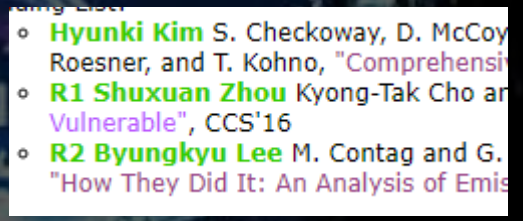
Comprehensive Experimental Analyses of Automotive Attack Surfaces

2018.9.27
Hyunki kim



Authers: **Stephen Checkoway**, Damon McCoy, Brian Kantor,
Danny Anderson, Hovav Shacham, Stefan Savage, (UCSD)
Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno (UW)

KAIST

Written by Sanha Park

Intro

Intro

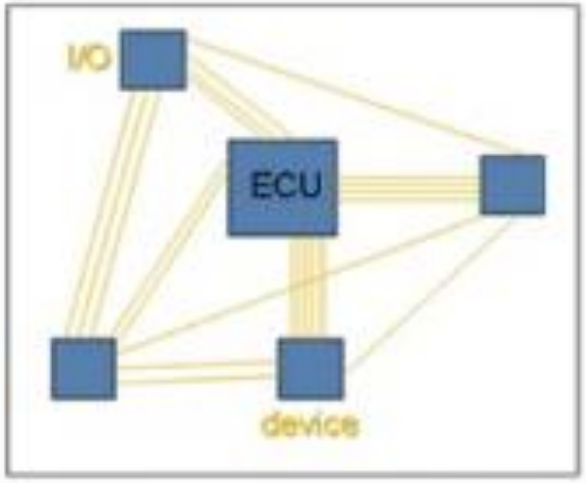
❖ Jeep Cherokee hacked in 2015

Why can we attack?

