Linda Bushnell
LB2@uw.edu

Vehicle Network Security and Resiliency: Analysis of the Controller Area Network Protocol

Joint work with Sang Sagong and Radha Poovendran

December 9, 2019
IEEE CDC Workshop
Nice, France

ELECTRICAL & COMPUTER ENGINEERING

Outline

Introduction

Controller Area Network (CAN)

Timing-based Attack & Detection

Voltage-based Attack & Detection

Conclusions

Introduction

Vehicle Network Security & Resiliency

Goal:

- Secure data networks in vehicles.

Contribution:

- Developed **timing-based** and **voltage-based** attacks on the physical vehicle network.
- Developed cyber methods to detect intrusions.

CAN Network:

- First deployed on Mercedes-Benz W140 in 1991.
- Network of Electronic Control Units (ECU).

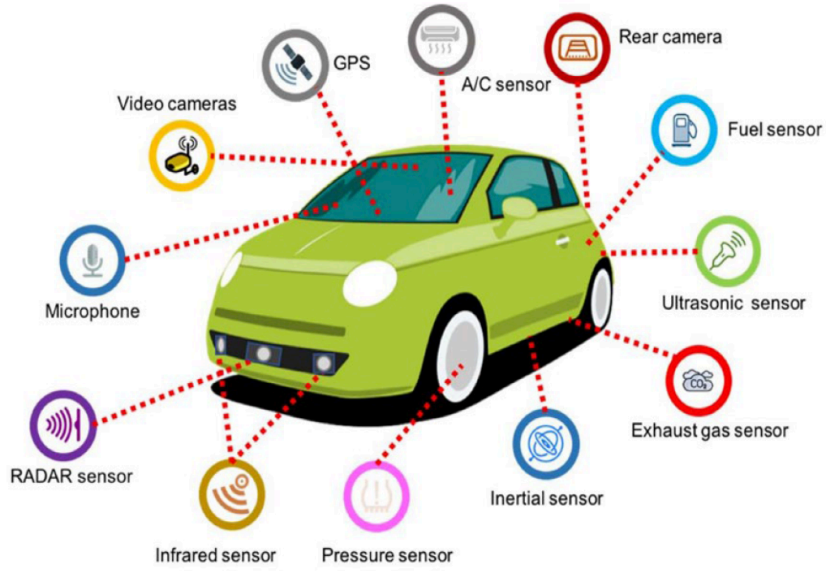


Figure from J. Guerrero-Ibáñez et al., Sensors, 2018

Controller Area Network in Vehicles

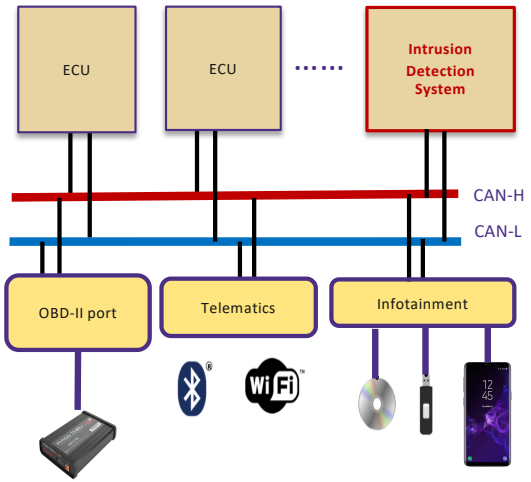
Problem Statement

The Problem:

- In-vehicle data network protocols were designed as closed networks.
- Data is not encrypted or authenticated.
- Newer cars have ECUs with outward-facing interfaces such as CD players, Wi-Fi, and Bluetooth.
- Adversaries can attack outward facing ECUs to block messages, transmit spoofed messages, create false alarms, etc.

The Solution:

- Develop intrusion detection software to detect attacks by tracking abnormal deviations in physical properties of the network, such as frequency and timing of messages or voltage levels.



