UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

Dipartimento di Ingegneria
"Enzo Ferrari"

Automotive Cyber Security

Lecture 3 – Attack surface of a modern vehicle

Mirco Marchetti

Università di Modena e Reggio Emilia

mirco.marchetti@unimore.it

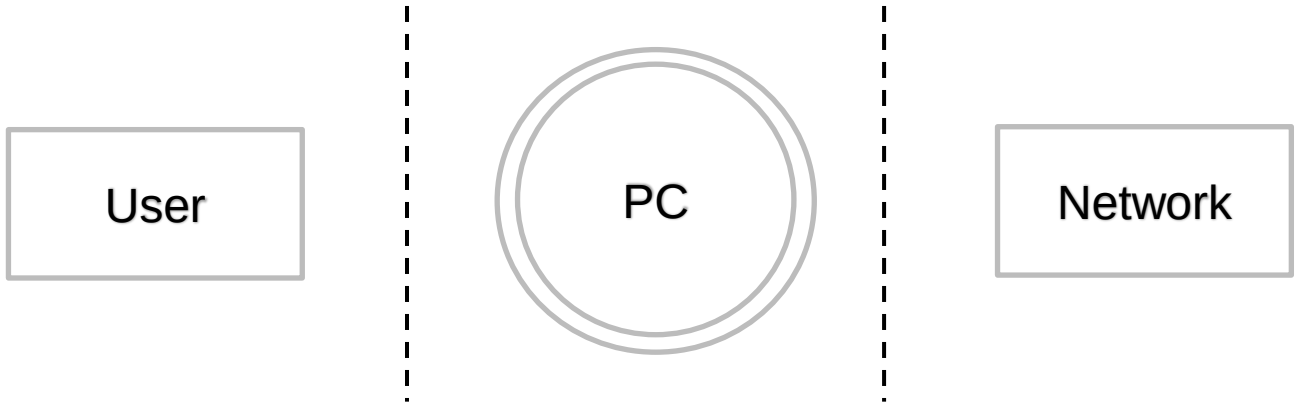
Attack surface

- Attack surface: sum of all attack vectors
- Attack vector: any “entry point”, or “communication channel” that an attacker can use to interact with its target
  - does not imply or require the existence of a software vulnerability
  - does not mean that the attacker can actually exploit this attack vector

- Related concept: Trust boundary

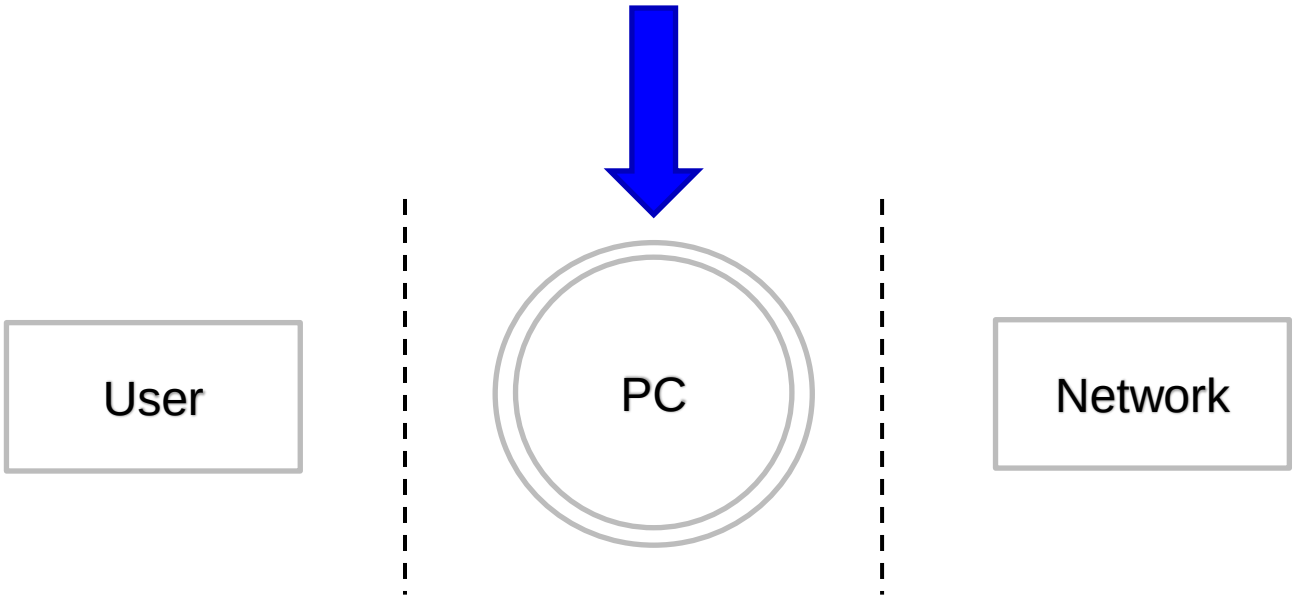
Trust boundary

- Trust boundary: the boundary of a system. It includes all subsystems (sw/hw components and data) that are trusted and that we want to defend from external attackers
- Example: consider a PC that receives data from its local user and from its network interface



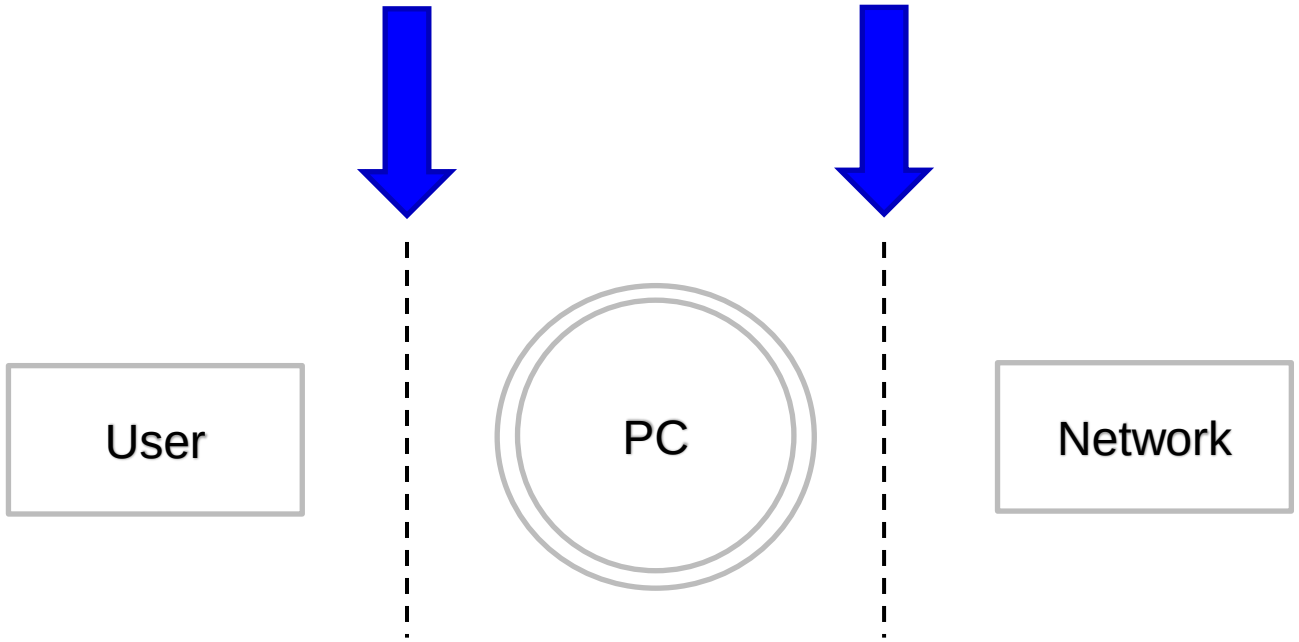
Trust boundary

The double circle indicates a “complex system”, i.e. a system that can be further refined by identifying its components



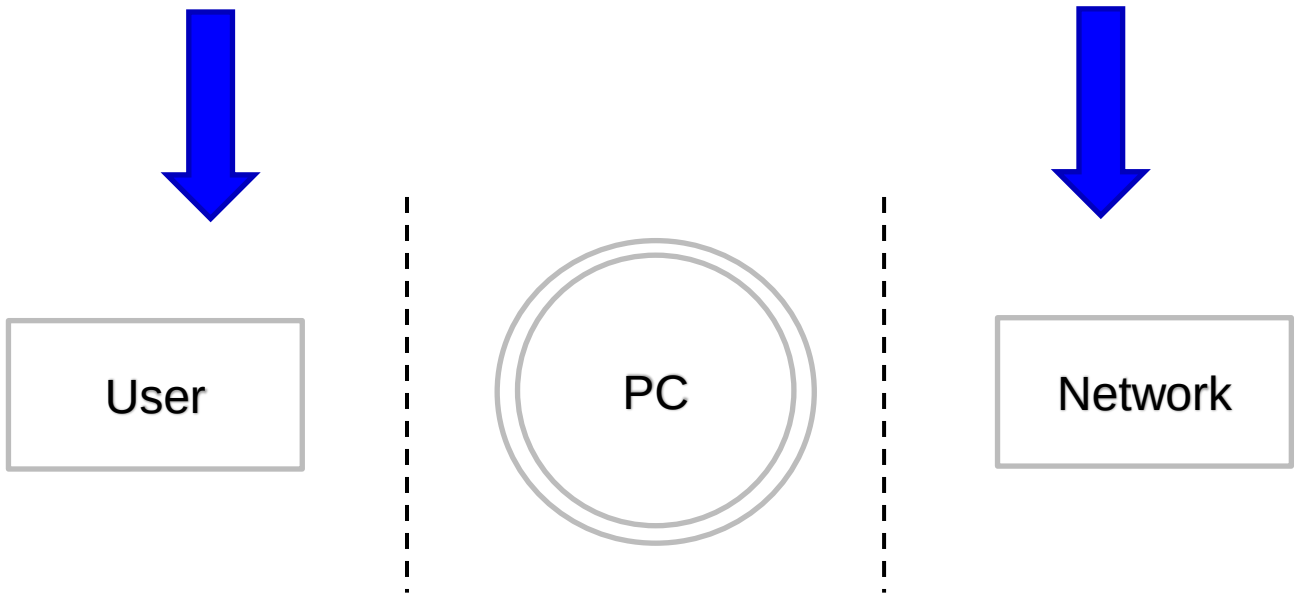
Trust boundary

The dashed lines indicate the trust boundaries. Communications among all the components of the complex system are trusted, communications coming from the outside are not trusted



