Security Concerns in Automotive Systems

James Martin

Main Questions

1. What sort of security vulnerabilities do modern cars face today?
2. To what extent are external attacks possible and practical?

Background

- Internal network of computers (ECUs)
  - Drivetrain, brakes, lighting, entertainment
- “Controller Area Network” of ECUs, “CAN”
- Components have buses
- Multiple CAN buses for subsets of ECUs
  - For example: one for safety-critical components like the engine, one for radio/entertainment center
  - Not done for security reasons, but bandwidth reasons

Background

- **Control of only one bus is needed to compromise the entire CAN**
- On a given bus, each component has at least implicit access to every other component
- Can spoof messages to isolated components

Attack surfaces

1. Indirect physical access
    1. Physical media: CD, USB, MP3 players
    2. OBD-II port (more on this later)
2. Short range wireless access (up to 300m)
    1. Bluetooth, keyless entry, tire pressure monitoring systems (TPMS)
3. Long range wireless access (>1km)
    1. Broadcast receivers (XM/HD Radio)
    2. Addressable channels such as OnStar (more on this later)

Vulnerabilities

Highlighted in this presentation:

1. OBD-II port
2. Malicious CDs/audio
3. Bluetooth attacks
4. Addressable channel attacks

OBD-II Port

- Mandated to be in every car by the US federal government
- Provides direct access to CAN
- Used by mechanics with diagnostic tools, or “PassThru” device
  - Standard SAE J2534 API to access and program ECUs, Windows API implemented as a DLL
  - Commonly connect to these devices with WiFi

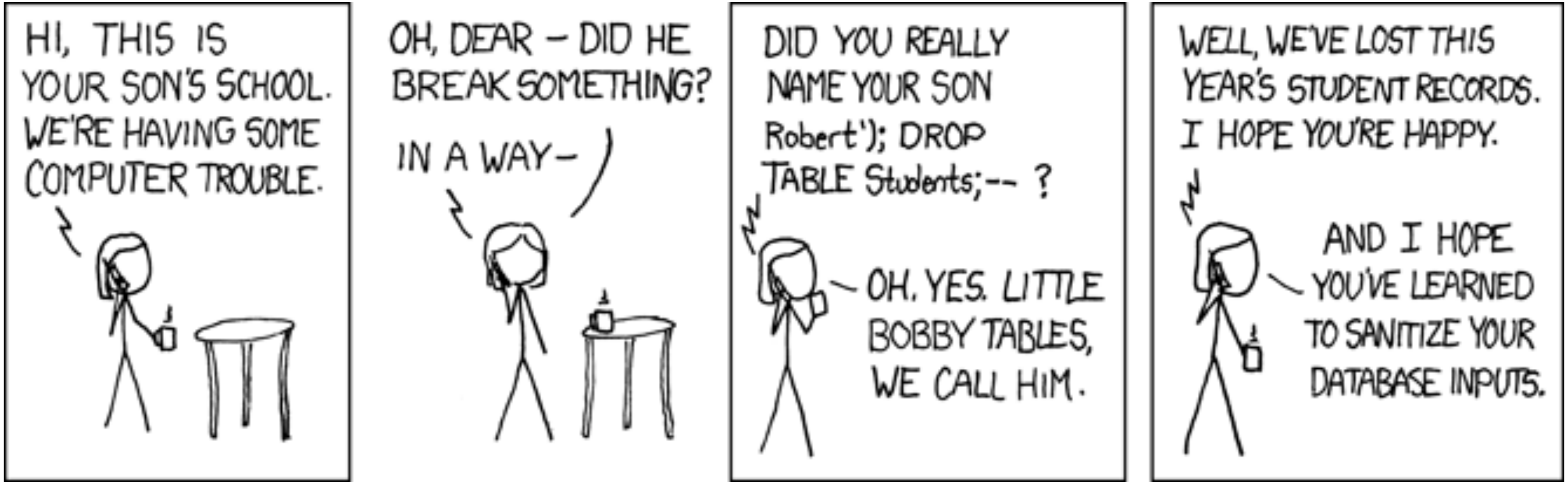
OBD-II Attack

- Vulnerabilities in the PassThru device itself
- Runs a variant of Linux on a SoC microprocessor
- Broadcasts a UDP packet over its connected network with its IP address and TCP port for access
- Only a single application can access a single PassThru device