Data network with outward-facing ECUs

Controller Area Network in Vehicles

Message Exchange & Data Format

- CAN protocol is a **multi-master** (multiple ECUs) and **broadcasting bus** (data available to every ECU) network.
- CAN message is composed of 7 fields.

SOF	*Arbitration*		*Control*			*Data*	*CRC*		*ACK*		*EOF*
SOF	Message ID	RTR	IDE	RB0	DLC	Data	CRC	CRCDEL	ACK	ACKDEL	EOF
0	11 bits	0	0	0	4 bits	8-64 bits	15 bits	1	1 bit	1	1111111

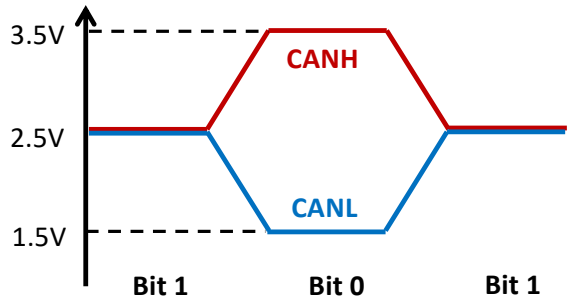
Data frame of CAN protocol

Controller Area Network in Vehicles

Bit Representation

A bit is represented by differential voltage, V_{Diff}, between CANH and CANL.

- Making the CAN bus more robust to external electromagnetic interferences.
- Bit 0 (dominant) if V_{Diff}=2.0V
- Bit 1 (recessive) if V_{Diff}=0.0V



CAN bus acts as a logical AND gate.

- If bits 0 and 1 are simultaneously transmitted, bit 0 is transmitted.

Controller Area Network in Vehicles

Intrusion Detection Systems

Difficult to incorporate authentication into the CAN protocol due to

- Lack of backward compatibility with the legacy systems.
- Resource constraints of ECUs, such as memory.

An Intrusion Detection System (IDS) **detects** abnormal behaviors on the CAN bus such as *message periodicity*, *clock skew*, and *voltage*, which indicates an attack.

- The system operator must mitigate the attack.

An Intrusion Response System (IRS) **detects** and **mitigates** the attack.

- The system operator does not have to act.

Timing-based Attack & Detection

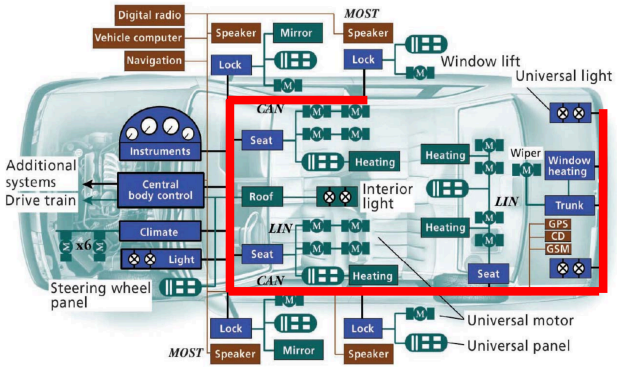
Motivation

How to fingerprint messages:

- Receiver ECU doesn't know when the periodic message was sent by the transmitter ECU; time stamp is added by the receiver.
- Each ECU has a hardware crystal, which is used to compute its internal clock; clocks are not synchronized.
- **Clock offset** is the difference between the clocks of the receiver ECU and transmitter ECU.
- **Clock skew** is the derivative of the clock offset and is a measure of how fast the transmitter ECU is wrt a receiver ECU → ECU **fingerprint.**

How to detect timing-based attacks:

- Intrusion detection software detects attacks based on clock skew deviations as seen from the receiver ECU.



ELECTRICAL & COMPUTER ENGINEERING

Timing-based Attack & Detection

Contribution

- Develop a *Network Time Protocol* (*NTP)-based IDS* that detects an intrusion by exploiting clock skew as defined in the NTP specification.
- Develop a *cloaking attack* that bypasses the IDS by manipulating the inter-transmission time of the spoofed messages.
- Verify performance of the timing-based attack and detection method on a *CAN bus testbed* and the *UW EcoCAR.*

System Model

An ECU transmits periodic messages according to its local clock.