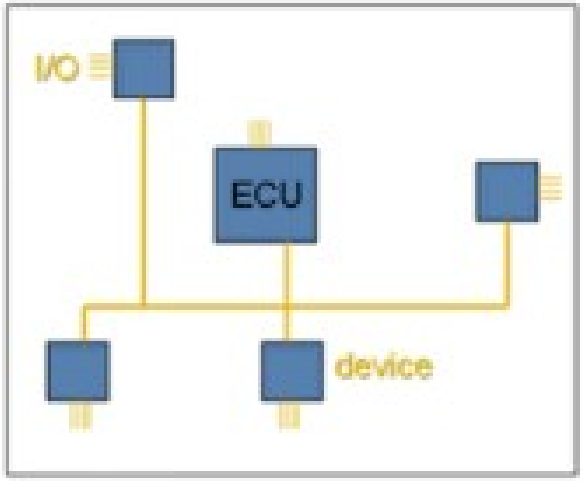
1980's

Today

Without CAN

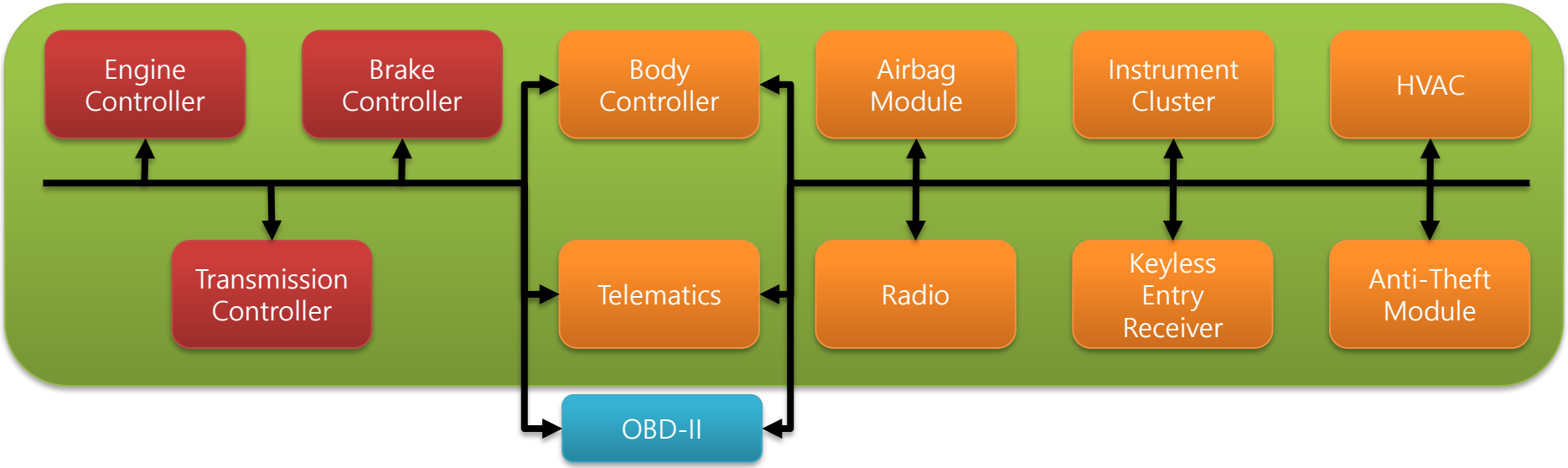


With CAN



Why can we attack?

Cars' system



- ECU(Electronic Control Unit) :
  - Ubiquitous computer controller
- ECU interconnection driven by safety, efficiency, and capability requirements
- But, also has some fatal shortcomings

Oakland 2010, they showed…

- Safety-critical systems can be compromised
  - Selectively enable/disable brakes
  - Stop engine
  - Control lights
- Owning one ECU = total compromise
- ECUs can be reprogrammed (while driving!)

- Limit: Need physical access

[Oakland'10] koscher et al. Experimental Security Analysis of a Modern Automobile.

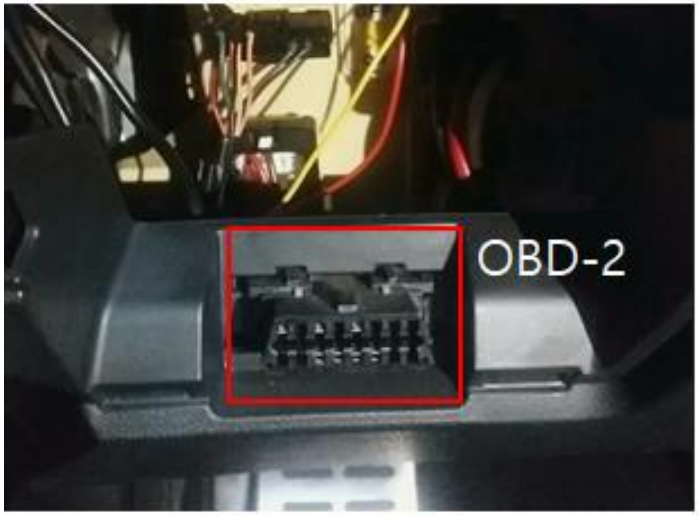
Threat model

- Technical (theoretical) Capabilities
  - Capabilities in analyzing the system
  - Focuses on making technical capabilities realistic

- Operational (real-time) capabilities
  - Show how malicious payload is delivered
  - Attack vector
    - Indirect physical access
    - short-range wireless access
    - long-range wireless access

Indirect physical

- Definition:
  - Attacks over physical interfaces
  - Constrained: Adversary may not **directly** access the physical interfaces herself
- OBD(stands for On Board Diagnostic)



Port

Scanner

PassThru

Indirect physical

- Definition:
  - Attacks over physical interfaces
  - Constrained: Adversary may not **directly** access the physical interfaces herself
- Extends attack surface to the device

Short-range wireless

- Definition: Attacks via short-range wireless communication (meters range or less)