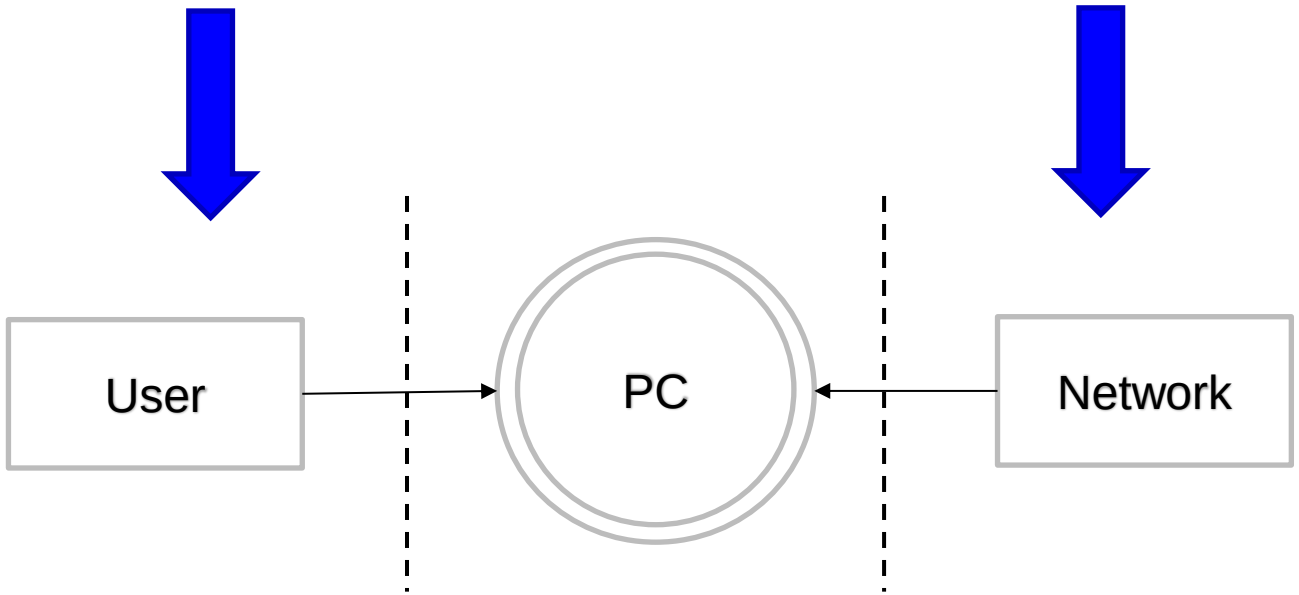
Trust boundary

Boxes indicate external components that can interact with the complex system i.e. by sending data to it



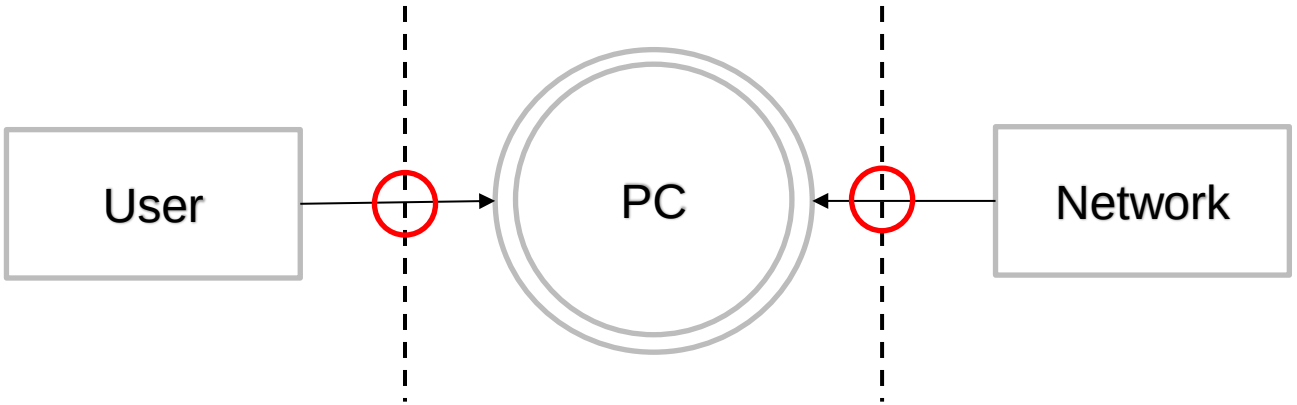
Trust boundary

Arrows represent communications/interactions



Trust boundary

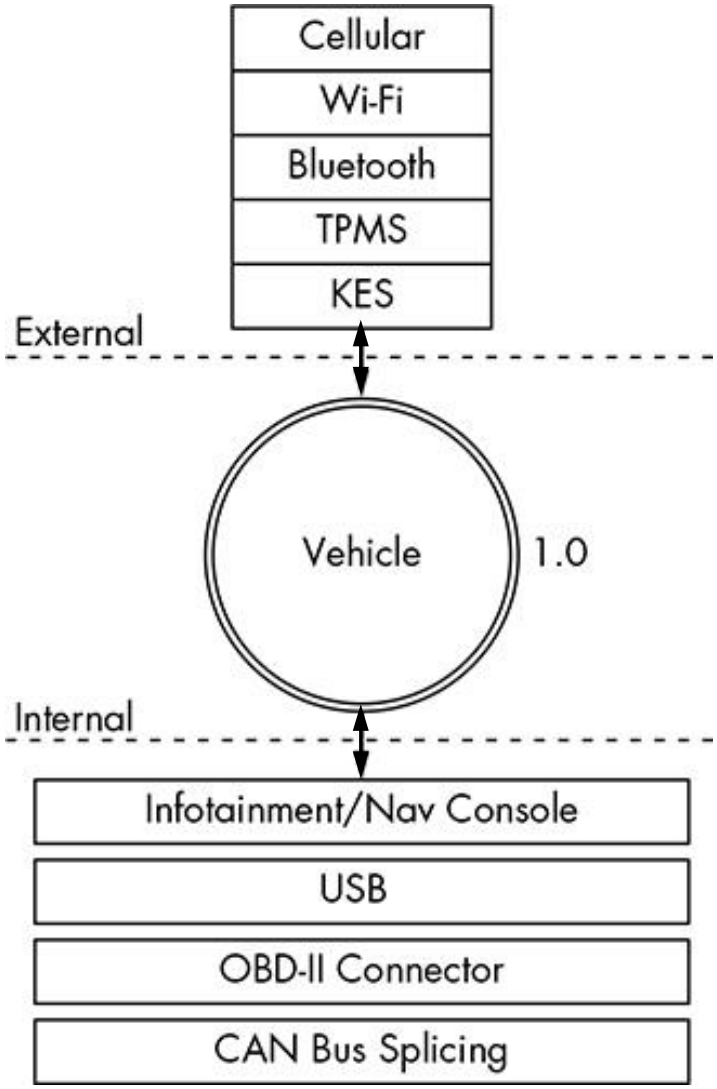
Whenever an interaction crosses a trust boundary, we have an attack vector. An explicit enumeration of all data flows and trust boundaries help in identifying required security controls



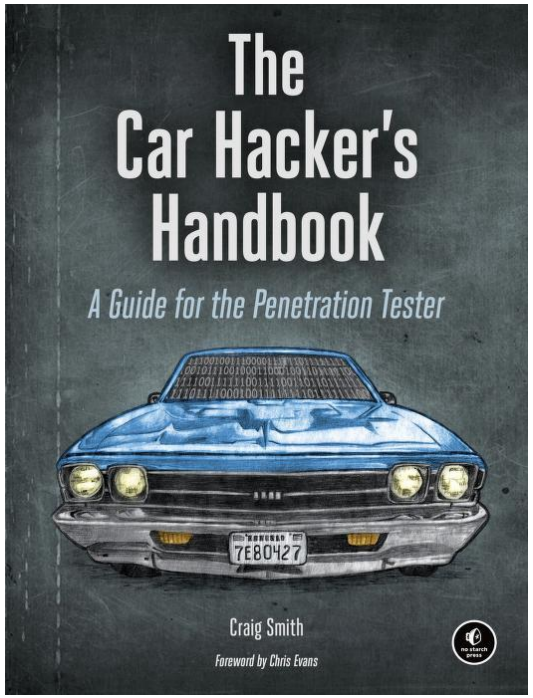
Attack surface of a vehicle

- Let us start simple:
  - Consider your entire vehicle as one complex system
  - Draw its trust boundary
  - Enumerate all possible interactions that cross the trust boundary