Clock skew of ECU A, S_A, is deviation of ECU A's clock from the reference clock (Receiver ECU) in unit time:[1]

$S_A \overset{\text{def}}{=} \frac{dC_A(t)}{dt} - 1$, where $C_A(t)$ is ECU A's clock when the reference clock is t.

e.g., if $S_A > 0$, then ECU A's clock runs faster than the reference clock.

Clock skew is unique for each ECU since it depends on the quartz hardware inside each ECU.

[1] D. Mills, RFC 1305, 1992.

Adversary Model

We consider two types of attackers, depending on the attacker's capability: *weak* and *strong.*

Weak attacker

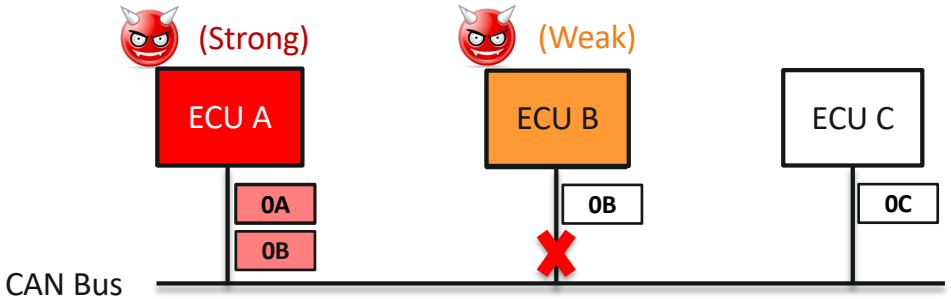
- **Suspends** message transmission from a compromised ECU.
- Performs suspension attack.

Strong attacker

- **Suspends** message transmission and **injects** spoofed messages from a compromised ECU.
- Performs suspension and replay attacks.

Masquerade Attack Model

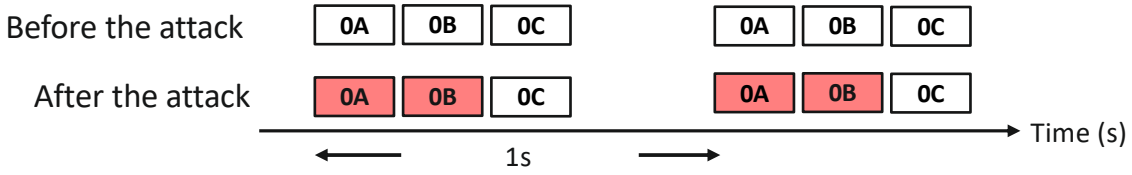
Let ECUs A, B, and C transmit 1Hz messages 0x0A, 0x0B, and 0x0C, respectively.



Weak attacker: suspends message transmission.

Strong attacker: suspends and injects spoofed message.

Message sequence according to the time.



ECU A pretends to be ECU B.

Does not change the traffic through the CAN bus.[2]

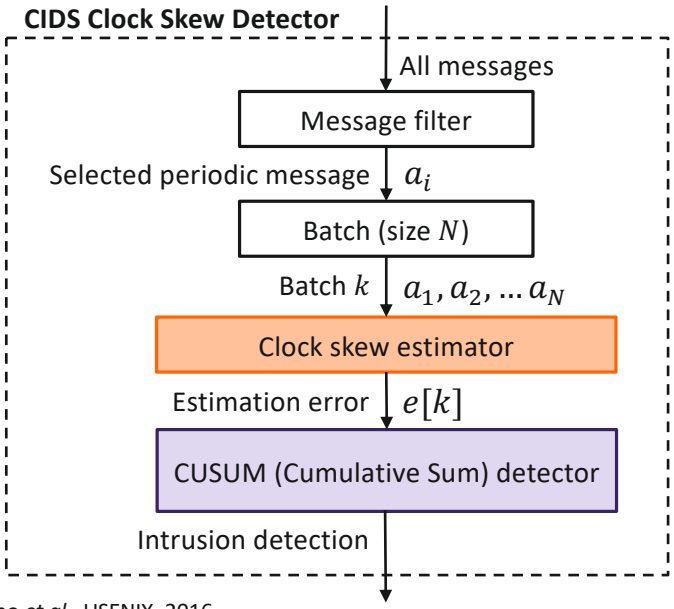
Key take-away: can be identified by intrusion detection software.