Bluetooth

TPMS

Remote key

Immobilizer

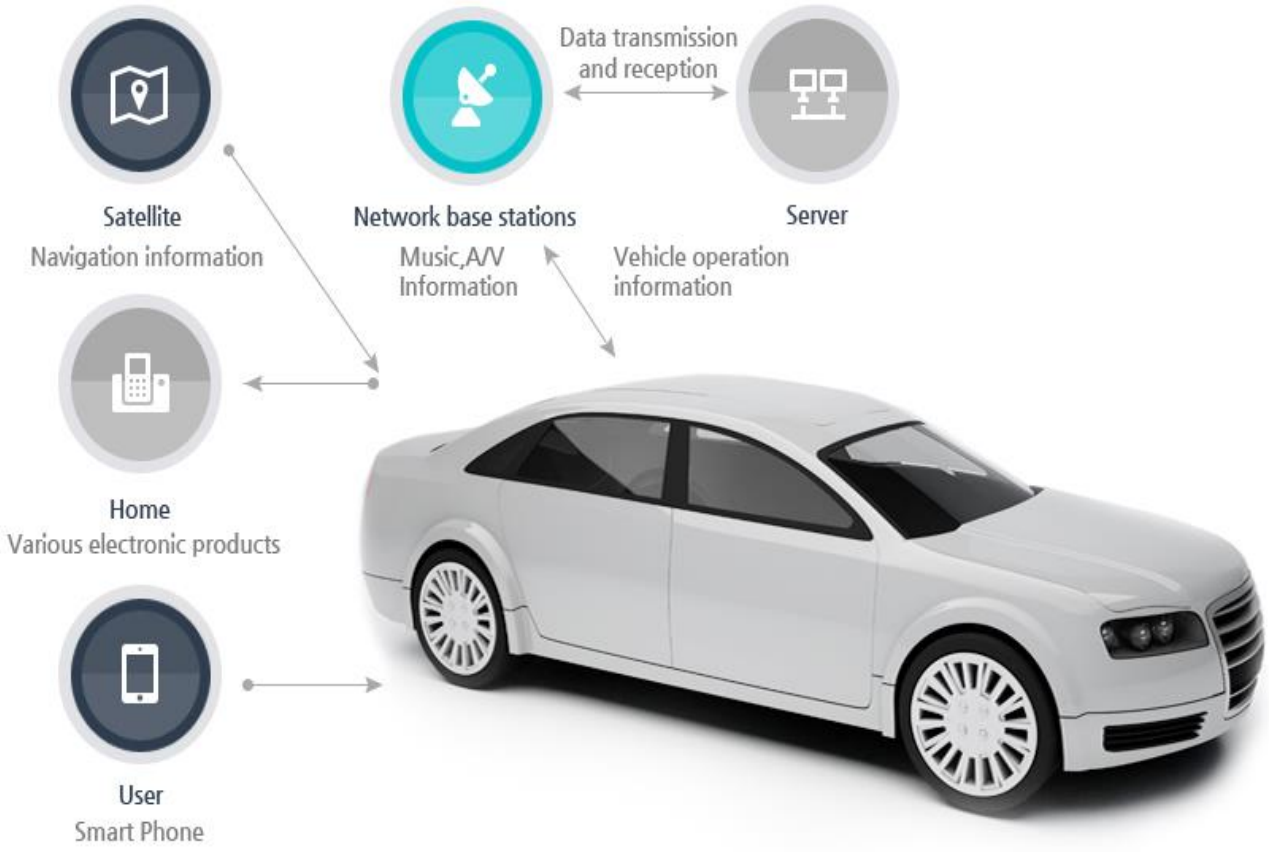
Long-range wireless

- Definition: Attacks via long-rage wireless communication (miles, global-scale)
- Broadcast channel
  - Satellite Radio, GPS, RDS

Satellite Radio

Long-range wireless

- Definition: Attacks via long-rage wireless communication (miles, global-scale)
- Addressable channel
  - Telematics



Attack surfaces explored in depth

- Components we compromised
  - **Indirect physical: Media player, OBDII**
  - **Short-range wireless: Bluetooth**
  - **Long-rage wireless: Cellular**

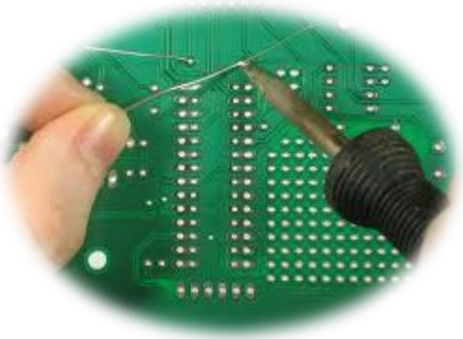
- Every attack vector leads to complete car compromise

Premise

- No direct physical access
- Already know how to deal with CAN signal
- Recent made sedan, 2 same model

Overall methodology

- Extract device's firmware
  - Read memory out over the CAN bus (CarShark)
  - Desolder flash memory chips in ECUs



