A car is a computer that drives.

Cory Doctorow
Cory Doctorow

A car is a computer that drives.

Sandy Clark

1. Everything is made of software.
2. Software is insecure.
Therefore...
Therefore...

Every car is insecure.

Also, bumps kill.
Connected Car

Source: http://articles.sae.org/14503/
Related Tutorials

**Getting Started with OBD-II**

*OCTOBER 8, 2015*

A general guide to the OBD-II protocols used for communication in automotive and industrial applications.

**CAN-Bus Shield Hookup Guide**

*OCTOBER 8, 2015*

A basic introduction to working with the CAN-Bus shield.

HOPE XI TALKS
You *really* want this book
List of tools, how to create your own car hacking test bench, how to attack ECUs

Source: https://www.nostarch.com/carhacking
You *really* want this book

List of tools, how to create your own car hacking test bench, how to attack ECUs

- UDSim - https://github.com/zombieCraig/UDSim
- http://opengarages.org

Source: https://www.nostarch.com/carhacking
You *really* want this book
List of tools, how to create your own car hacking test bench, how to attack ECUs
UDSim - https://github.com/zombieCraig/UDSim
http://opengarages.org
50% off e-book from nostarch.com, use code JUSTBECAUSE.
○ CANtact - CAN to USB Converter based on the STM32F0

○ CANtact-app
  https://github.com/linklayer/cantact-app

○ Python library

○ Open source hardware
Blackhat 2016 Talks
Wanted a tool to meet the CAN timing constraints (up to 1Mbps, so no UART).

Also needed a MITM configuration (multiple I/F).

Open source framework

STM32F4DISCOVERY Board

Lots of RT firmware work

https://bitbucket.org/jcdemay/canspy

Source: http://ubm.io/2cBdRuA
Advanced CAN Injection - Miller & Valasek

CAN Message injection: contention and conflict issues.

2010 & newer vehicles attacks are speed limited.

A *lot* of reverse engineering of binary blobs.

Two problems

1. Real ECU is spewing real data
2. Target ECU is expecting a counter

message un-confliction, exploits a CAN bus counter.

1. Start to reprogram the ECU, it goes into boot-rom mode, then stop, then drive, which takes the ECU offline.
2. PSCM: PAM in boot-rom, fake speed with counter trick, send PAM message to turn steering wheel.
3. Brakes as well.

Recommends IDS for CAN Bus.
- CAN Message injection: contention and confliction issues.
- 2010 & newer vehicles attacks are speed limited.
- CAN Message injection: contention and confliction issues.
- 2010 & newer vehicles attacks are speed limited.
- A *lot* of reverse engineering of binary blobs.
- Two problems
  1. Real ECU is spewing real data
O CAN Message injection: contention and conflicton issues.

O 2010 & newer vehicles attacks are speed limited.

O A *lot* of reverse engineering of binary blobs.

O Two problems
  1. Real ECU is spewing real data
  2. Target ECU is expecting a counter

O Recommends IDS for CAN Bus.
Advanced CAN Injection - Miller & Valasek

- CAN Message injection: contention and confliction issues.
- 2010 & newer vehicles attacks are speed limited.
- A *lot* of reverse engineering of binary blobs.
- Two problems
  1. Real ECU is spewing real data
  2. Target ECU is expecting a counter
- message un-confliction, exploits a CAN bus counter.

Recommends IDS for CAN Bus.
- CAN Message injection: contention and confliction issues.
- 2010 & newer vehicles attacks are speed limited.
- A *lot* of reverse engineering of binary blobs.
- Two problems
  1. Real ECU is spewing real data
  2. Target ECU is expecting a counter
- *message un-confliction*, exploits a CAN bus counter.
  1. Start to reprogram the ECU, it goes into boot-rom mode, then stop, then drive, which takes the ECU offline.

Recommends IDS for CAN Bus.
• CAN Message injection: contention and confliction issues.
• 2010 & newer vehicles attacks are speed limited.
• A *lot* of reverse engineering of binary blobs.
• Two problems
  1. Real ECU is spewing real data
  2. Target ECU is expecting a counter
• message un-confliction, exploits a CAN bus counter.
  1. Start to reprogram the ECU, it goes into boot-rom mode, then stop, then drive, which takes the ECU offline.
  2. PSCM: PAM in boot-rom, fake speed with counter trick, send PAM message to turn steering wheel.

Recommends IDS for CAN Bus.
- **CAN Message injection:** contention and conflicton issues.
- **2010 & newer vehicles attacks are speed limited.**
- **A *lot* of reverse engineering of binary blobs.**
- **Two problems**
  1. Real ECU is spewing real data
  2. Target ECU is expecting a counter
- **message un-confliction**, exploits a CAN bus counter.
  1. Start to reprogram the ECU, it goes into boot-rom mode, then stop, then drive, which takes the ECU offline.
  2. **PSCM:** PAM in boot-rom, fake speed with counter trick, send PAM message to turn steering wheel.
  3. Brakes as well.

