VIDEN

Attacker Identification on In-Vehicle Networks

Kyong-Tak Cho and Kang G. Shin

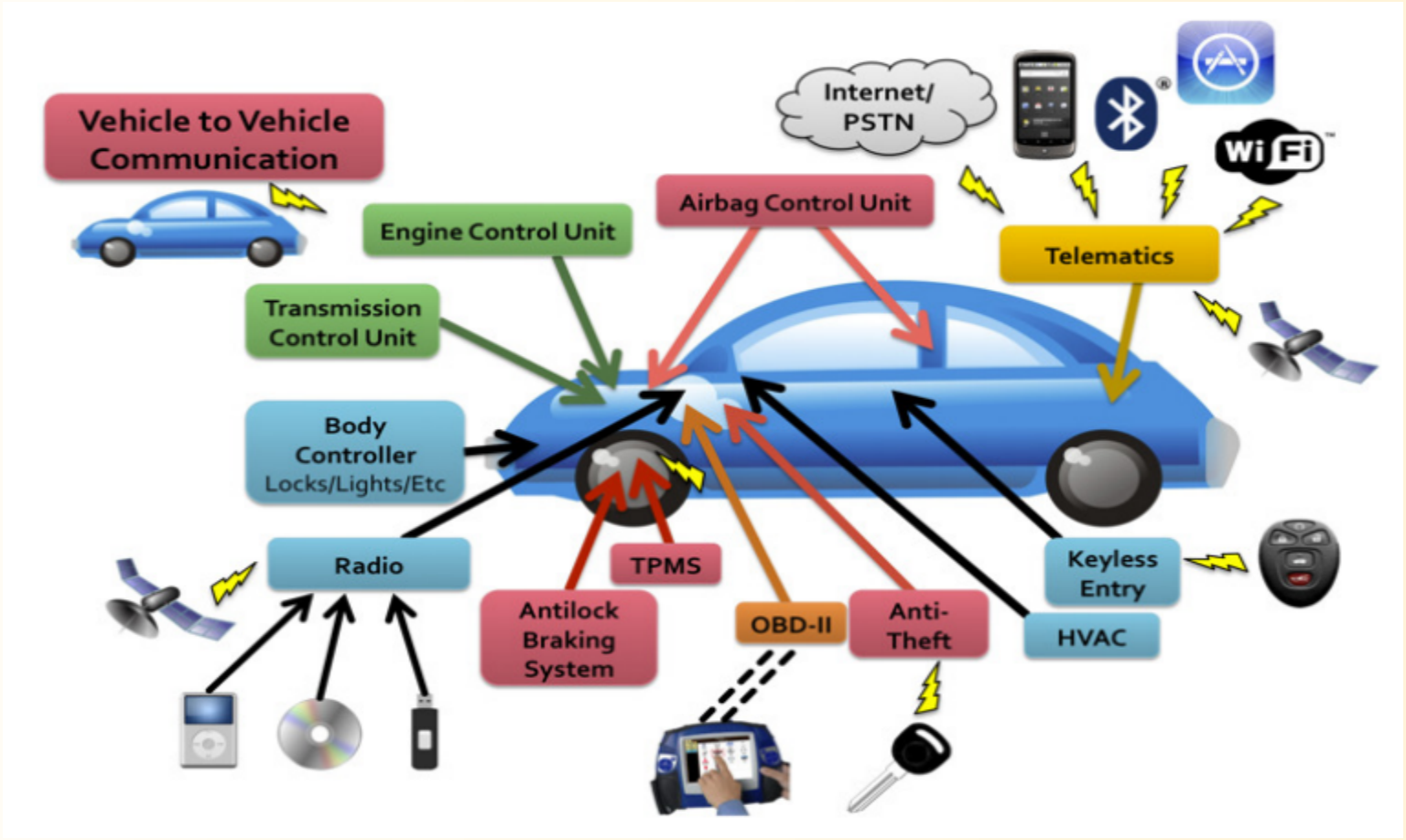
Presented by
Alokparna Bandyopadhyay
Fall 2018, Wayne State University

Overview

- Introduction
- CAN Message Transmission
- System and Threat Model
- VIDEN
- Evaluation
- Conclusion