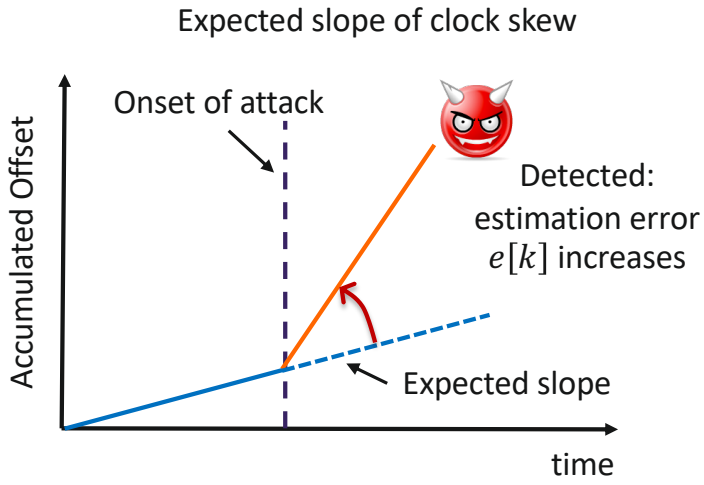
[2] K. Cho *et al.*, USENIX, 2016.

Clock-based IDS

Clock-based IDS detects an intrusion by tracking abnormal deviations in the clock skew of ECUs.[2]

- Computes the clock skew of periodic messages.
- Tracks the clock skew deviation over time using Recursive Least Square (RLS) algorithm.



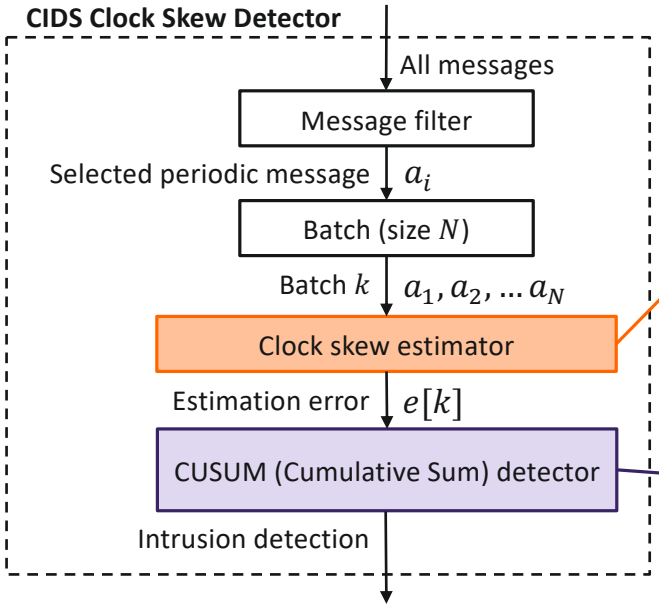


[2] K. Cho *et al.*, USENIX, 2016.

ELECTRICAL & COMPUTER ENGINEERING

Tracking Clock Skew Deviation by Clock-based IDS

Detail procedures of computing and tracking the clock skew.