High level: Vehicle 1.0



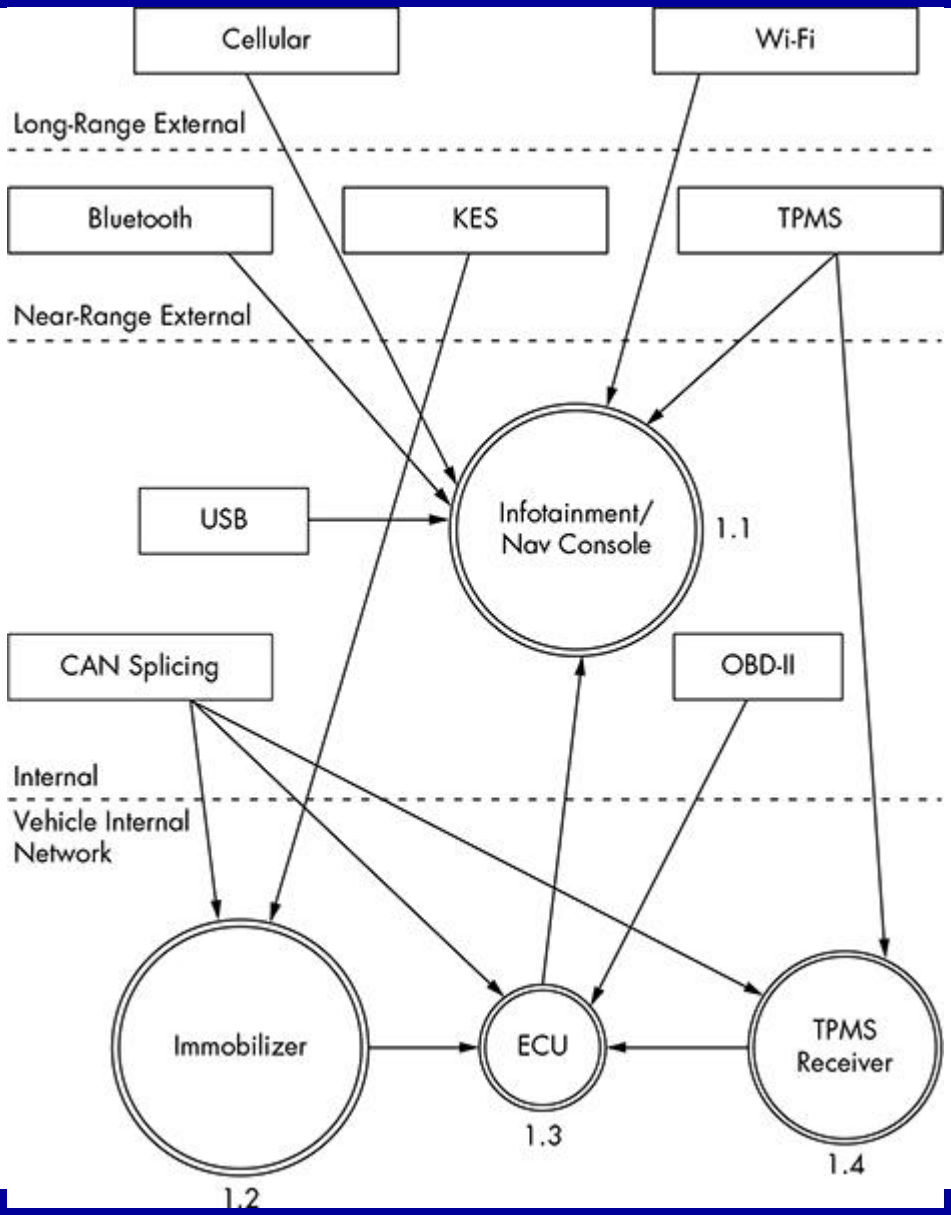
This and some of the the following pictures are from: The Car Hacker's Handbook



High-level threats

- We can already identify possible threats
  - Just high-level scenarios, no focus on the technical side
- Examples
  - Remotely take over a vehicle
  - Shut down a vehicle
  - Spy on vehicle occupants
  - Unlock a vehicle
  - Steal a vehicle
  - Track a vehicle
  - Thwart safety systems
  - Install malware on the vehicle
  - …

Drill down (1)



- Drill down by splitting the complex system into multiple components
- Components are still complex systems
- Attack vectors may cross multiple boundaries

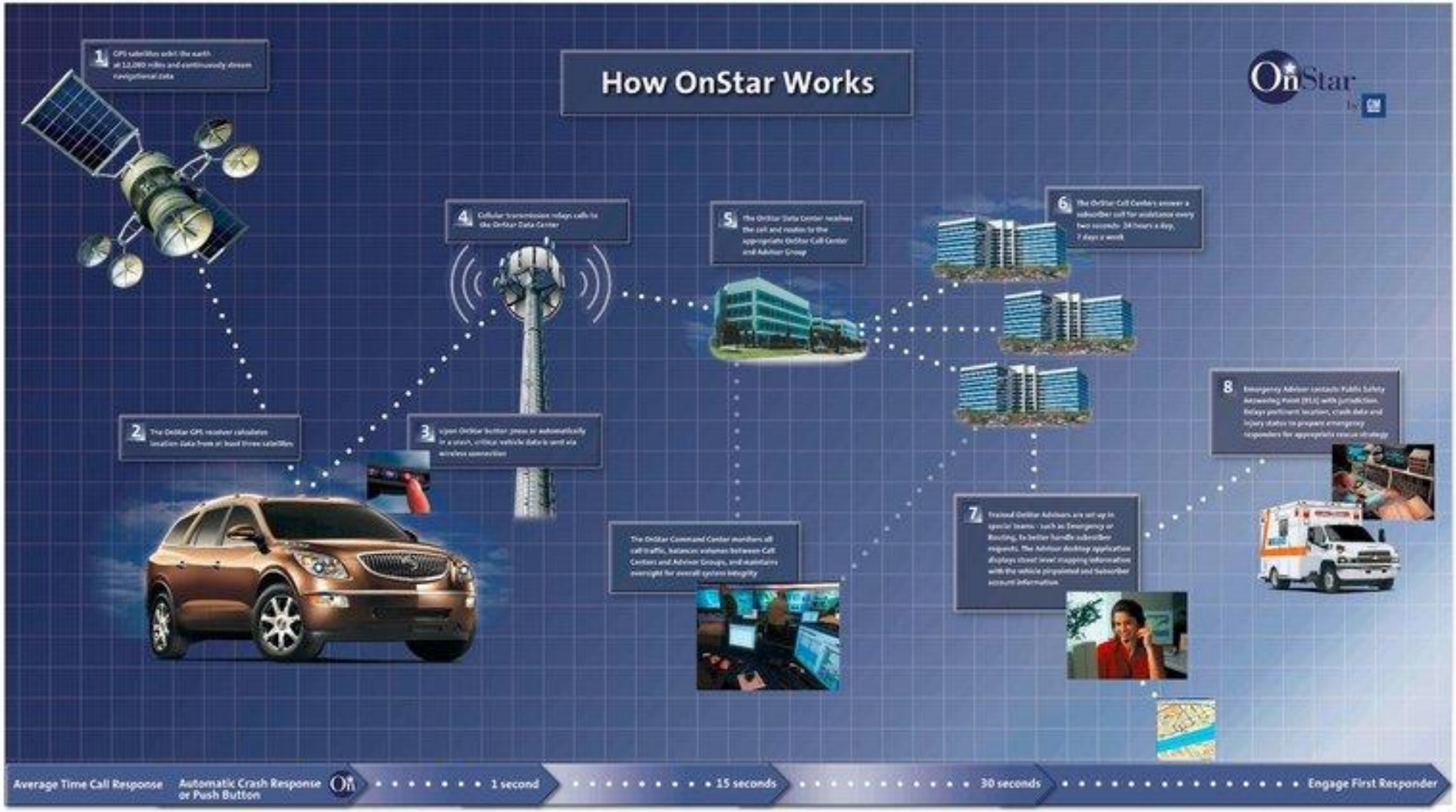
High-level threats

- We will now skim through these attack vector and bring some relevant examples of past attacks/security incidents
- Threats classified based on the attack vector
  - Cellular 
  - Wi-Fi
  - KES
  - TPMS
  - Infotainment
  - USB
  - Bluetooth
  - CAN

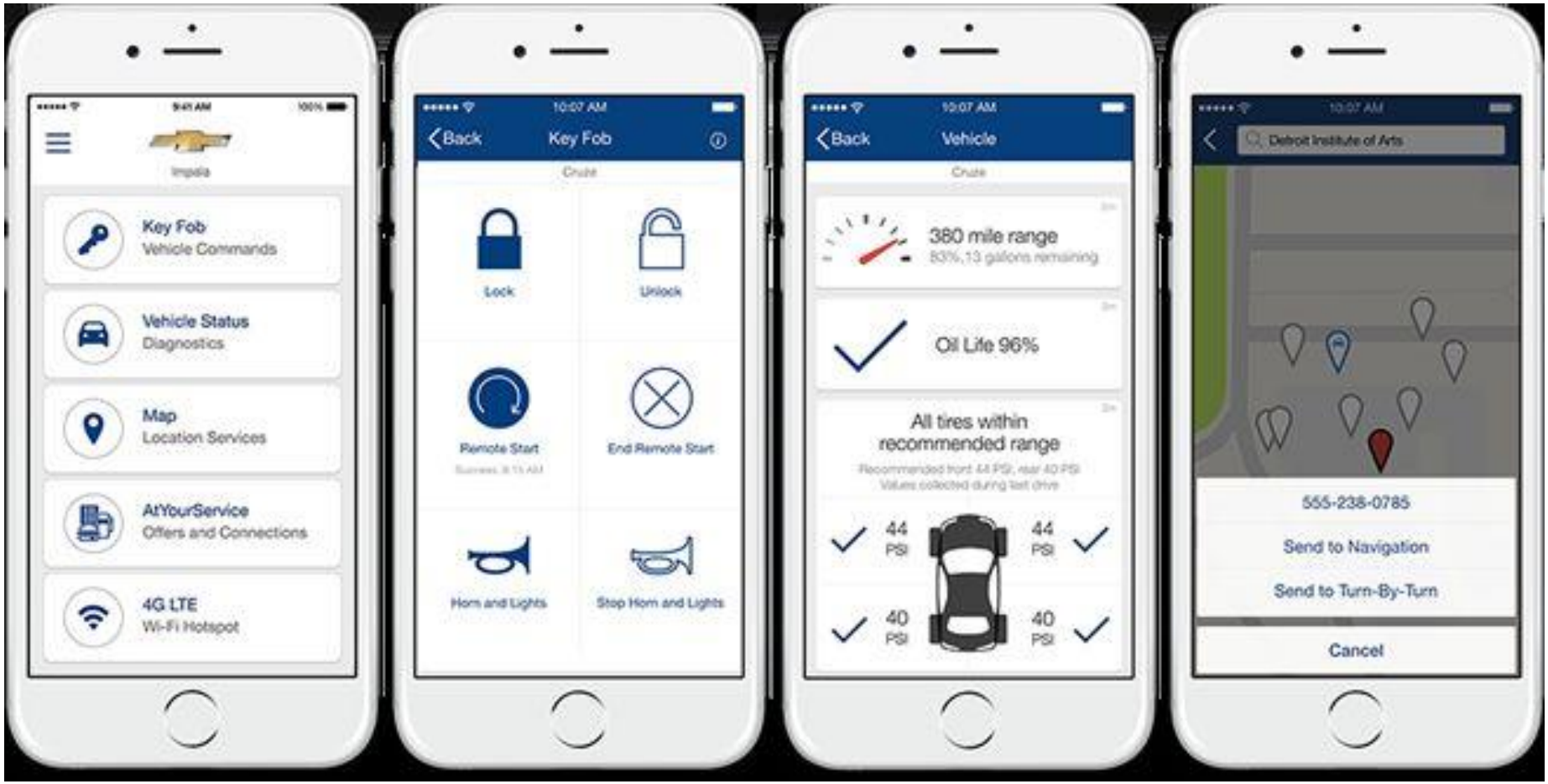
Cellular

- Access the internal vehicle network from anywhere
- Exploit the application in the infotainment unit that handles incoming calls
- Access the subscriber identity module (SIM) through the infotainment unit
- Use a cellular network to connect to the remote diagnostic system (OnStar)
- Eavesdrop on cellular communications
- Jam distress calls
- Track the vehicle’s movements
- Set up a fake Global System for Mobile Communications (GSM) base station