- Reverse engineering firmware
  - IDA Pro
  - Custom tools

- Identify and test vulnerable code paths

Indirect physical: Media player attack

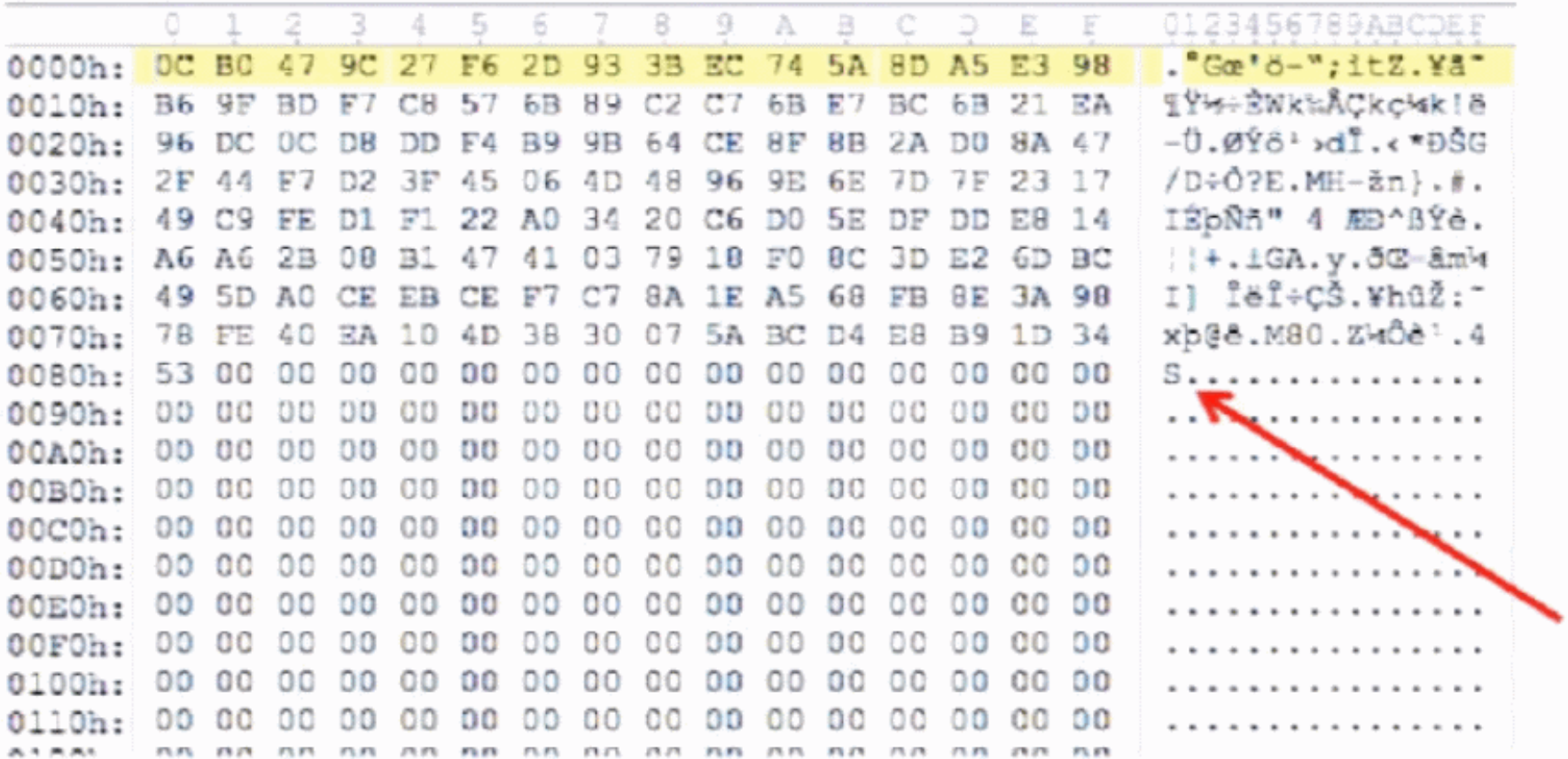
- Code for ISO-9660 leads to
  - Vulnerable : in a module that uploads firmware.