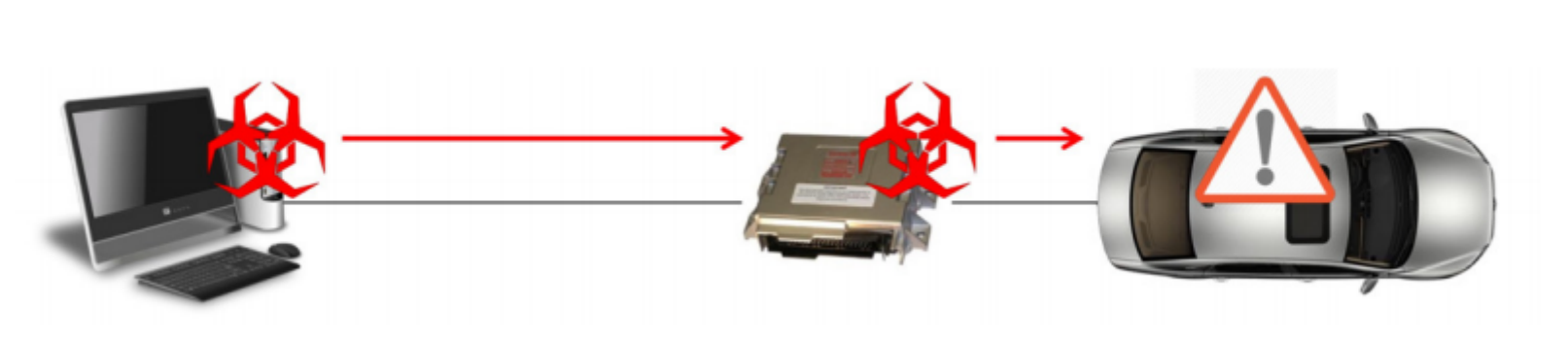
Introduction

Automotive Components of a Modern Car



Security Concerns

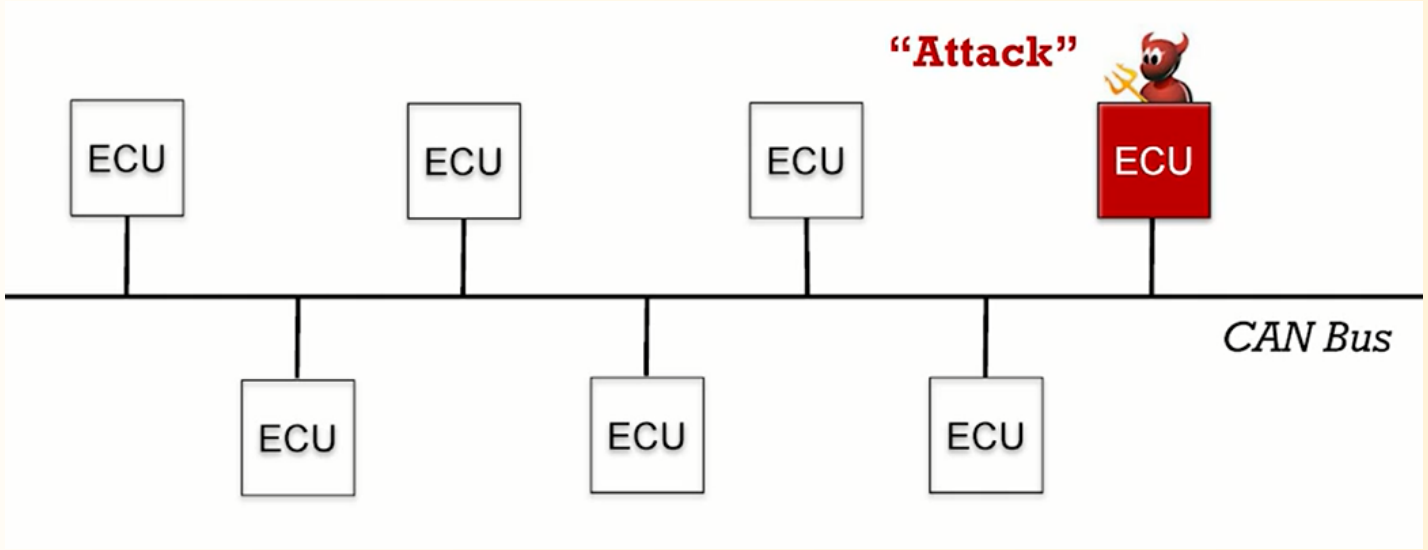
- Modern cars with remote and/or driverless control has various remote access points
  - Attackers exploit them remotely to compromise Electronic Control Units (ECUs) of a vehicle
  - Remotely control or even shut down a vehicle



Vehicle Cyber Attacks

What is a CAN Bus?

Controller Area Network Bus (CAN Bus) is an inexpensive low-speed specialized in-vehicle communication network for interconnecting the automotive components inside a vehicle



Defense against Attacks

Related Works:

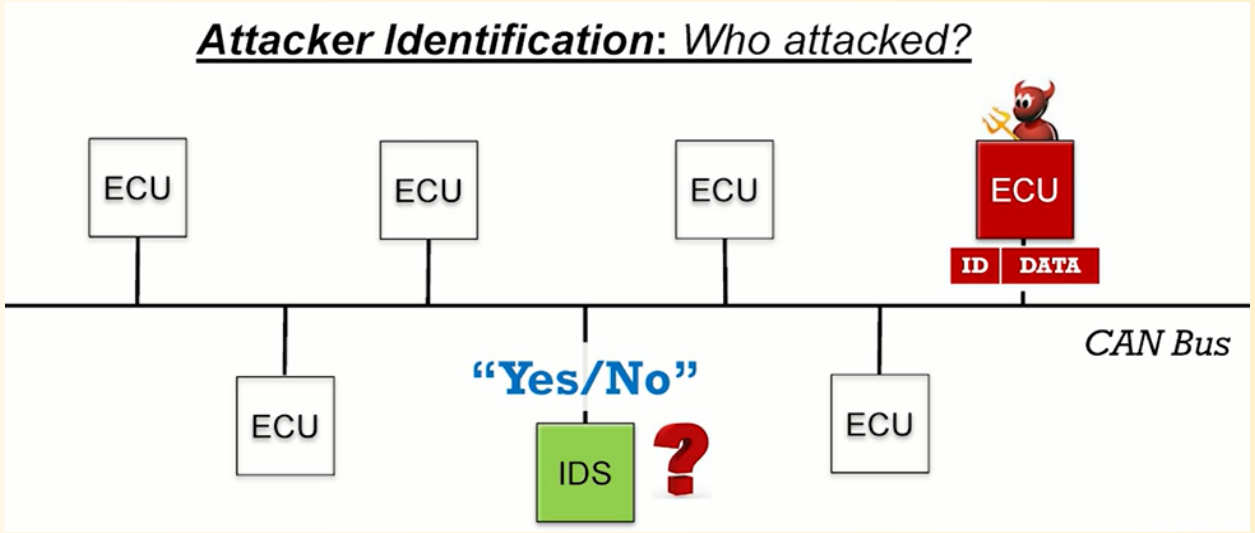
- Efficient Intrusion Detection Systems (IDS) are proposed in the past to identify presence of an attack

Problems:

- Fails to identify the attacker ECU
- Blindly treats all ECUs as (possible) attackers
- Highly expensive to patch all ECUs

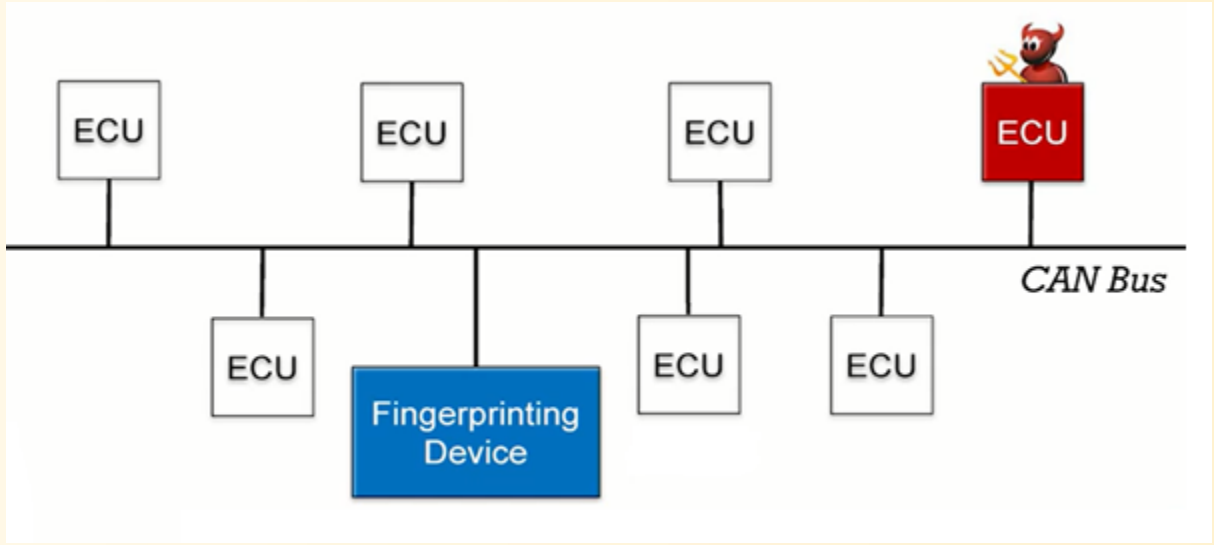
Motivation for VIDEN

- Attacker Identification is essential
  - Forensic
  - Isolation of attacker
  - Security patch on the attacker ECU
- Economical and logical approach



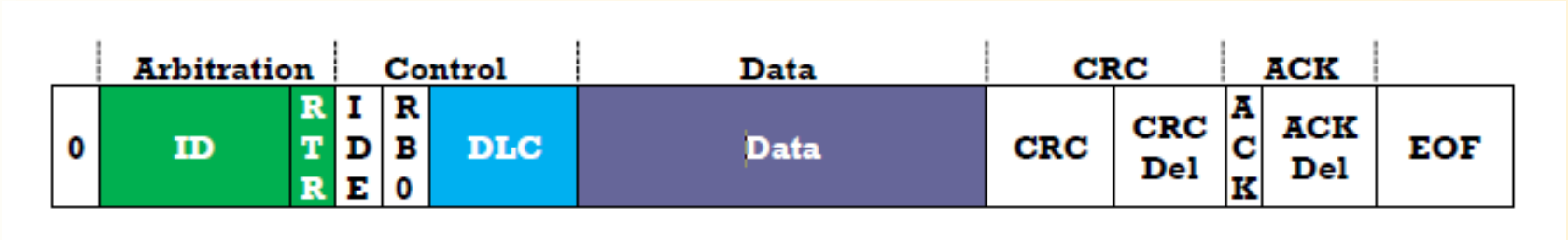
Motivation for VIDEN cont.