(… OnStar …)


How OnStar Works
OnStar
1
2
3
4
5
6
7
8
Average Time Call Response
Automatic Crash Response or Push Button
1 second
15 seconds
30 seconds
Engage First Responder

(… OnStar …)


Impala
Key Fob
Vehicle Commands
Vehicle Status
Diagnostics
Map
Location Services
AtYourService
Offers and Connections
4G LTE
Wi-Fi Hotspot
Back
Key Fob
Cruze
Lock
Unlock
Remote Start
End Remote Start
Horn and Lights
Stop Horn and Lights
Back
Vehicle
Cruze
380 mile range
Oil Life 96%
All tires within recommended range
44 PSI
44 PSI
40 PSI
40 PSI
Detroit Institute of Arts
555-238-0785
Send to Navigation
Send to Turn-By-Turn
Cancel

(… OnStar …)


On
GM
48°
11:26
AUDIO
PHONE
NAV
PANDORA
WEATHER
OnStar
Button To Exit

(… OnStar …)

(… OnStar …)

The 29-year-old hacker who was able to take control over GM cars tells us how easy it was to pull off

Cadie Thompson Jul. 30, 2015, 5:19 PM

It only took a few days and $100 for Sammy Kamkar to create a device that can take over any GM vehicle that has the OnStar system.

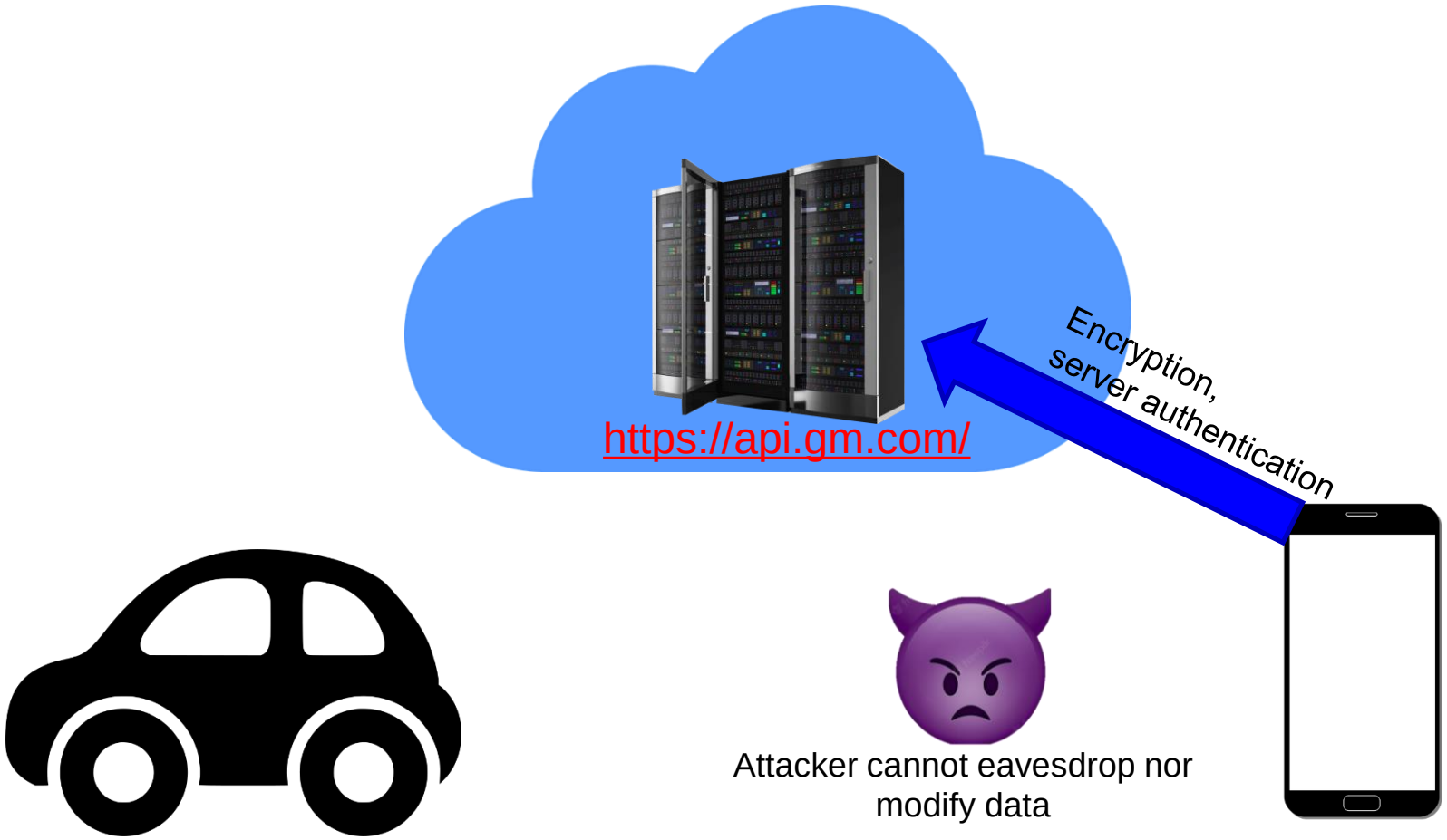
The 29-year-old software

https://www.businessinsider.com/gm-onstar-hacker-reveals-just-how-easy-it-was-to-attack-car-2015-7

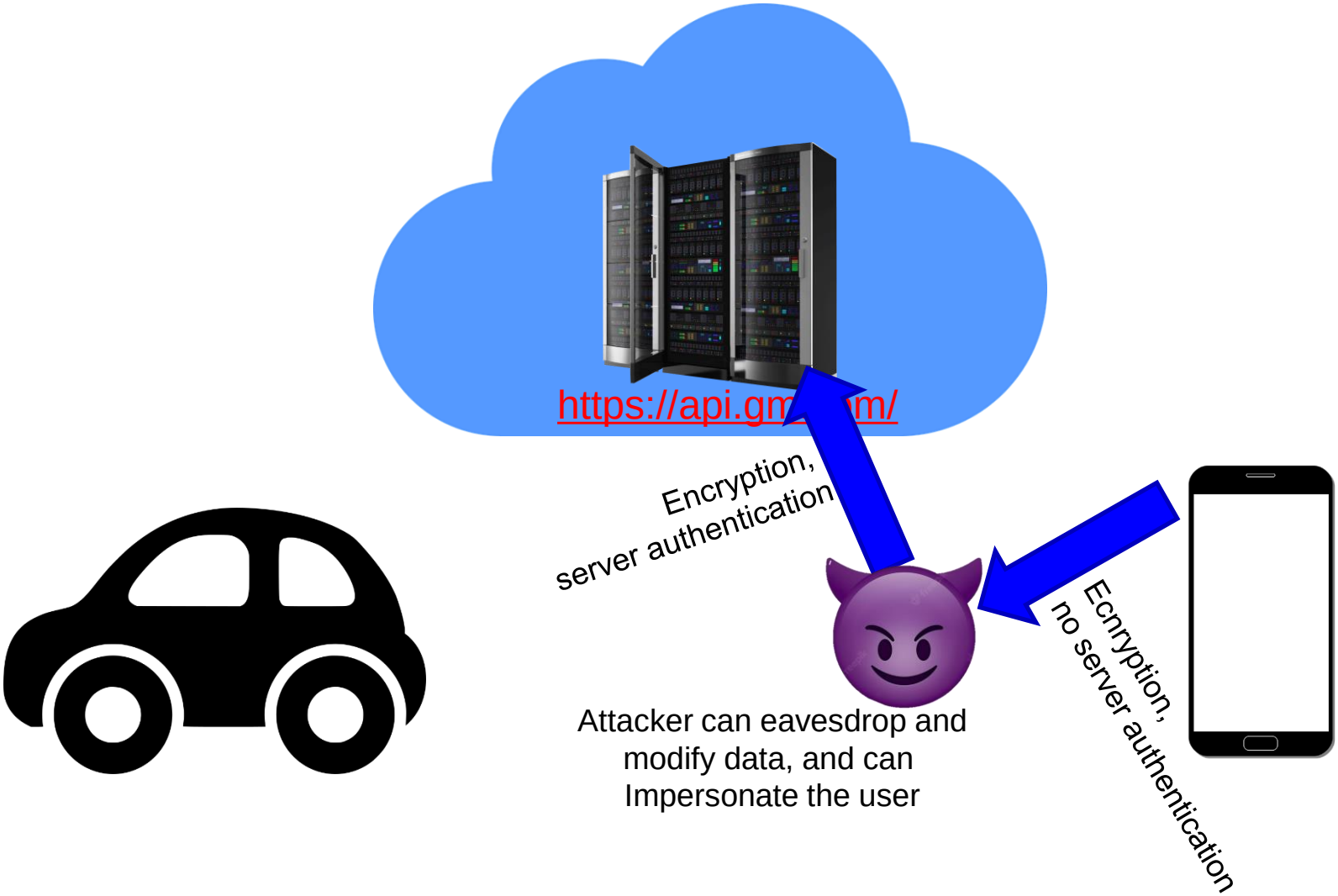
What happened here?