./usr/share/scripts/update/installer/system_module_check.lua

```
 91    local fname= string.format("%s/swdl.iso", os.getenv("USB_STICK")
or "/fs/usb0")
 92    local FLAGPOS=128
 93
 94    local f = io.open(fname, "rb")
 95    if f then
 96       local r, e = f:seek("set", FLAGPOS)
 97       if r and (r == FLAGPOS) then
 98          local x = f:read(1)
 99          if x then
100             if x == "S" then
101                print("system_module_check: skip ISO integrity
check")
```

Indirect physical: Media player attack

- Code for ISO-9660 leads to
  - Vulnerable : in a module that uploads firmware

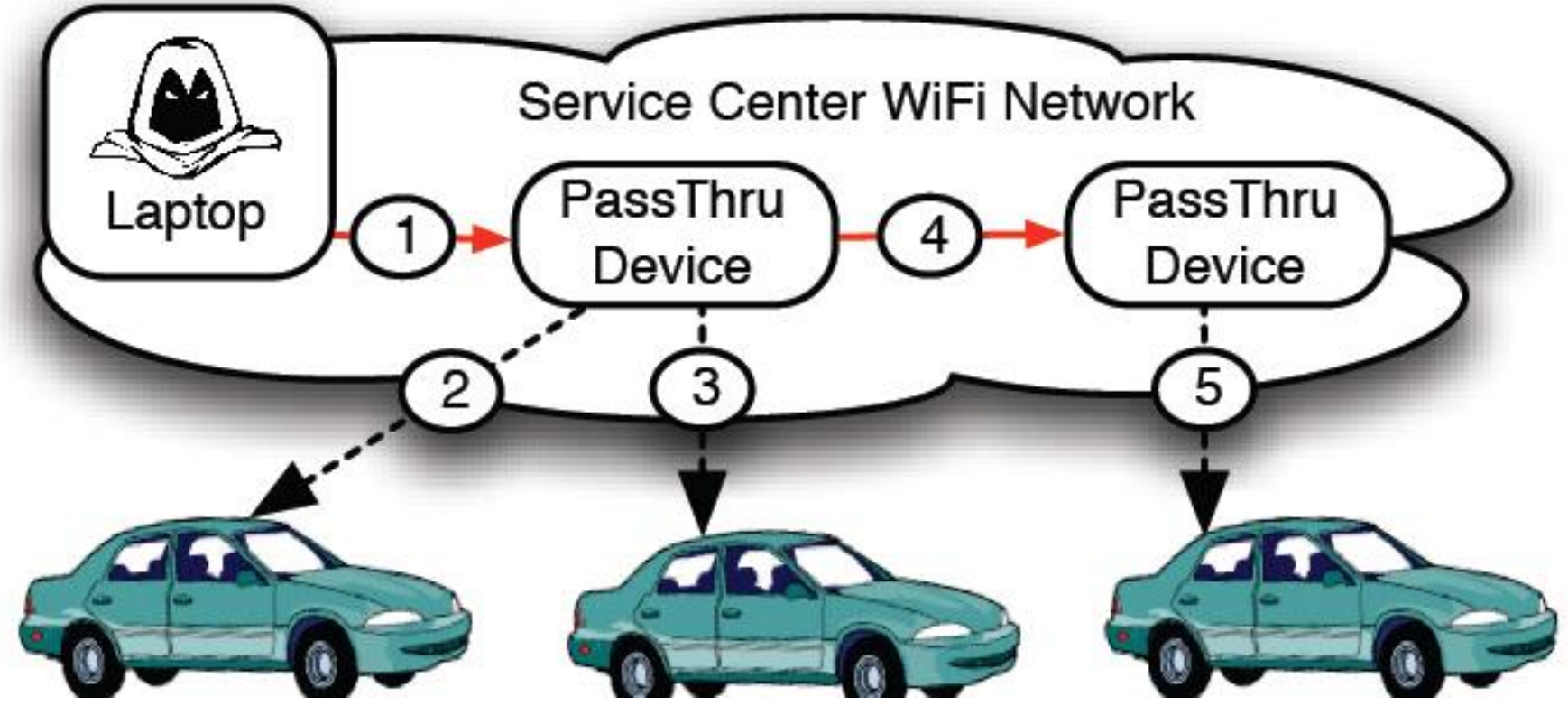


Short-range wireless: OBDII

❖ PassThru device has no authentication method

1. Connect to same WiFi with device to get to CAN bus

2. Implant malicious code inside the device



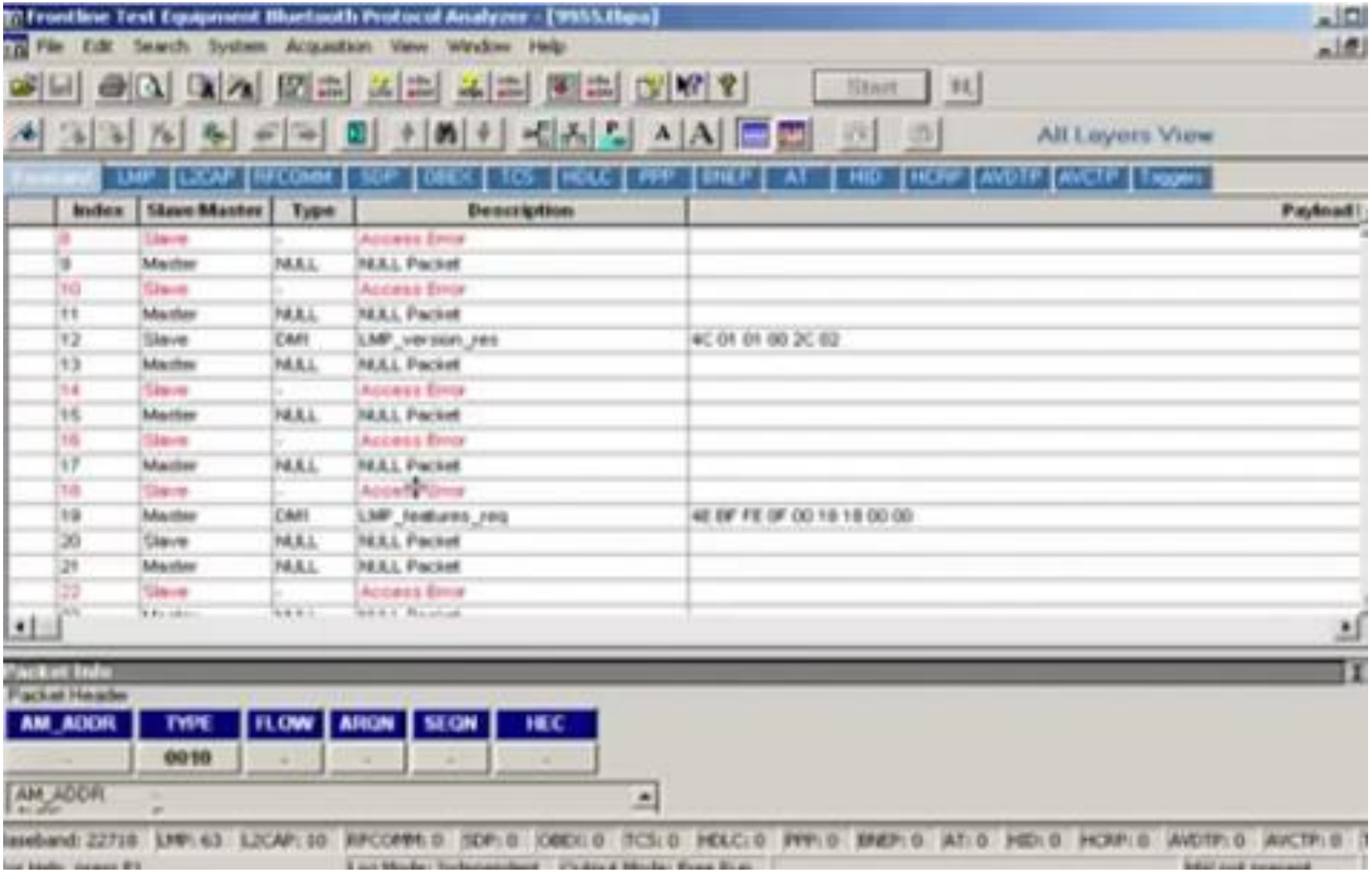
Short-range wireless: Bluetooth attack

❖ Custom-built code contains vulnerability
  - Strcpy() bug → execute arbitrary code(Bufferoverflow)

1. Using owner's smartphone as stepping-stone
   - Trojan Horse application
   - Check whether other party is telematics unit
     → if so it sends our attack payload