- Fingerprints the transmitter ECUs on CAN Bus via voltage measurements
- Uses the fingerprints for attacker identification
- Why voltage?
  - Small inherent discrepancies in voltage outputs of ECUs during message injection
  - Capture this output voltage and use it for fingerprinting



CAN Message Transmission

CAN Data Frame

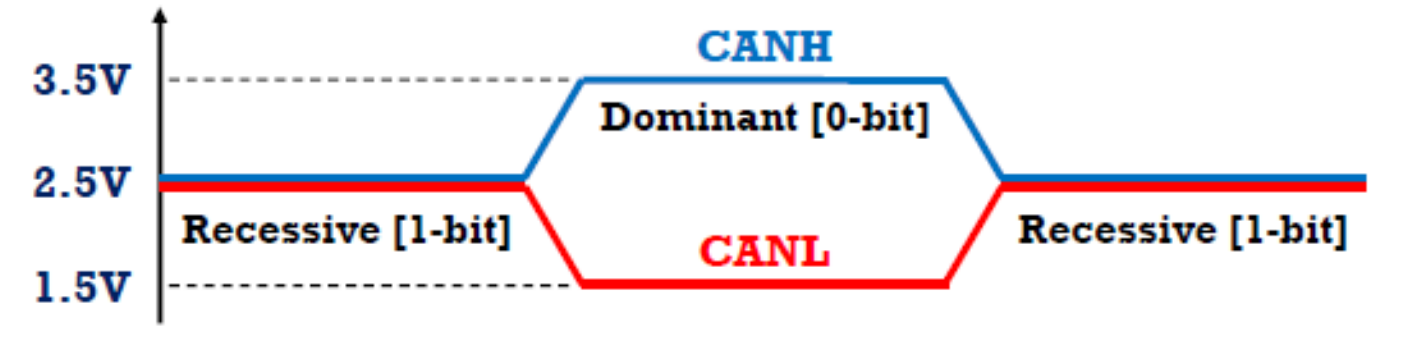


Format of a standard CAN data frame

- All fields within the CAN data frame are sent on the bus by the 'transmitter ECU' except for the Acknowledgment (ACK) slot
- ACK slot is used by all other recipient ECUs at the same time to acknowledge the transmitted message
  - 0-bit : Correctly received
  - 1-bit: Not received

Message Transmission

- CAN transceivers have two dedicated CAN wires: CAN High and CAN Low
- Agreed to output certain voltage levels at CANH and CANL
- Differential voltage determines Dominant 0-bit or Recessive 1-bit

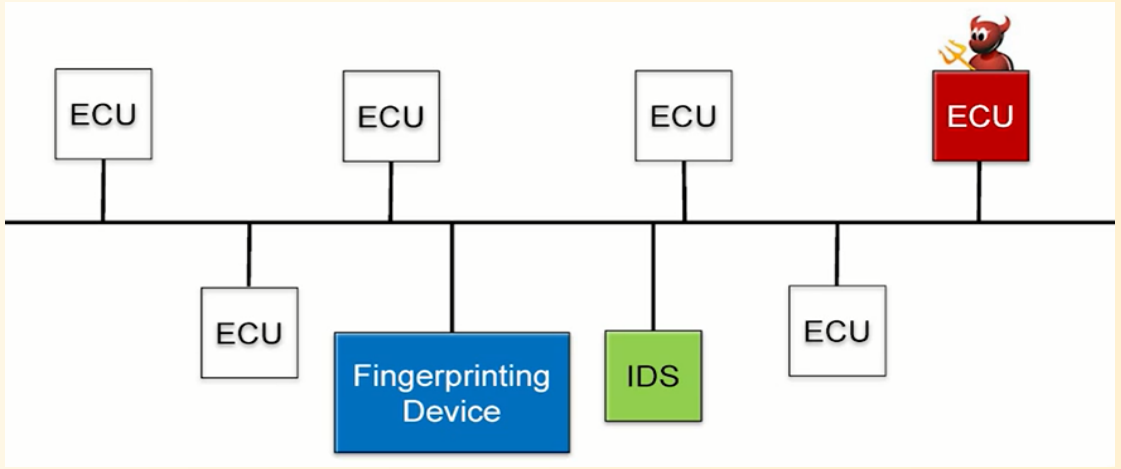


Message Transmission via Output Voltage

System and Threat Model

System Model

- In-vehicle protocol used: *CAN Bus*
- CAN bus is assumed to be equipped with:
  - *Intrusion Detection System* (IDS) :
    - Detects the presence of an attack
  - Timing and voltage-based *Fingerprinting Device*
    - Identifies the source of the (detected) attack