Recommends IDS for CAN Bus.
- CAN Message injection: contention and confliction issues.
- 2010 & newer vehicles attacks are speed limited.
- A *lot* of reverse engineering of binary blobs.
- Two problems
  1. Real ECU is spewing real data
  2. Target ECU is expecting a counter
- message un-confliction, exploits a CAN bus counter.
  1. Start to reprogram the ECU, it goes into boot-rom mode, then stop, then drive, which takes the ECU offline.
  2. PSCM: PAM in boot-rom, fake speed with counter trick, send PAM message to turn steering wheel.
  3. Brakes as well.
- Recommends IDS for CAN Bus.
DEF CON 2016 Tracks
CAN I Haz Secretz? Vidal & Noeischer

CANBadger Hardware Overview

- Powered by mBed LPC1768 or LPCXPresso LPC1769
- 128KB XRAM
- 2x DB9 CAN Interfaces + 2x Debug headers
- SD card
- ECU Power control by software
- UART
- 4 GPIOs
- Standalone mode, USB mode (CDC Device), or Network mode
- Can be powered by PSU, External battery, or OBD2
- Has a blinky dual color LED. Everyone loves blinky LEDs, right?
- Complete board assembly under $25
- CANBadger can control GPS via UART
- CANBadger can control GPS via UART
- How do those insurance dongles work?
- CANBadger can control GPS via UART
- How do those insurance dongles work?
- CANBadger has emulator. Records data and playbacks.
Can you Trust Autonomous Vehicles—Liu, Yan, Xu
How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.
○ How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.
○ First fatal tesla crash using autopilot on May 7, 2016.
How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.

First fatal Tesla crash using Autopilot on May 7, 2016.

Spoof Tesla HMI
○ How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.
○ First fatal Tesla crash using Autopilot on May 7, 2016.
○ Spoof Tesla HMI
○ Attacking ultra-sonic sensors (proximity)
○ How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.
○ First fatal tesla crash using autopilot on May 7, 2016.
○ Spoof tesla HMI
○ attacking ultrasonic sensors (proximity)
  1. Prevent cars from taking your parking spot :)

Takeaways
1. Sensors should fail safe.
2. Sensor should have anomaly detection
3. Redundancy: MIMO, different types, sensor data fusion
○ How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.

○ First fatal tesla crash using autopilot on May 7, 2016.

○ Spoof tesla HMI

○ attacking ulta-sonic sensors (proximity)
  1. Prevent cars from taking your parking spot :)
  2. attacks: jamming, spoofing, quieting (Arduino + ultrasonic transducer or signal generator)
How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.

First fatal tesla crash using autopilot on May 7, 2016.

Spoof tesla HMI

attacking ultra-sonic sensors (proximity)
1. Prevent cars from taking your parking spot :)
2. attacks: jamming, spoofing, quieting (Arduino + ultrasonic transducer or signal generator)

attacks on cameras (blinding with laser pointer)
How can we attack the sensors? The reliability of the sensors affect the reliability of the driving.

First fatal tesla crash using autopilot on May 7, 2016.

Spoof tesla HMI

attacking ultra-sonic sensors (proximity)
  1. Prevent cars from taking your parking spot :)
  2. attacks: jamming, spoofing, quieting (Arduino + ultrasonic transducer or signal generator)

attacks on cameras (blinding with laser pointer)

Takeaways
  1. Sensors should fail safe.
  2. sensor should have anomaly detection
  3. Redundancy: MIMO, different types, sensor data fusion
DEF CON 2016 Car Hacking Village
1. The badge supports 2x Dual Wire CAN (ISO 11898-2) channels with full gateway support.
2. Fully-controllable pass-through
3. Interrupt messages or drop it
5. The badge also sports a 128×128 color LCD which is fully accessible via the PAWN scripts.
Conclusion
Insights

CAN Bus feels like the early Internet

Exploit the boundaries of assumptions.

Defensive design.

Reliability engineering is security engineering.

See Peter Neumann and Computer Related Risks
CAN Bus feels like the early Internet

Exploit the boundaries of assumptions.
CAN Bus feels like the early Internet

Exploit the boundaries of assumptions.

- Defensive design.
Insights

**CAN Bus feels like the early Internet**

Exploit the boundaries of assumptions.

- Defensive design.
- Reliability engineering is security engineering.
CAN Bus feels like the early Internet

Exploit the boundaries of assumptions.

- Defensive design.
- Reliability engineering is security engineering.
- See Peter Neumann and *Computer Related Risks*
Get more security researchers at AVC!
1. I wonder if I could trigger a competitor’s stop switch :)

Get more security researchers at AVC!
Get more security researchers at AVC!

1. I wonder if I could trigger a competitor’s stop switch :) 
2. Ultrasonic-proof (quieted) obstacles
Get more security researchers at AVC!

1. I wonder if I could trigger a competitor’s stop switch :)  
2. Ultrasonic-proof (quieted) obstacles  
3. Actively malicious obstacles targeting sensors
1. I wonder if I could trigger a competitor’s stop switch :) 
2. Ultrasonic-proof (quieted) obstacles 
3. Actively malicious obstacles targeting sensors 
4. Pen-test pit-stop
Get more security researchers at AVC!

1. I wonder if I could trigger a competitor’s stop switch :) 
2. Ultrasonic-proof (quieted) obstacles 
3. Actively malicious obstacles targeting sensors 
4. Pen-test pit-stop
1. I wonder if I could trigger a competitor’s stop switch :)  
2. Ultrasonic-proof (quieted) obstacles  
3. Actively malicious obstacles targeting sensors  
4. Pen-test pit-stop  

**MAKE AVC ENTRIES RESILIENT TO ATTACKS**  
You know, like real-world systems should be!
Das Ende

- www.cryptotronix.com
- Just ask for Josh ;)

![Image of a cartoon character]