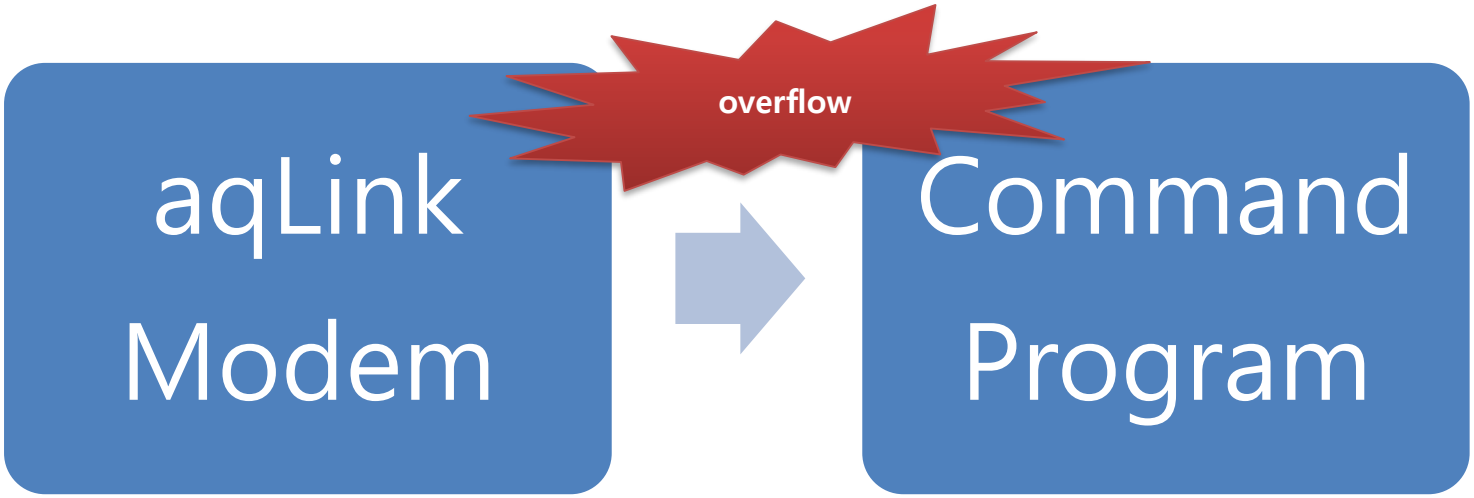
2. Can directly pair with Bluetooth undetectably
   - USRP software radio
   - MAC address ; 2ways to get
   - Brute force PIN ;10hrs per car