Average offset

$$O_{avg}[k] = \frac{1}{N-1}\sum_{i=2}^{N}(a_i - (a_1 + (i-1)\boldsymbol{\mu_T[k-1]}))$$

Accumulated offset

$$O_{acc}[k] = O_{acc}[k] + \left|\boldsymbol{O_{avg}[k]}\right| \sim \boldsymbol{S \cdot t + e}$$

Estimation error by RLS

$$e[k] = O_{acc}[k] - S[k-1] \cdot t[k]$$

Normalized error: $\theta_e = \frac{e[k] - \mu_e}{\sigma_e}$

Upper control limit: $L^+ = \max(0, L^+ + \theta_e - \kappa)$

Lower control limit: $L^- = \max(0, L^- - \theta_e - \kappa)$

Intrusion detected: $\max(L^+, L^-) \geq \Gamma$

ELECTRICAL & COMPUTER ENGINEERING

NTP-based IDS

Clock-based IDS computes the clock skew using

- Average offset using mean of the interarrival time in the *previous* batch.
- Accumulated offset by adding an *absolute value* of the average offset.

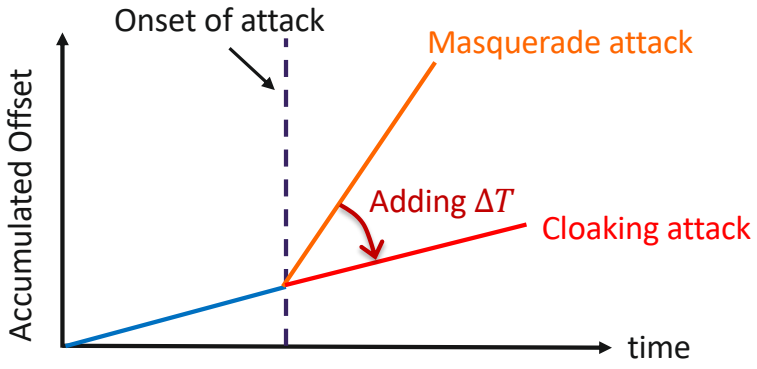
We develop an *NTP-based IDS* to improve the clock-based IDS by changing:

- Average offset $O_{avg}[k] = \frac{T-(a_N-a_0)}{N}$ — Use current data batch
- Accumulated offset $O_{acc}[k] = O_{acc}[k-1] + O_{avg}[k]$ — Don't use absolute value

Next: Analyze both IDS on timing-based attacks.

Cloaking Attack on Clock Skew Detector

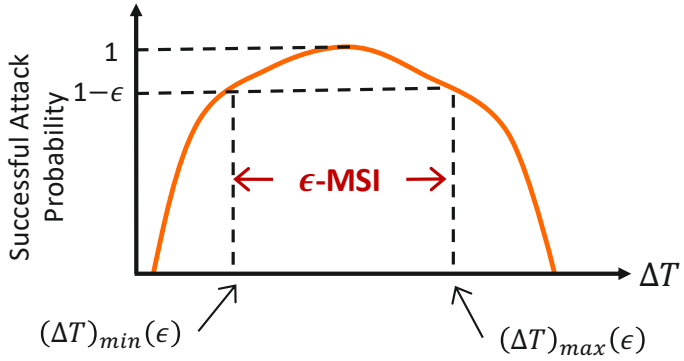
- CIDS and NTP-based IDS compute clock skew using only the interarrival times.
- Idea: develop a *cloaking attack* that is **not detected** by CIDS and NTP-based IDS.
- Cloaking attack = masquerade attack with strong attacker ECU A adding an offset ΔT to the timing of the spoofed message to **exactly match** the timing of ECU B's message.
- Key take-away: cannot be identified by intrusion detection software.