- System model considers only remotely compromised ECUs
  - Originally installed on the vehicle's CAN bus and remotely controlled
- Physically compromised ECUs which are later attached to the CAN bus network are not considered

Threat model

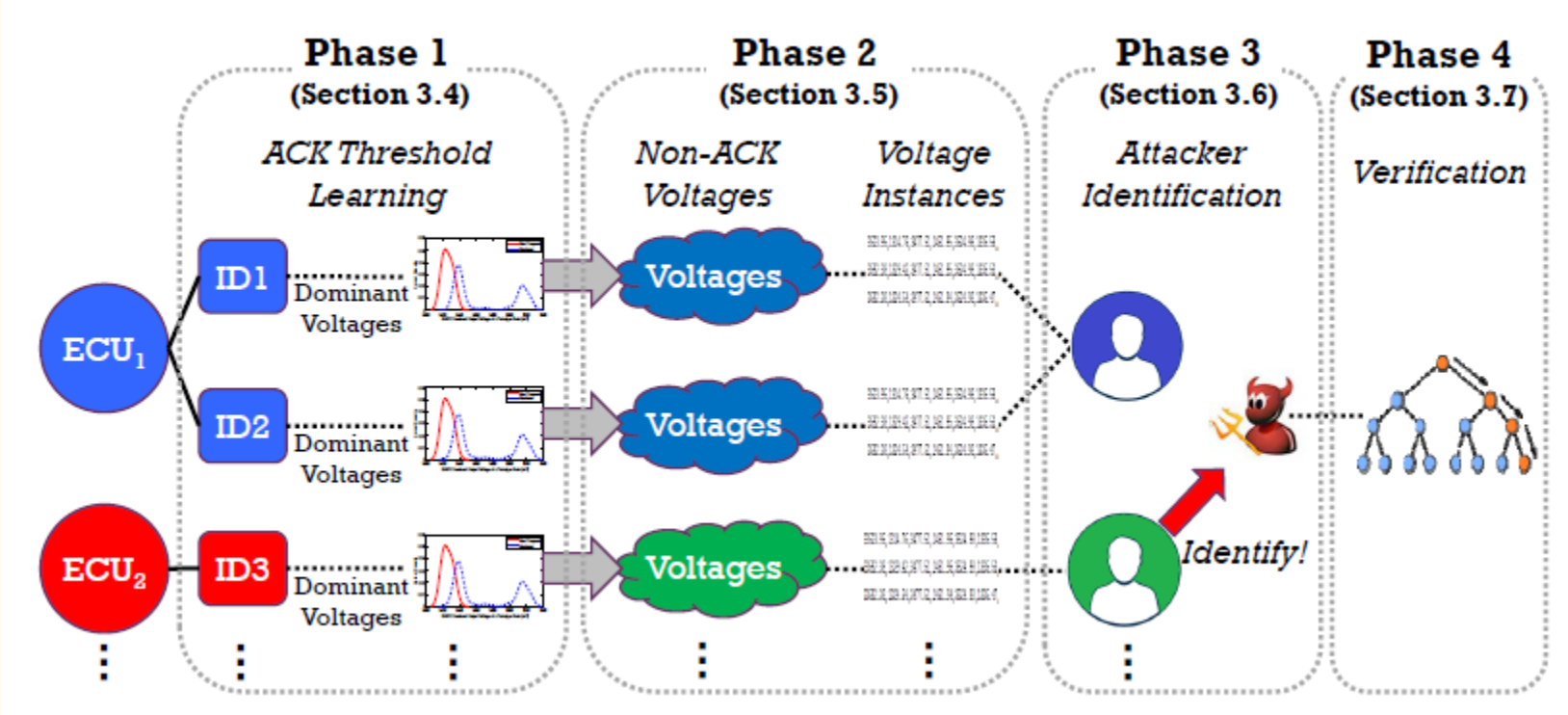
- Attacker Goal:
  - Vehicle maneuver control
  - Hide the identity of the attacker ECU
    - Evade the Fingerprinting Device
- Attacker performs impersonations when injecting attack messages
  - Arbitrary impersonation
  - Targeted impersonation
- Three types of adversaries are considered
  - Naïve
  - Timing-aware
  - Timing-voltage-aware

VIDEN

Voltage-based attacker identification

Overview of Viden

Viden Fingerprints ECUs via voltage measurements and achieves attacker identification in four phases



Phases of Viden

- Phase 1: ACK Threshold Learning
  - Executed when Viden is initialized and every time it is updated
  - Measures the dominant CANH & CANL voltages and maps them to the received message's ID in the ECU's receive buffer
  - Learns the ACK Threshold for that message ID
  - Uses this threshold to determine whether this measured voltage outputs from the actual message transmitter or not

Phases of Viden cont.