- Install the app on your phone and analyze traffic…
  - The endpoint api is https
    https://api.gm.com
    can only sniff encrypted traffic
- Try to perform a Man in the Middle (MitM)
  - No chance MitM will work on https…
  - … unless the application does **not** verify the digital certificate. Which is exactly what happened on the iOS App

Normal workflow



Attack workflow



From Samy Kamkar presentation at DEFCON 23

```
# https://api.gm.com/api/v1/oauth/token
POST /api/v1/oauth/token HTTP/1.1
Host: api.gm.com
Content-Type: text/plain
Connection: keep-alive
Proxy-Connection: keep-alive
Accept: application/json
User-Agent: RemoteLink/1022 CFNetwork/711.1.16 Darwin/14.0.0
Accept-Language: en
Accept-Encoding: gzip, deflate
Content-Length: 422

eyJ0eXAiOiJKV1MiLCJhbGciOiJIUzI1NiJ9.eyJwYXNzd29yZCI6InRlc3RwYXNzIiwiZGV
2aWNlX2lkIjoiMDdBNTE2NkItNjE4Mi00NTBGLUJCMTQtQzY0MkU5MkZFMkVCIiwic2NvcGU
iOiJwcml2IG1zc28iLCJncmFudF90eXBlIjoicGFzc3dvcmQiLCJ1c2VybmFtZSI6InRlc3R
1c2VyIiwidGltZXN0YW1wIjoiMjAxNS0wNy0yNFQyMzoxODoxNy43NzlaIiwiY2xpZW50X2l
kIjoiUkxfaU9TLWk3OF8yMDMiLCJub25jZSI6IjM3Qzg5Q0E4LTM5RUUtNDM2NS1CQjA3LUU
zQzU1REUyNUIyMyJ9.3C0rc8RUuHwQBDylnswoSPDE9QeUfvDZEJjstjXkzko=
```

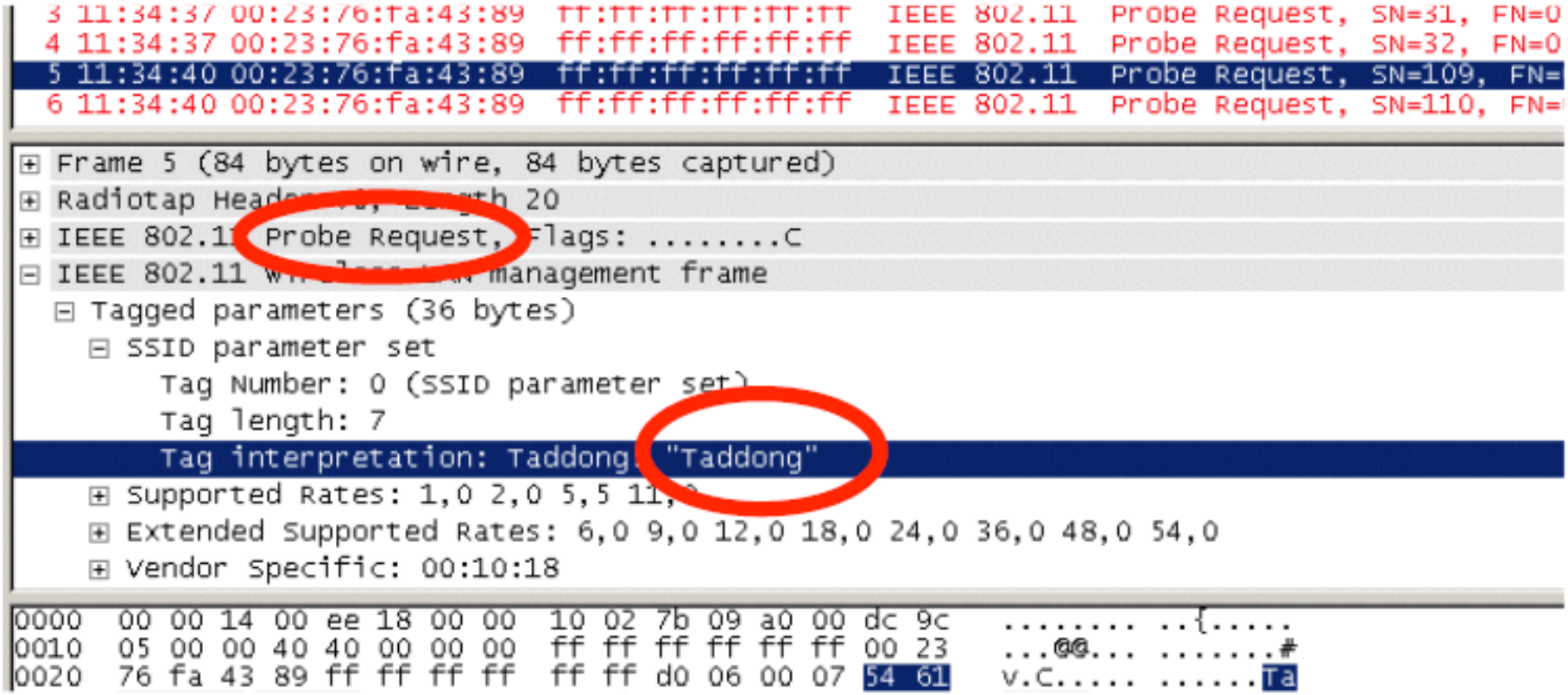
From Samy Kamkar presentation at DEFCON 23


```
{
    "typ": "JWS",
    "alg": "HS256"
} {
    "password": "testpass",
    "device_id": "07A5166B-6182-450F-BB14-C642E92FE2EB",
    "scope": "priv msso",
    "grant_type": "password",
    "username": "testuser",
    "timestamp": "2015-07-24T23:18:17.779Z",
    "client_id": "RL_iOS-i78_203",
    "nonce": "37C89CA8-39EE-4365-BB07-E3C55DE25B23"
}
```

So… how to exploit that?

- STEP 1: place a device nearby the car that spoofs a SSID known to the mobile phone of your victim (like UNIMORE, can be generated on-demand by sniffing at wifi probes)

WiFi probes

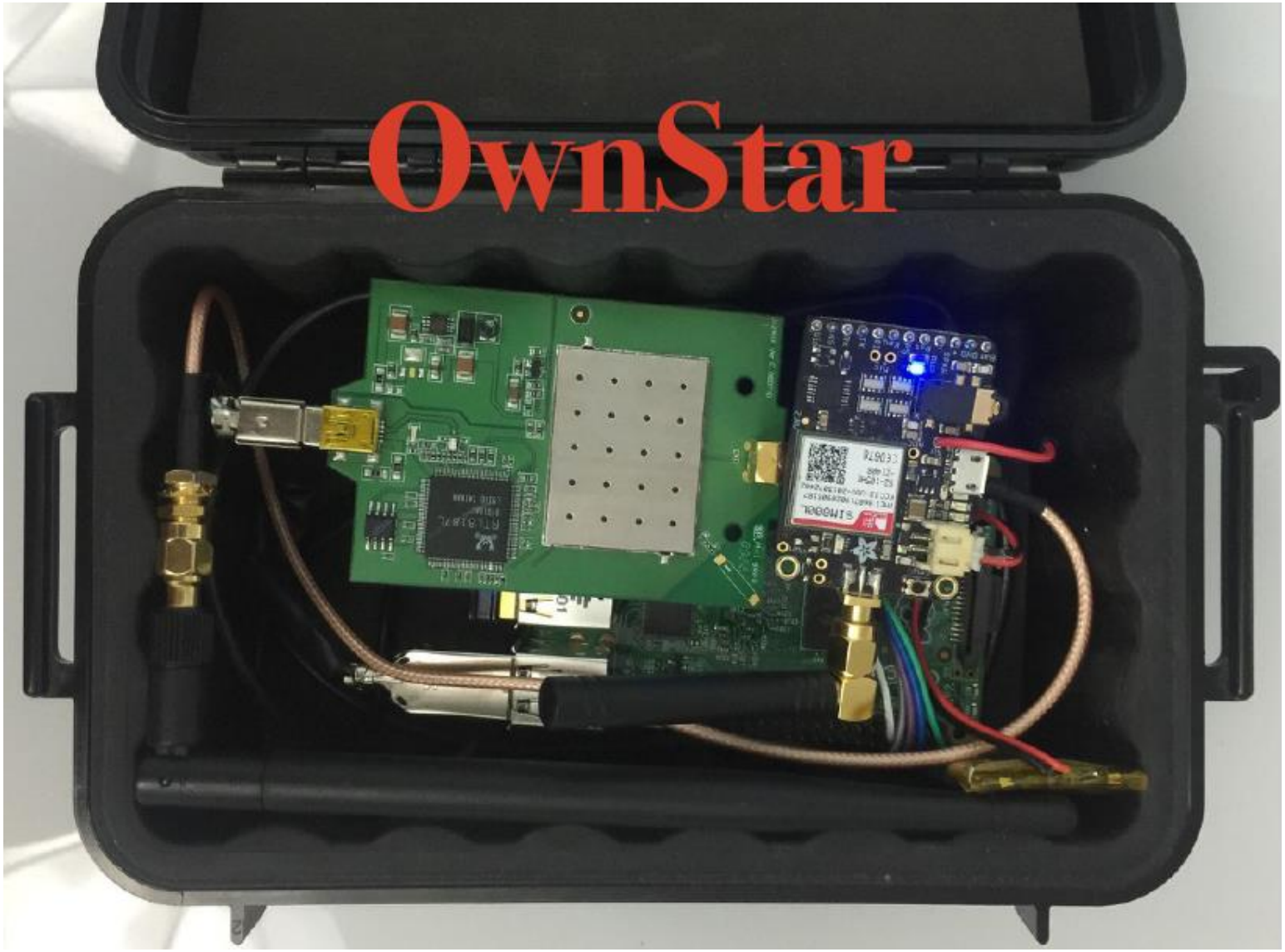

3 11:34:37 00:23:76:fa:43:89 ff:ff:ff:ff:ff:ff IEEE 802.11 Probe Request, SN=31, FN=0
4 11:34:37 00:23:76:fa:43:89 ff:ff:ff:ff:ff:ff IEEE 802.11 Probe Request, SN=32, FN=0
5 11:34:40 00:23:76:fa:43:89 ff:ff:ff:ff:ff:ff IEEE 802.11 Probe Request, SN=109, FN=
6 11:34:40 00:23:76:fa:43:89 ff:ff:ff:ff:ff:ff IEEE 802.11 Probe Request, SN=110, FN=
Frame 5 (84 bytes on wire, 84 bytes captured)
Radiotap Header
IEEE 802.11 Probe Request, Flags: C
IEEE 802.11 management frame
Tagged parameters (36 bytes)
SSID parameter set
Tag Number: 0 (SSID parameter set)
Tag length: 7
Tag interpretation: Taddong "Taddong"
Supported Rates: 1,0 2,0 5,5 11,0
Extended Supported Rates: 6,0 9,0 12,0 18,0 24,0 36,0 48,0 54,0
Vendor Specific: 00:10:18
0000 00 00 14 00 ee 18 00 00 10 02 7b 09 a0 00 dc 9c
0010 05 00 00 40 40 00 00 00 ff ff ff ff ff ff 00 23
0020 76 fa 43 89 ff ff ff ff ff ff d0 06 00 07 54 61

So… how to exploit that?