ELECTRICAL & COMPUTER ENGINEERING

Performance Metric

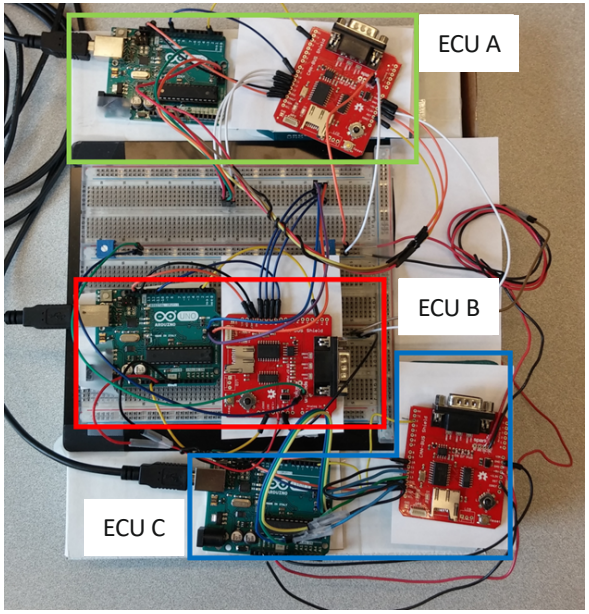
- Measure of how effectively CIDS and NTP-based IDS can detect an intrusion.
- Cloaking attack is successful for a range of ΔT: Adversary does not have to be exact.
- Performance Metric: ϵ-Maximum Slackness Index (MSI): $\epsilon\text{-MSI} = (\Delta T)_{max}(\epsilon) - (\Delta T)_{min}(\epsilon)$