- Phase 2: Deriving a Voltage Instance
  - Viden uses the learned ACK Threshold to select and process only *non-ACK voltages* that are outputted solely by the message transmitter
  - Uses them to derive a *voltage instance* – set of 6 tracking points *F1 – F6* that reflect the transmitter ECU's voltage output behavior

Phases of Viden cont.

- Phase 3: Attacker Identification
  - Exploits every newly derived voltage instance to construct/update the voltage profile of the message transmitter ECU
  - Messages from the same ECU have almost equivalent instances

    → same voltage profile

    → *FINGERPRINT*
  - Attack scenario:
    - IDS identifies an attack
    - Viden constructs a voltage profile for the attack messages
    - Maps the new profile to the existing voltage profiles (fingerprints) and identifies the attacker ECU

Phases of Viden cont.

- Phase 4: Attacker Verification
  - Verification of attacker is necessary!
    - *Voltage Profile Collision*: Different ECUs, near-equivalent voltage profile
    - *Targeted impersonation*: Attacker ECU mimic some other ECU's voltage output behavior
  - Machine classifiers are run with *momentary voltage instances* as their inputs

Security of Viden

- Naïve adversary
  - Imprudent and continuous attack message injections
  - Un-aware of how ECUs are fingerprinted

    → Cannot evade Viden
- Timing-aware adversary
  - Tries to evade fingerprinting device via timing analysis
  - Viden identifies attacker ECUs using voltage measurements irrespective of message timings

    → Cannot evade Viden

Security of Viden cont.

- Timing-voltage-aware adversary
  - Aware of voltage-based fingerprinting mechanism
  - Tries to evade Viden's fingerprinting device
    - Change the supply voltage
    - Manipulate the output voltage levels
  - Viden continuously updates the voltage profiles in real time

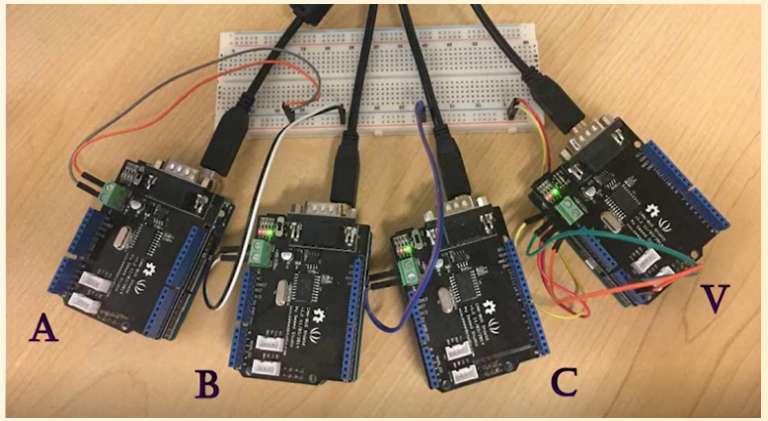
    → Minimize/nullify model-exam discrepancy

    → Difficult to evade Viden

Evaluation

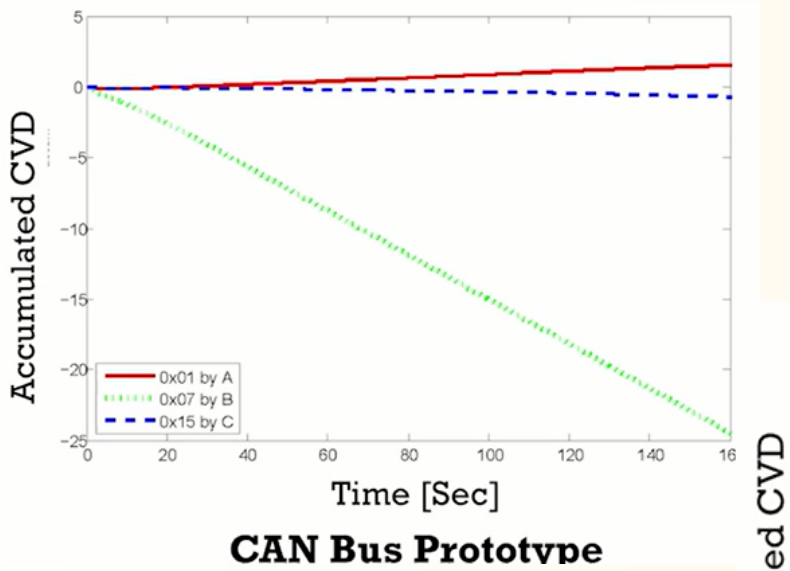
Evaluation Setup

- CAN Bus prototype is configured with four interconnected ECU nodes
- Node A, B, C inject messages 0x01, 0x07, and 0x15 at random message intervals within 20ms – 200ms
- Node V runs Viden and constructs voltage profiles for messages 0x01, 0x07, and 0x15 from nodes A – C
- Two real life cars
  - 2013 Honda Accord
  - 2015 Chevrolet Trax
- A laptop and the Viden node is used to read messages from the CAN Buses of both cars