- STEP 1: place a device nearby the car that spoofs a SSID known to the mobile phone of your victim (like UNIMORE, can be generated on-demand by sniffing at wifi probes)
- STEP 2: wait for the car owner to get back. His phone will lock to the known SSID and use your device as its hotspot
- STEP 3: the device waits for your phone to send a DNS request for api.gm.com
- STEP 4: the device replies with the address of a fake server that you use as MitM (may be deployed on the same device)
- STEP 5: your App sends the authentication request to your MitM server. Does not check certificate, so everything looks fine… and now you have username and password
- STEP 6: at your leisure, log in to the real api.gm.com using stolen credentials, get owner's PII, locate the car, unlock it, start its engine, drive away…

Device: components…

Device: put together



Samy Kamkar presentation

- Pdf slides
  - https://samy.pl/defcon2015/2015-defcon.pdf
- Video of the talk
  - https://www.youtube.com/watch?v=UNgvShN4USU

(… OnStar …)

GM » Shanghai Onstar : Security Vulnerabilities

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9

Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending

Copy Results Download Results

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2017-12697	200		+Info	2018-01-09	2018-02-01	4.3	None	Remote	Medium	Not required	Partial	None	None
A Man-in-the-Middle issue was discovered in General Motors (GM) and Shanghai OnStar (SOS) SOS iOS Client 7.1. Successful exploitation of this vulnerability may allow an attacker to intercept sensitive information when the client connects to the server.														
2	CVE-2017-12695	287			2018-01-09	2018-02-01	4.0	None	Remote	Low	Single system	None	Partial	None
An Improper Authentication issue was discovered in General Motors (GM) and Shanghai OnStar (SOS) SOS iOS Client 7.1. Successful exploitation of this vulnerability may allow an attacker to subvert security mechanisms and reset a user account password.														
3	CVE-2017-9663	200		+Info	2018-01-09	2018-02-01	5.0	None	Remote	Low	Not required	Partial	None	None
An Cleartext Storage of Sensitive Information issue was discovered in General Motors (GM) and Shanghai OnStar (SOS) SOS iOS Client 7.1. Successful exploitation of this vulnerability may allow a remote attacker to access an encryption key that is stored in cleartext in memory.														

https://www.cvedetails.com/vulnerability-list/vendor_id-17514/product_id-42999/GM-Shanghai-Onstar.html

Another example: 2014 Jeep Cherokee

The cellular connectivity is made possible by a Sierra Wireless AirPrime AR5550, which can be seen below.

Another example: 2014 Jeep Cherokee

Looking at the network configuration of the Uconnect system we can see that it has several interfaces used for communications. It has an interface for the internal Wi-Fi communications, uap0, and another PPP interface, ppp0, presumably used to communicate with the outside world, via Sprint's 3G services.

```
# ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 33192
        inet 127.0.0.1 netmask 0xff000000
pflog0: flags=100<PROMISC> mtu 33192
uap0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        address: 30:14:4a:ee:a6:f8
        media: <unknown type> autoselect
        inet 192.168.5.1 netmask 0xffffff00 broadcast 192.168.5.255
ppp0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1472
        inet 21.28.103.144 -> 68.28.89.85 netmask 0xff000000
```

The 192.168.5.1 address is the address of the Uconnect system to any hosts connected to the Wi-Fi access point. The IP address 68.28.89.85 is the one that anyone on the Internet would see if the Uconnect system connected to them. However, port 6667 is not open at that address. The 21.28.103.144 address is the actual address of the interface of the Uconnect facing the Internet, but is only available internally to the Sprint network.

Another example: 2014 Jeep Cherokee

Even more shocking to us that connectivity was not limited to individual towers or segments. It turns out that any Sprint device anywhere in the country can communicate with any other Sprint device anywhere in the country. For example, below is a session of Chris in Pittsburgh verifying he can access the D-Bus port of the Jeep in St. Louis.

```
$ telnet 21.28.103.144 6667

Trying 21.28.103.144...

Connected to 21.28.103.144.

Escape character is '^]'.

a

ERROR "Unknown command"
```

Another example: 2014 Jeep Cherokee

To find vulnerable vehicles you just need to scan on port 6667 from a Sprint device on the IP addresses 21.0.0.0/8 and 25.0.0.0/8. Anything that responds is a vulnerable Uconnect system (or an IRC server). To know for sure, you can try to telnet to the device and look for the ERROR "Unknown command" string.

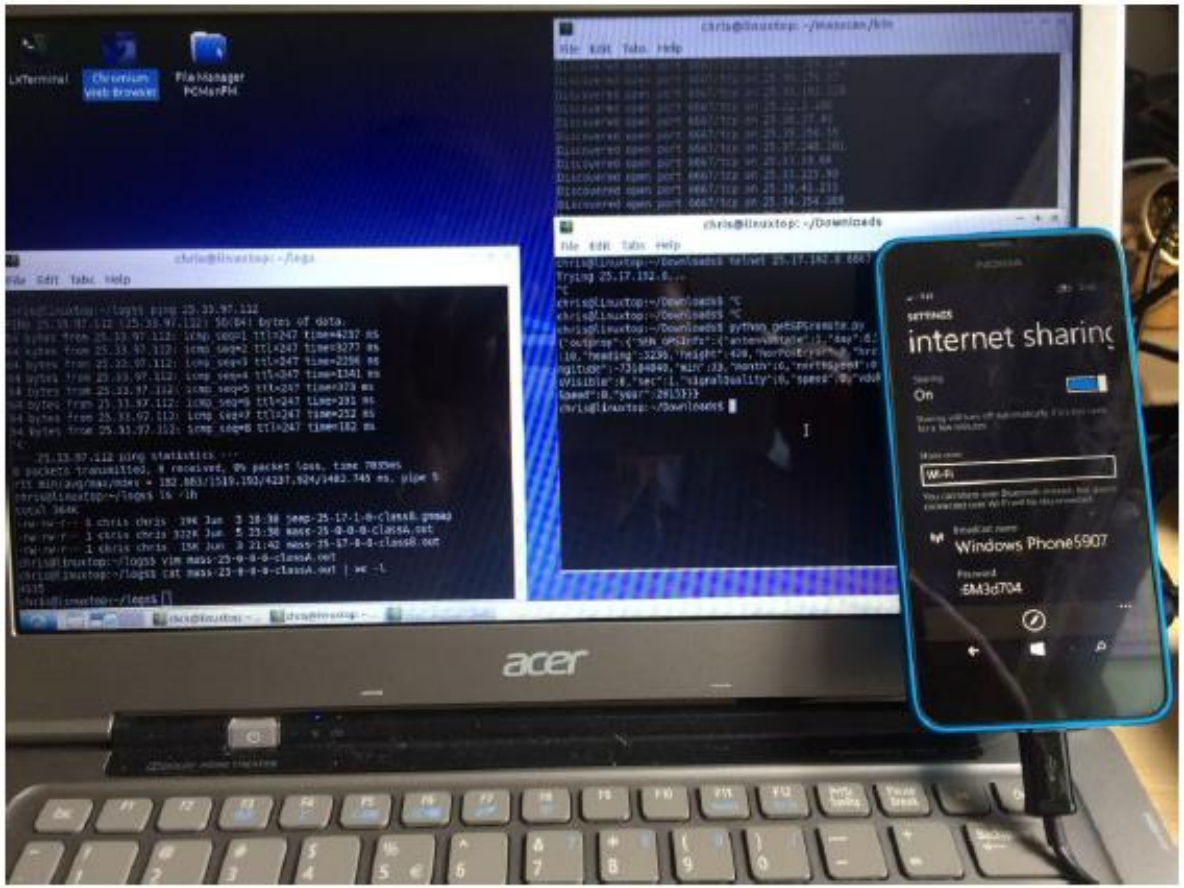


Figure: Scanning setup

Many other examples…

The following is a list of vehicles observed during scanning that seem vulnerable:

2013 DODGE VIPER
2013 RAM 1500
2013 RAM 2500
2013 RAM 3500
2013 RAM CHASSIS 5500
2014 DODGE DURANGO
2014 DODGE VIPER
2014 JEEP CHEROKEE
2014 JEEP GRAND CHEROKEE
2014 RAM 1500
2014 RAM 2500
2014 RAM 3500
2014 RAM CHASSIS 5500
2015 CHRYSLER 200
2015 JEEP CHEROKEE
2015 JEEP GRAND CHEROKEE

High-level threats

- Threats classified based on the attack vector
  - Cellular
  - Wi-Fi 
  - KES
  - TPMS
  - Infotainment
  - USB
  - Bluetooth
  - CAN

WiFi

- Access the vehicle network from up to 100 metres away (possibly more…)
- Find an exploit for the software that handles incoming connections
- Install malicious code on the infotainment unit
- Break the Wi-Fi password
- Set up a fake dealer access point to trick the vehicle into thinking it’s being serviced
- Intercept communications passing through the Wi-Fi network
- Track the vehicle (BSSID + MAC address of the wifi NIC

WiFi

- All standard wifi attacks (jamming, deauth, cracking)
  - Nothing car-specific there
- Insecure services exposed through wifi…

WiFi

- SSH server with weak/known credentials
  - MAZDA → jci:root or root:jci or cmu:root or root: cmu

Just google for ssh password of your car

1 WI-FI AP function of CMU is not equal WI-FI function of CMU.
Please review the post#1317.
You can find WIFI AP TOGGLE then touch it.
Display will show WIFI AP TOGGLE: ON

2 Use your laptop to scan accessible WIFI connection.
You will see cmu_xx.xx.xx.xx..... then connect it.

3 CMU has DHCP server.

4 You input ssh root@192.168.53.1
Password is jci

5 It does work on v.55

https://mazda3revolution.com/forums/2014-2018-mazda-3-skyactiv-audio-electronics/57714-infotainment-project-201.html#post1350146