ELECTRICAL & COMPUTER ENGINEERING

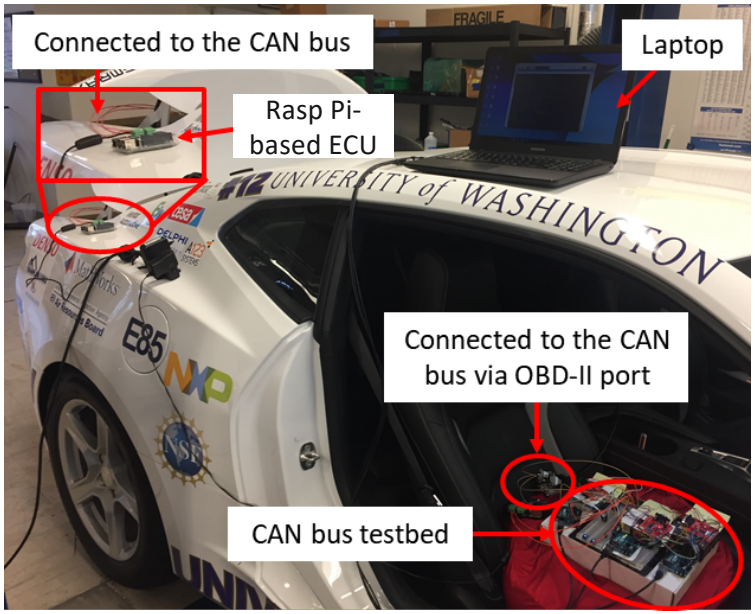
Experiments

Testbeds



CAN bus testbed

- Arduino UNO + CAN bus shield
- 2 different messages, 20msg/s



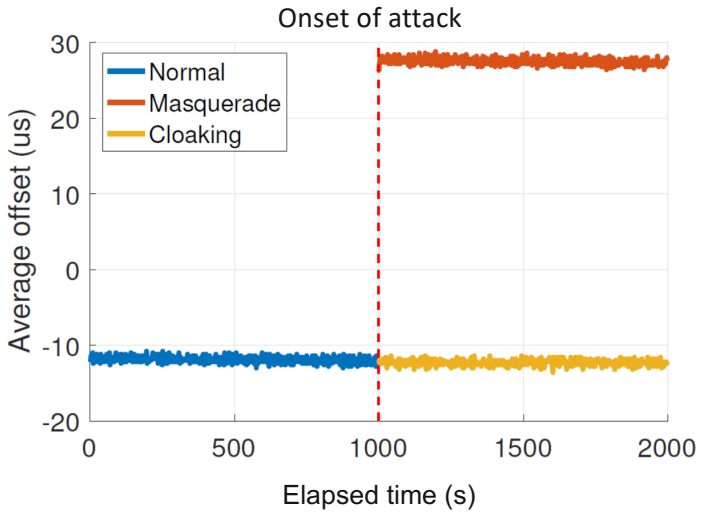
UW EcoCAR

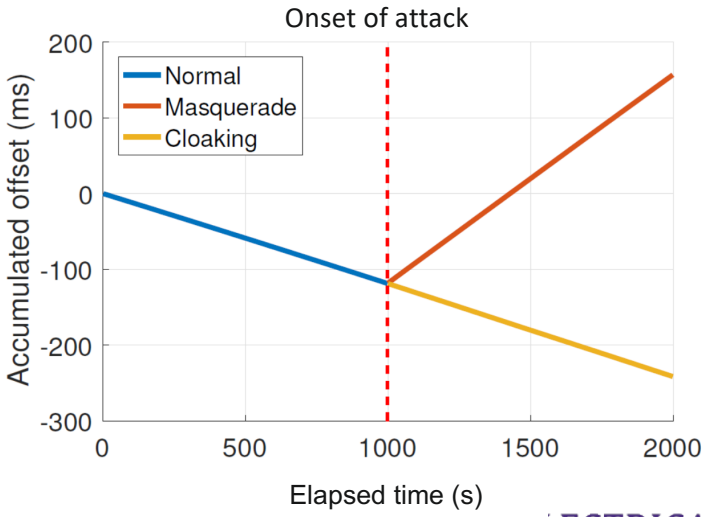
- Rasp Pi-based ECU for data collection
- 89 different messages, ≈2500msg/s

Experiments

Masquerade vs. Cloaking Attacks

- CAN testbed: average and accumulated offsets are computed by the NTP-based IDS.
- Slope of the curve in the accumulated offset plot does not change after the cloaking attack.
- Cloaking attack bypasses both CIDS and NTP-based IDS.