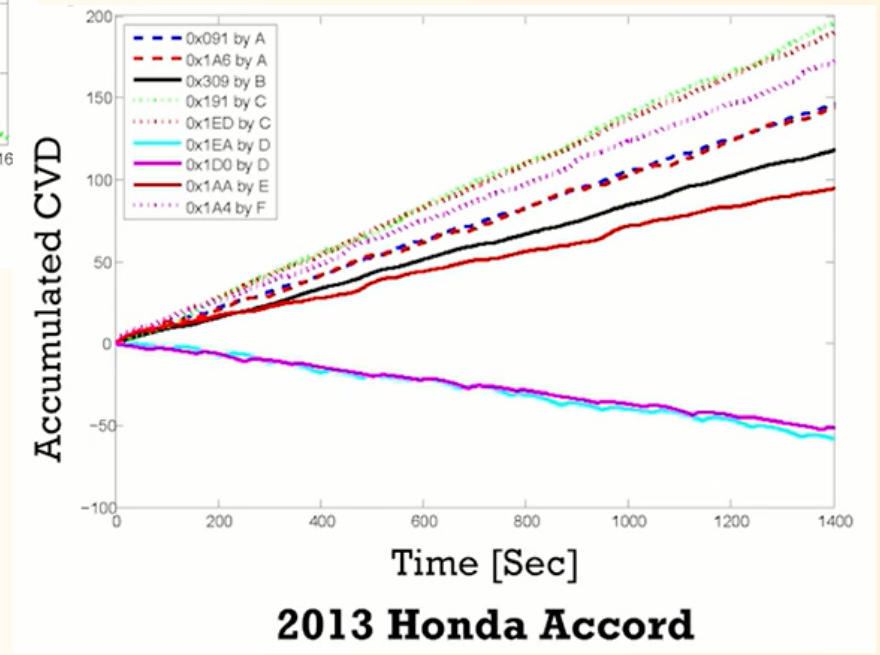


CAN Bus Prototype

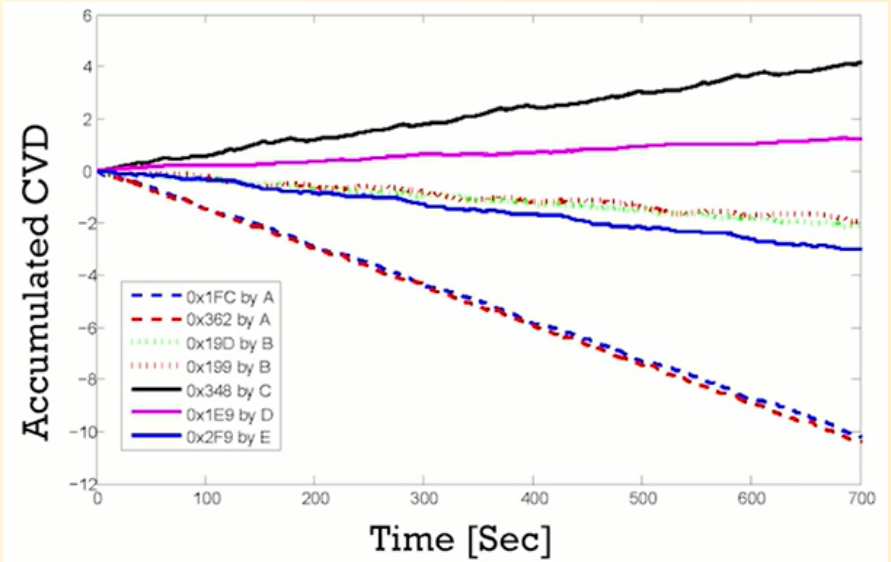
Different Voltage Profiles as Fingerprints