OBD-II Attack

- PassThru device has a proprietary unauthenticated API for configuring network state
- Vulnerable to shell injection attacks due to input validation bugs
- Linux distro these devices run includes telnet, ftp, and netcat installed
  - Can open arbitrary telnet connections

Code injections in a comic


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

XKCD 327: Exploits of a Mom

OBD-II Attack

1. Contact any PassThru device
2. Exploit with shell injection
3. Install malicious binary
4. Binary sends pre-programmed messages over the CAN bus for the OBD-II port
5. Installs malware onto the car’s telematics unit, waits on certain environmental triggers

Conclusion

- An attacker can compromise a dealership network
- One PassThru device can be turned into a worm that seeks out other PassThru devices in range and infects them
- Now you can infect any car that comes into the dealership and is diagnosed by a PassThru device

Malicious CD/Audio attack

- Media player recognizes an ISO 9660-formatted CD with a certain file name
- Reflashes the unit with data on CD if user does not press appropriate button
- Media player can also parse complex files, certain vulnerabilities in the parser
  - Allows for a buffer overflow attack *

Malicious CD/Audio attack

1. Modified a WMA file that plays fine on PC
2. Sends arbitrary CAN packets when played by media player in car
3. Small overhead to file size
4. Easy to spread via P2P networks
    i. So be careful when torrenting pop music

Bluetooth attacks

Indirect:

- Infect smartphone with Trojan horse app that delivers a malicious payload if paired with a telematics unit

Direct:

- A little more involved
- Once a channel is created, can deliver a payload

Bluetooth attacks

- Sniff Bluetooth traffic, get Bluetooth MAC address of vehicle
- Brute force pairing PIN
    - Takes up to 13.5 hours
    - But can attack in parallel, target parking garages
    - “Expect to brute force a PIN for at least one car within a minute”
    - Assumes the cars have been pre-paired with at least one device

Addressable channel attacks

- aqLink software modem present in most North American cars
  - Converts between analog waveforms and digital bits
- Connects cars to Telematics Call Center (TCC) operated by manufacturer
- Pure data calls for remote diagnostics uses “stealth mode” which does not indicate a call is in progress

Addressable channel attacks

- Reverse engineered aqLink protocol
- Decoded parameters for demodulating digital bits from the raw analog signal
  - Basically found a way to send signals to the modem that were encoded into the bits they wanted
- Decoded packet structure
- Gateway and Command programs in the telematics unit recognize and process packets

Addressable channel attacks

- Stack-based buffer overflow possible in the Gateway program
    - However, need to send 300 bytes for this
    - Authentication required within 12 seconds, takes 14 to transmit 300 bytes
- Attack authentication challenge
    - Random three byte challenge packet
- Random number generator re-initialized when telematics unit starts, seeded w/ constant

Addressable channel attacks

- Code parsing authentication responses has a bug, authenticates without correct response (1 out of 256 times)
- Attack is challenging with car off, telematics unit can shut down when call ends