WiFi

- Telnet (!?!)

```
# /tmp/telnet 10.0.0.16
Trying 10.0.0.16...
Connected to 10.0.0.16.
Escape character is '^]'.


QNX Neutrino (rcc) (ttyp0)

login: root
Password:


    ___          _ _   __  __ ___ _____
   /   |_   _  __| (_) |  \/  |_ _|  _  \
  / /| | | | |/ _  | | | |\/| || || |_)_/
 / __  | |_| | (_| | | | |  | || || |_) \
/_/  |_|__,__|\__,_|_| |_|  |_|___|_____/

/ > ls -la
total 37812
lrwxrwxrwx  1 root      root            17 Jan 01 00:49 HBpersistence -> /mnt/efs-persist/
drwxrwxrwx  2 root      root            30 Jan 01 00:00 bin
lrwxrwxrwx  1 root      root            29 Jan 01 00:49 config -> /mnt/ifs-root/usr/apps/
config
drwxrwxrwx  2 root      root            10 Feb 16  2015 dev
dr-xr-xr-x  2 root      root             0 Jan 01 00:49 eso
drwxrwxrwx  2 root      root            10 Jan 01 00:00 etc
dr-xr-xr-x  2 root      root             0 Jan 01 00:49 hbsystem
lrwxrwxrwx  1 root      root            20 Jan 01 00:49 irc -> /mnt/efs-persist/irc
drwxrwxrwx  2 root      root            20 Jan 01 00:00 lib
drwxrwxrwx  2 root      root            10 Feb 16  2015 mnt
dr-xr-xr-x  1 root      root             0 Jan 01 00:37 net
drwxrwxrwx  2 root      root            10 Jan 01 00:00 opt
dr-xr-xr-x  2 root      root      19353600 Jan 01 00:49 proc
drwxrwxrwx  2 root      root            10 Jan 01 00:00 sbin
dr-xr-xr-x  2 root      root             0 Jan 01 00:49 scripts
dr-xr-xr-x  2 root      root             0 Jan 01 00:49 srv
lrwxrwxrwx  1 root      root            10 Feb 16  2015 tmp -> /dev/shmem
drwxr-xr-x  2 root      root            10 Jan 01 00:00 usr
dr-xr-xr-x  2 root      root             0 Jan 01 00:49 var
/ >
```

BlackBerry QNX

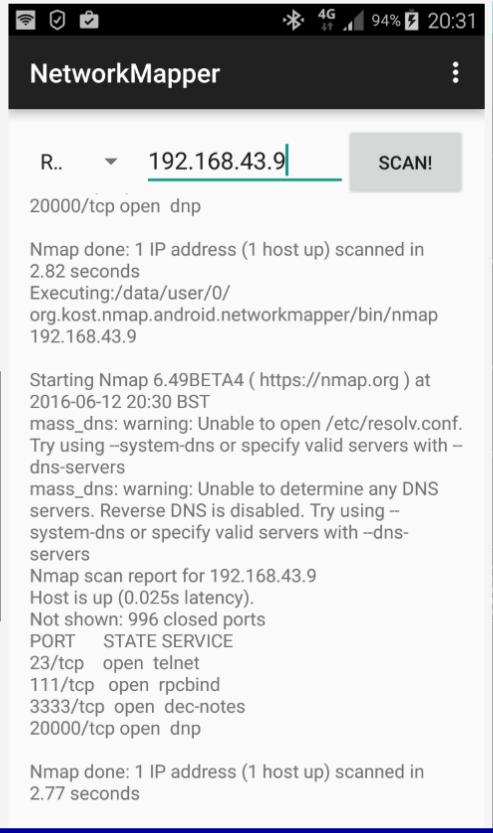
- http://blackberry.qnx.com/en/solutions/industries/automotive/index

WiFi

- Ian Tabor also showed an analysis of the IVI system within the 2015 DS5 1955 Limited Edition. He connected to the device over TCP port 23 (telnet) without any authentication and executed commands.

Having connected to the WiFi, I used NMAP to scan the IP address that was issued to the IVI unit, to the right is the screenshot of the NMAP scan.

Port	Service
23/tcp	telnet
111/tcp	rpcbind
3333/tcp	dec-notes
20000/tcp	dnp



WiFi

- Vulnerable services exposed through wifi
- Daan Keuper and Thijs Alkemade from Computest gained access to the IVI system's root account for Volkswagen and Audi: https://www.computest.nl/documents/9/The_Connected_Car._Research_Rapport_Computest_april_2018.pdf

After further research, we found a service on the Golf with an exploitable vulnerability. Initially we could use this vulnerability to read arbitrary files from disk, but quickly could expand our possibilities into full remote code execution. This attack only worked via the Wi-Fi hotspot, so the impact was limited. You have to be near the car and it must connect with the Wi-Fi network of the attacker. But we did have initial access:

```
$ ./exploit 192.168.88.253
[+] going to exploit 192.168.88.253
[+] system seems vulnerable...
[+] enjoy your shell:
uname -a
QNX mmx 6.5.0 2014/12/18-14:41:09EST nVidia_Tegra2(T30)_Boards armle
```

High-level threats

- Threats classified based on the attack vector
  - Cellular
  - Wi-Fi
  - KES 
  - TPMS
  - Infotainment
  - USB
  - Bluetooth
  - CAN

Keyless Entry Systems, Keyfobs, Immobilizer

- Send malformed key fob requests that put the vehicle's immobilizer in an unknown state.
- Actively probe an immobilizer to drain the car battery
- Drain the power from the key fob
- Lock out a key
- Capture cryptographic information leaked from the immobilizer during the handshake process → insecure protocols
- Brute-force the key fob algorithm → small key spaces
- Clone the key fob
- Jam the key fob signal → possibly just unintentional interferences https://www.theverge.com/2012/12/28/3812804/rogue-radio-station-responsible-for-keyless-entry-interference

Attack to KES

- Attacks enabled by weak crypto
  - https://www.wired.com/story/hackers-steal-tesla-model-s-seconds-key-fob/
  - https://www.youtube.com/watch?v=aVlYuPzmJoY

Keyless Entry Systems, Keyfobs, Immobilizer

- Attacks enabled by a wrong assumptions in the security model and design of KES
  - https://www.youtube.com/watch?v=bR8RrmEizVg
  - https://youtu.be/xHCUpLBGIKQ
  - https://www.manchestereveningnews.co.uk/news/greater-manchester-news/keyless-car-relay-theft-advice-14496158

Fahrzeug-hersteller	Modell	Erst-zulas-sung	Reichweite der Keyless-Verlängerung in Testhalle	Illegales Öffnen möglich?	Illegaler Motorstart möglich?
Audi	A3	10/2015	Max.	Ja	Ja
	A4	9/2015	Max.	Ja	Ja
	A6	9/2014	Max.	Ja	Ja
BMW	730d	8/2015	Max.	Ja	Ja
Citroen	DS4 CrossBack	11/2015	Max.	Ja	Ja
Ford	Galaxy	5/2014	Max.	Ja	Ja
	Eco-Sport	10/2015	Max.	Ja	Ja
Honda	HR-V	6/2015	Max.	Ja	Ja
Hyundai	Santa Fee	8/2015	Max.	Ja	Ja
KIA	Optima	11/2015	Max.	Ja	Ja
Lexus	RX 450h	12/2015	Max.	Ja	ja
RangeRover	Evoque	9/2015	Max.	Ja	ja
Renault	Traffic	11/2015	Max	Ja	Ja
Mazda	CX-5	3/2015	Max.	Ja	Ja
MINI	Clubman	8/2015	Max.	Ja	Ja
Mitsubishi	Outlander	12/2013	Max.	Ja	Ja
Nissan	Qashqai+2	11/2013	Max.	Ja	Ja
	Leaf	05/2012	Max.	Ja	Ja
Opel	Ampera	03/2012	Max.	Ja	Ja
SsangYong	Tivoli XDi	09/2015	Max.	Ja	Ja
Subaru	Levorg	8/2015	Max	Ja	Ja
Toyota	RAV4	12/2015	Max.	Ja	Ja
VW	Golf 7 GTD	10/2013	Max.	Ja	Ja
	Touran 5T	12/2015	Max.	Ja	Ja

High-level threats

- Threats classified based on the attack vector
  - Cellular
  - Wi-Fi
  - KES
  - TPMS 
  - Infotainment
  - USB
  - Bluetooth
  - CAN

Tire Pressure Monitoring System

- Send an impossible condition to the engine control unit (ECU), causing a fault that could then be exploited
- Trick the ECU into overcorrecting for spoofed road conditions
- Put the TPMS receiver or the ECU into an unrecoverable state that might cause a driver to pull over to check for a reported flat or that might even slow or shut down the vehicle (limp mode)
- Track a vehicle based on the TPMS unique IDs
- Spoof the TPMS signal to set off internal alarms

Tire Pressure Monitoring System

- Vehicle tracking from up to 40 meters with low cost equipment, remotely light the warning indicator, disable the TPMS system
  - http://www.winlab.rutgers.edu/~gruteser/xu_tpms10.pdf (just take a look at the conclusion section)
- Active TPMS also exist
  - Ability to inflate a low-pressure tire from a reserve of high-pressure air
  - Not for passenger vehicles
  - Attacks may lead to over-inflated tires

High-level threats

- Threats classified based on the attack vector
  - Cellular
  - Wi-Fi
  - KES
  - TPMS
  - Infotainment 
  - USB
  - Bluetooth
  - CAN

Infotainment

- Put the console into debug mode
- Alter diagnostic settings
- Find an input bug that causes unexpected results
- Install malware to the console
- Use a malicious application to access the internal CAN bus network
- Use a malicious application to eavesdrop on actions taken by vehicle occupants
- Use a malicious application to spoof data displayed to the user, such as the vehicle location

Infotainment

- Researchers at Dutch firm Computest have disclosed multiple vulnerabilities in the infotainment system of some Volkswagen and Audi models, allowing them to remotely access the system and commandeer the microphone, navigation system, and speakers.
  - https://www.zdnet.com/article/vw-audi-security-multiple-infotainment-flaws-could-give-attackers-remote-access/
- As a proof of concept, we have created a demonstration malicious app that exploits heap overflow vulnerabilities discovered in the implementation of MirrorLink 10 on the IVI. This vulnerability can allow attackers to gain control flow of a privileged process executing on the IVI
  - https://www.usenix.org/conference/woot16/workshop-program/presentation/mazloom
- […] Cîrlig and Tanase showed a proof-of-concept malware program—a Bash script—that when executed via USB, continuously looked for open Wi-Fi hotspots, connected to them and could exfiltrate newly collected data. By combining this malware with location data from the GPS, an attacker could also track the car in real time on a map.
  - https://motherboard.vice.com/en_us/article/3kvw8y/researchers-hack-car-infotainment-system-and-find-sensitive-user-data-inside