CAN Bus Prototype



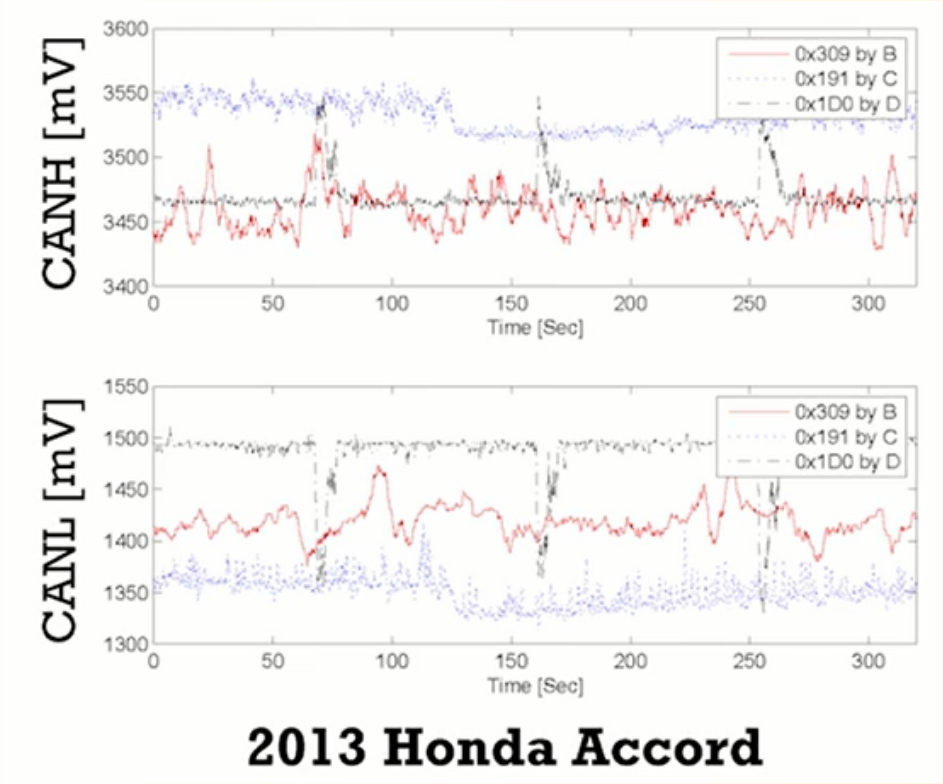
2013 Honda Accord

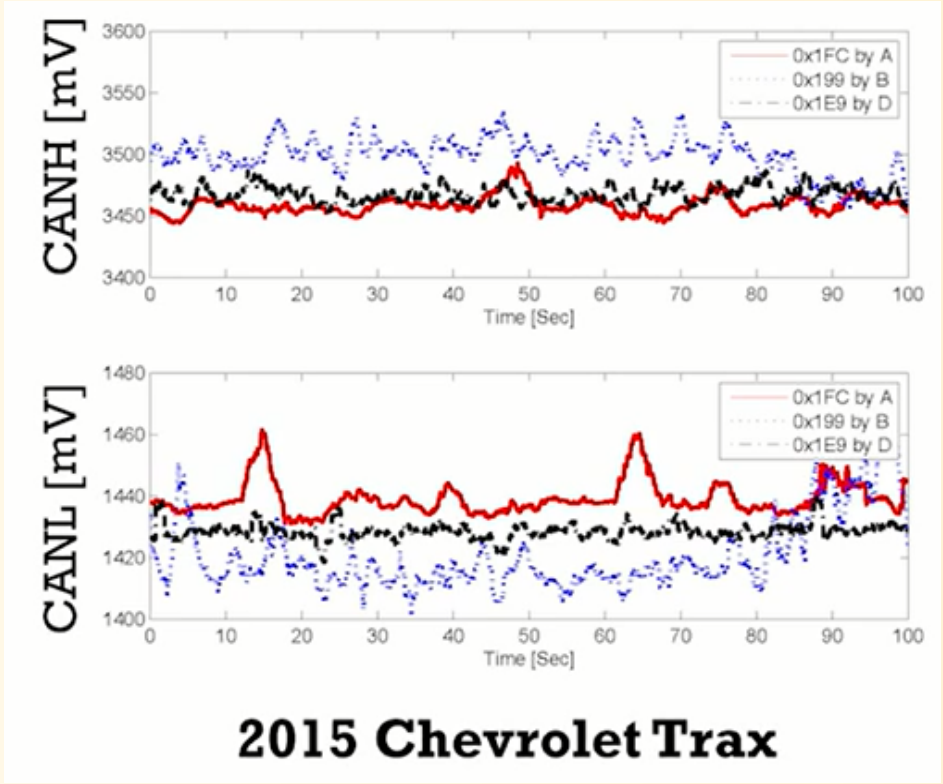


2015 Chevrolet Trax

Voltage Outputs in Real Vehicles

Most frequently measured “non-ACK voltages”





Voltage output levels by different nodes are clearly discriminable

Simulation based evaluation

2000 different attack timings and behavior were considered in both the real vehicles

	#ECUs	False Identification Rate
2013 Honda Accord	6	0.2%
2015 Chevrolet Trax	11	0.3%

Conclusion

Conclusion

- Viden: Voltage based Attacker Identification mechanism on the In-Vehicle network CAN Bus
- Fingerprints transmitter ECUs based on voltage measurements
- Exploits the fingerprints to identify the attacker ECU once an intrusion is detected
- No change in protocol/messages required → low-cost and economic
- Pinpoints the attacker ECU for
  - ✓ Isolation
  - ✓ Forensic
  - ✓ Security patch

THANK YOU