Conclusion

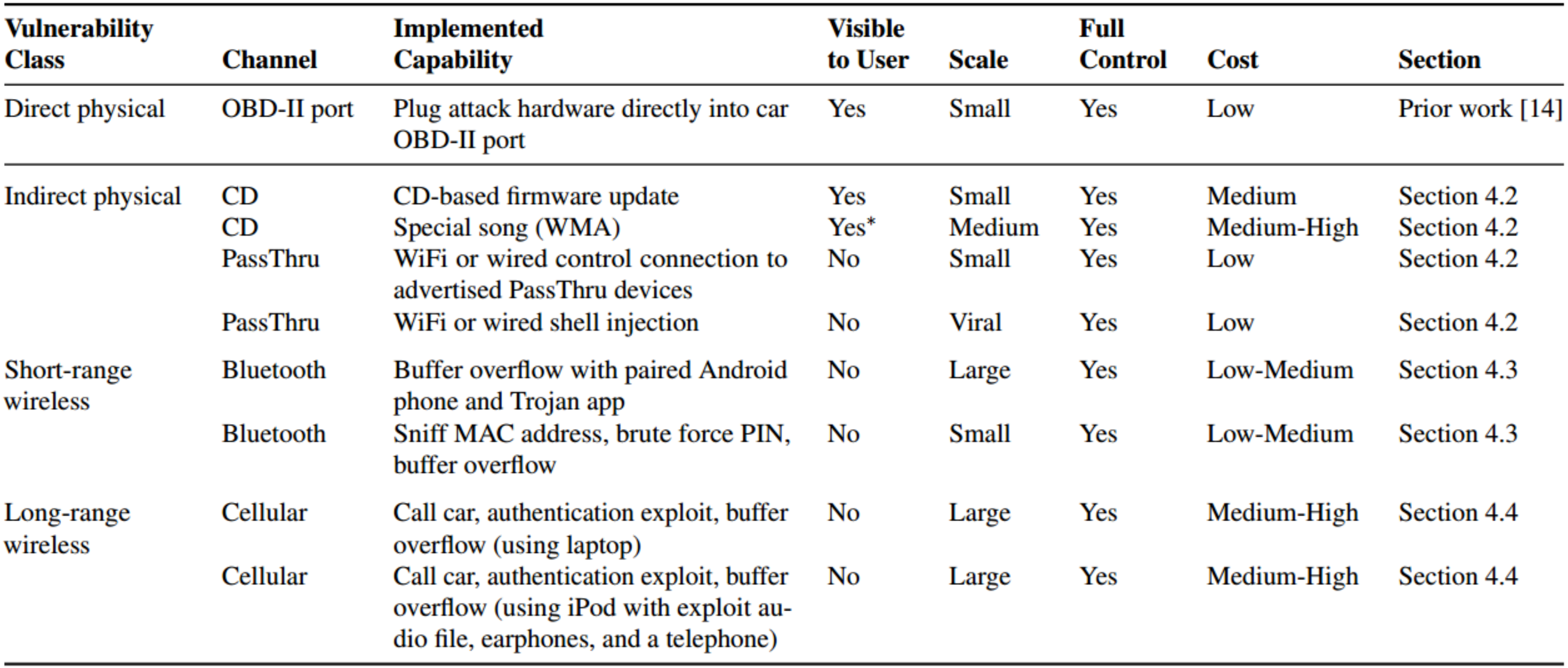
1. Laptop with aqLink compatible software
2. Call car repeatedly until it authenticates
3. Change timeout from 12 to 60 seconds
4. Re-call the car and deliver payload
5. Exploit telematics unit to download code from Internet over IP-addressable 3G

It gets even easier...

Found that the entire attack does not rely on the car’s responses:

- Encode an audio file with modulated post-authentication exploit payload
- Manually dial car from office phone
- Play the audio
- Same results

Summary table of vulnerabilities



Vulnerability Class	Channel	Implemented Capability	Visible to User	Scale	Full Control	Cost	Section
Direct physical	OBD-II port	Plug attack hardware directly into car OBD-II port	Yes	Small	Yes	Low	Prior work [14]
Indirect physical	CD	CD-based firmware update	Yes	Small	Yes	Medium	Section 4.2
	CD	Special song (WMA)	Yes*	Medium	Yes	Medium-High	Section 4.2
	PassThru	WiFi or wired control connection to advertised PassThru devices	No	Small	Yes	Low	Section 4.2
	PassThru	WiFi or wired shell injection	No	Viral	Yes	Low	Section 4.2
Short-range wireless	Bluetooth	Buffer overflow with paired Android phone and Trojan app	No	Large	Yes	Low-Medium	Section 4.3
	Bluetooth	Sniff MAC address, brute force PIN, buffer overflow	No	Small	Yes	Low-Medium	Section 4.3
Long-range wireless	Cellular	Call car, authentication exploit, buffer overflow (using laptop)	No	Large	Yes	Medium-High	Section 4.4
	Cellular	Call car, authentication exploit, buffer overflow (using iPod with exploit audio file, earphones, and a telephone)	No	Large	Yes	Medium-High	Section 4.4

Comprehensive Experimental Analyses of Automotive Attack Surfaces

Practicality

How practical are these attacks?

- Certainly take time and effort
  - Pretty unlikely to create mass destruction w/ malware
- Theft is a real threat
  - Compromise car, send GPS and Vehicle Identification Number to a server, profit
- Surveillance also a threat
  - Compromise telematics unit, use in-cabin mic to record conversations

Questions?

Thanks for listening!

Sources

- Comprehensive Experimental Analyses of Automotive Attack Surfaces
- Experimental Security Analysis of a Modern Automobile
- http://www.today.com/video/today/52609500#52609500
- http://drewtech.com/support/J2534/index.html