Short-range wireless: Bluetooth attack



Long-range wireless: Cellular attack

1. Attack @ Lowest level of protocol stack



Use 1024bytes packet size

Maximum 100bytes packet

Car theft

1. Compromise car
2. Get Car's INFO (GPS…)
3. Unlock doors
4. Start engine
5. Bypass anti-theft

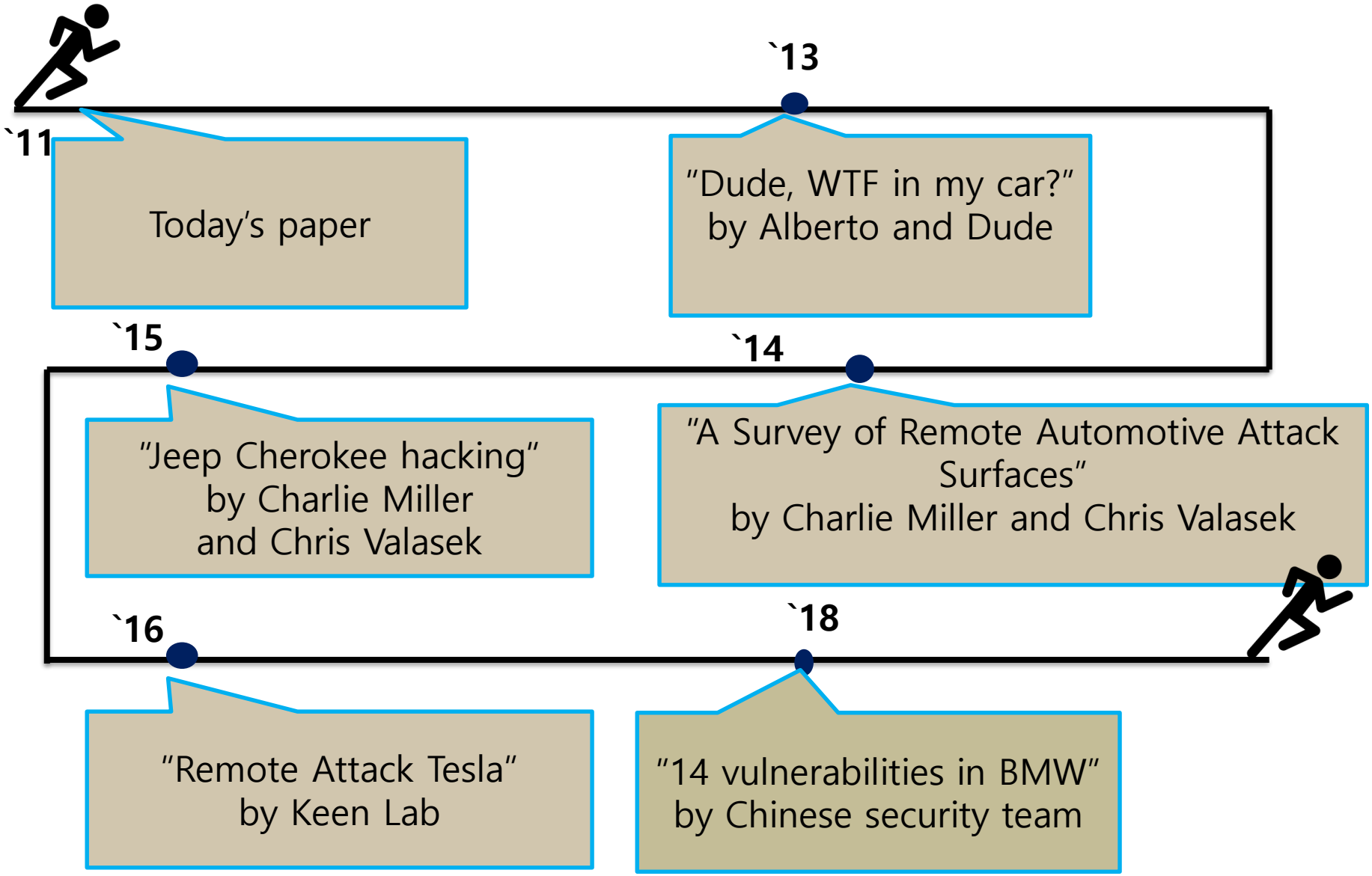
Surveillance

- Compromised car
- Continuously report GPS coordinates
- Stream audio recorded from the in-cabin mic
  - Detect voice (VAD)
  - Compress audio
  - Stream to remote computer

  E.g.) Professor Yongdae Kim

Where to go from here?

Where to go from here?



Where to go from here?

- Stakeholders responding today:
  - SAE, USCAR, US DOT

- Recommendation : lessons from the PC world
  - Avoid unsafe function
  - Remove unnecessary binaries e.g.) ftp/telnet/vi
  - ASLR (Address Space Layout Randomization)
  - Stack cookies
  - Limited inbound calls

Where to go from here?

Achieve excellence in automotive software security

Penetration testing	Replicate the steps a threat agent takes to find vulnerabilities, and receive clear guidance on how to eliminate them in your server-side applications and APIs.
Dynamic application security testing (DAST)	Identify security vulnerabilities while web applications are running, without the need for source code.
Mobile application security testing (MAST)	Find vulnerabilities regardless of where they exist, including in client-side code, server-side code, third-party libraries, and underlying mobile platforms.
Embedded application security testing (EAST)	Verify the functional and security performance of embedded systems, and identify vulnerabilities in the embedded software stack.
Software composition analysis (SCA)	Detect third-party open source components in source code and binaries. Track and remediate vulnerabilities during development and in containers in production. Identify third-party licenses, and set policies to avoid noncompliance.
Tools	Synopsys provides industry-leading tools for software composition analysis, static code analysis, fuzz testing and protocol testing, and interactive security testing.
Architecture and design	Security testing and threat modeling help you find architectural, design, and system defects and flaws.
Cloud security	Run applications securely in the Cloud.
Agile and CI/CD	Build security into modern agile SDLCs.
Training	Synopsys creates security training courses delivered as instructor-led, eLearning, and virtual classes.
Build Security In programs	Synopsys offers the BSIMM, the Maturity Action Plan, security metrics, and software security initiative programs.

SAE INTERNATIONAL ISO MISRA AUTOMOTIVE GRADE LINUX UL DEKRA

Where to go from here?

- Future work
  - Developing new protocol alternative to CAN bus
  - Research how to encrypt CAN message
  - CAN monitoring system to catch external attack

Summary

- Current autos have broad (and increasing) external attack surface
- They demonstrated real attacks that compromised safety-critical systems
- **Industry and government are responsible**


Q&A