High-level threats

- Threats classified based on the attack vector
  - Cellular
  - Wi-Fi
  - KES
  - TPMS
  - Infotainment
  - USB 
  - Bluetooth
  - CAN

USB

- Install malware on the infotainment unit
- Exploit a flaw in the USB stack of the infotainment unit
- Attach a malicious USB device with specially crafted files designed to break importers on the infotainment unit, such as the address book and MP3 decoders
- Install modified update software on the vehicle

USB

- Owners of Mazda cars have been modding and installing apps to their infotainment using MZD-AIO-TI (MZD All In One Tweaks Installer) in the Mazda3Revolution forum since 2014.
- https://github.com/shipcod3/mazda_getInfo

High-level threats

- Threats classified based on the attack vector
  - Cellular
  - Wi-Fi
  - KES
  - TPMS
  - Infotainment
  - USB
  - Bluetooth 
  - CAN

Bluetooth

- Exploit a flaw in the Bluetooth stack of the infotainment unit
- Upload malformed information, such as a corrupted address book designed to execute code
- Jam the Bluetooth device

Bluetooth

- Example of bluetooth vulnerability in automotive system
  - https://nvd.nist.gov/vuln/detail/CVE-2017-9212
- The Car Wisperer attack, or how to connect to a car bluetooth system to spy on its occupants
  - https://www.thesecuritybuddy.com/bluetooth-security/what-is-car-whisperer/

High-level threats

- We can also refine threats specific to the Infotainment
  - Still not detailed, but related to the component that receives the input
- Attacks classified based on the input method
  - Cellular
  - Wi-Fi
  - KES
  - TPMS
  - Infotainment
  - USB
  - Bluetooth
  - CAN 

CAN

- Install a malicious diagnostic device to send packets to the CAN bus
- Plug directly into a CAN bus to attempt to start a vehicle without a key
- Plug directly into a CAN bus to upload malware
- Install a malicious diagnostic device to track the vehicle
- Install a malicious diagnostic device to enable remote communications directly to the CAN bus, making a normally internal attack now an external threat

High-level threats

- We can also refine threats specific to the Infotainment
  - Still not detailed, but related to the component that receives the input
- Attacks classified based on the input method
  - Cellular
  - Wi-Fi
  - KES
  - TPMS
  - Infotainment
  - USB
  - Bluetooth
  - CAN
  - GPS 

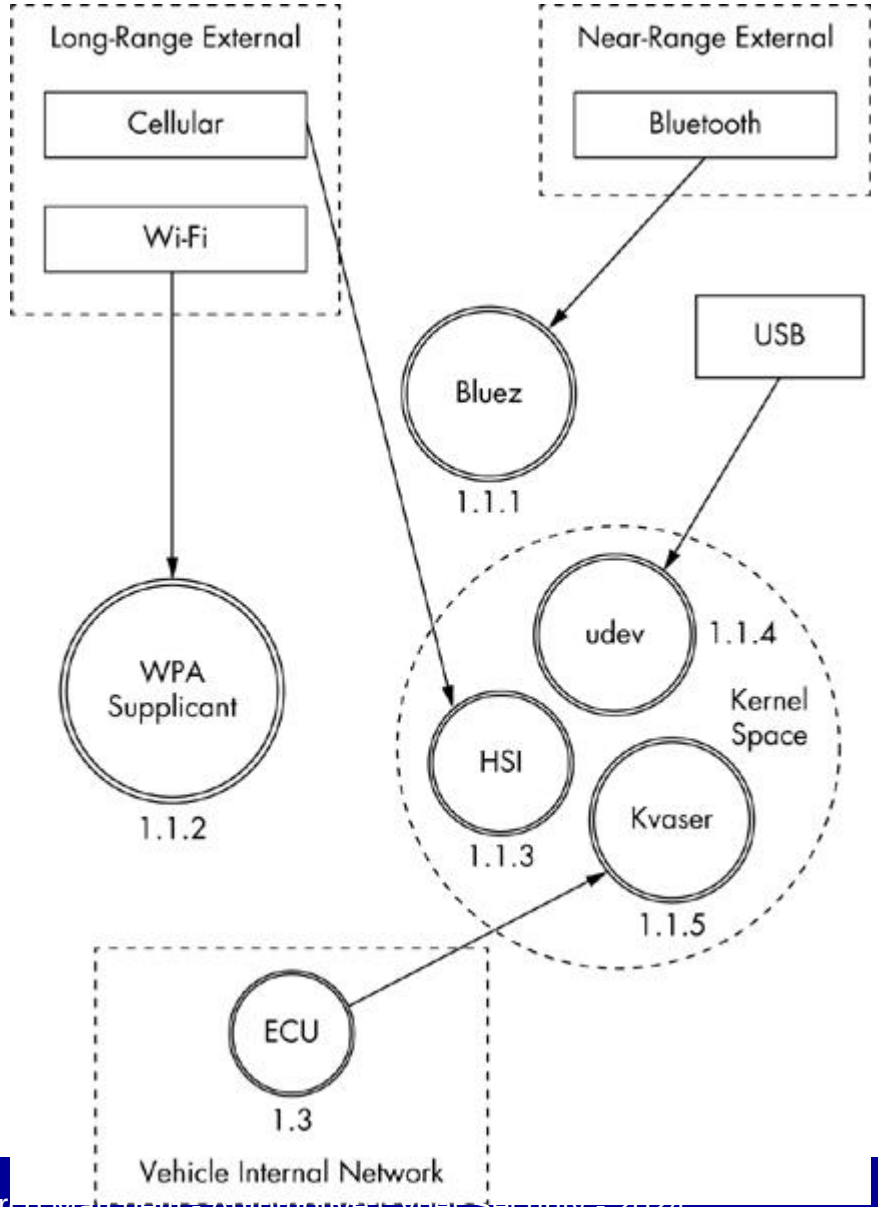
GPS

- Broadcast malformed GPS signals trying to elicit unexpected behaviors
  - Quite unlikely… well defined message formats, few inputs https://en.wikipedia.org/wiki/GPS_signals#Navigation_message
- Jam GPS signals, inhibiting geoloc abilities
- Can broadcast false geoloc information
  - Will confuse the satellite navigator, possibly leading to wrong driving instructions… not such a big deal…
  - Other consequences?

GPS

- Safety-relevant consequences are possible, depending on how your car use the GPS signal…
  - example: can interfere with OnStar or eCall systems
- Problem: brake on radar return when passing under a bridge
  - https://forums.tesla.com/forum/forums/anyone-have-tacc-hit-breaks-when-going-under-bridges
- Solution: do not brake on radar return for selected GPS coordinates
  - What could possibly go wrong?
  - https://www.reddit.com/r/teslamotors/comments/9y6zpb/another_close_call_with_autopilot_today_merging/

Drill down (2)



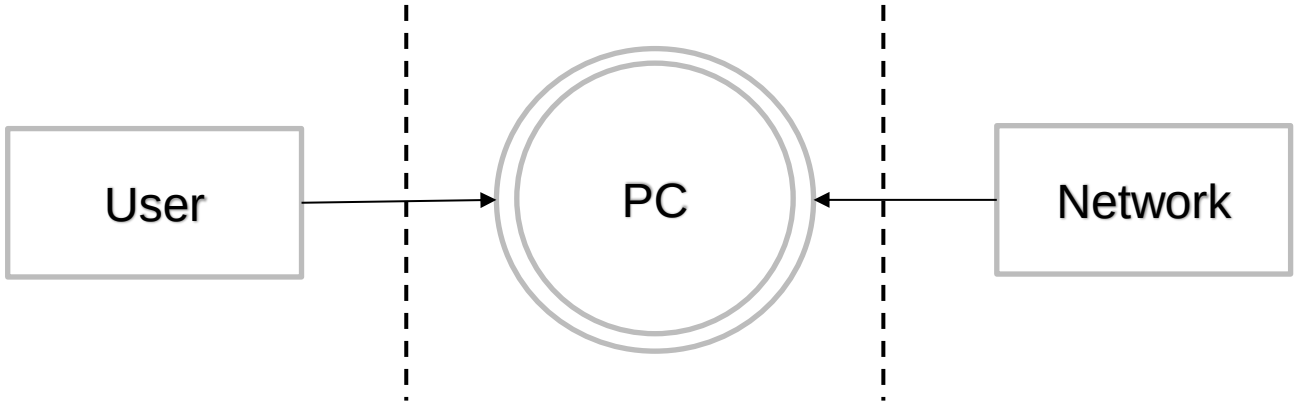
- The Infotainment system is further split into its components
- Assume the infotainment is based on embedded linux
- Note the trust boundary between kernel-space and user-space

Software specific vulnerabilities

- Identified software libraries/packages/kernel modules
  - Bluez, WPA Supplicant, udev, HSI, Kvaser
- Follow standard approach for VA/PT
  - Try to fingerprint a specific version
  - Look for known vulnerabilities, Proof of Concepts, exploits
- Examples:
  - Bluez: https://www.cvedetails.com/vulnerability-list/vendor_id-8316/Bluez.html
  - Udev: https://www.cvedetails.com/vulnerability-list/vendor_id-7630/product_id-17249/Kernel-Udev.html
  - Wpa_supplicant: https://www.cvedetails.com/vulnerability-list/vendor_id-7630/product_id-17249/Kernel-Udev.html

… ok …

- But what is this?



- It is a Data Flow Diagram!
  - This kind of diagram is commonly used to identify attack vectors within a threat modeling approach

… and there are tools for that

- Most common tool: Microsoft Threat Modeling Tool
- Download for free (as in free beer, not free software) from

https://www.microsoft.com/en-us/download/details.aspx?id=49168

- Not automotive specific

Automotive Template

- Automotive template for Microsoft Threat Modeling Tool:

https://github.com/nccgroup/The_Automotive_Threat_Modeling_Template

- Far for being complete and correct, yet useful to kickstart a threat modeling process