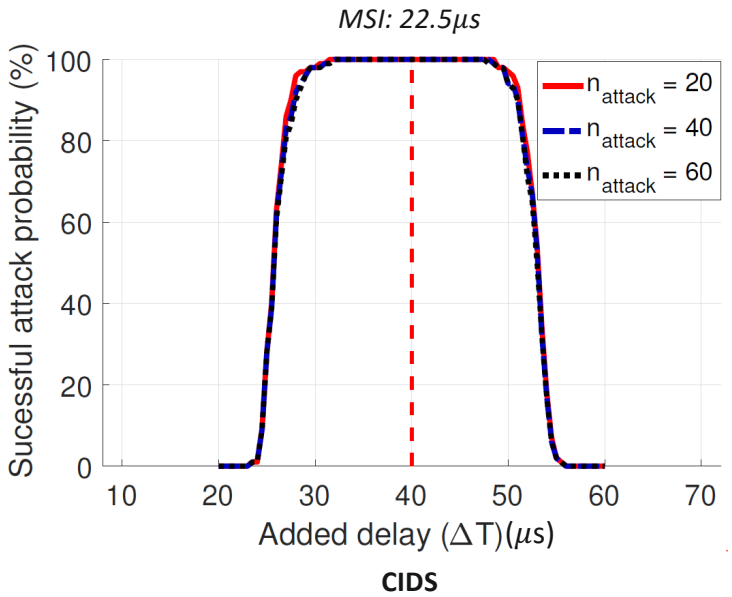


ELECTRICAL & COMPUTER ENGINEERING

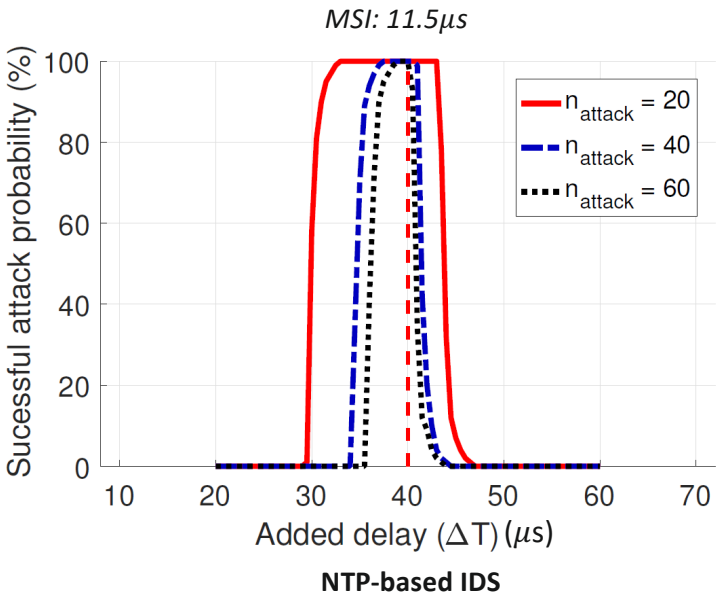
Experiments

Performance Metric

- CAN testbed: NTP-based IDS can detect smaller clock skew deviation.
  - 11μs smaller MSI when ϵ=0.05 and attack duration n_{attack}=20



CIDS



NTP-based IDS

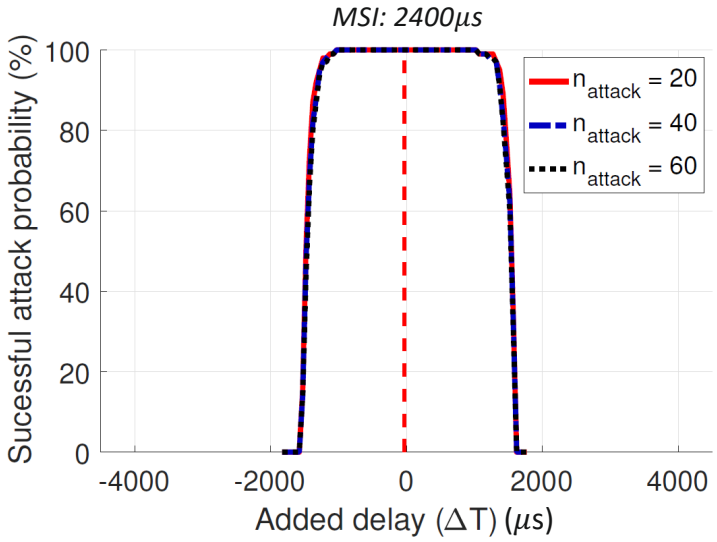
ELECTRICAL & COMPUTER ENGINEERING

Experiments

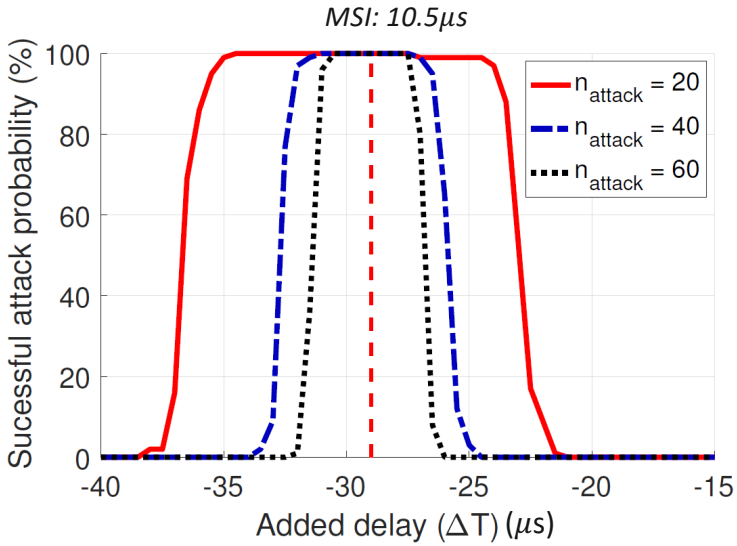
Performance Metric

- EcoCAR testbed: NTP-based IDS can detect smaller clock skew deviation.

  ≈2390μs smaller MSI when ϵ=0.01 and attack duration n_{attack}=20



CIDS



NTP-based IDS

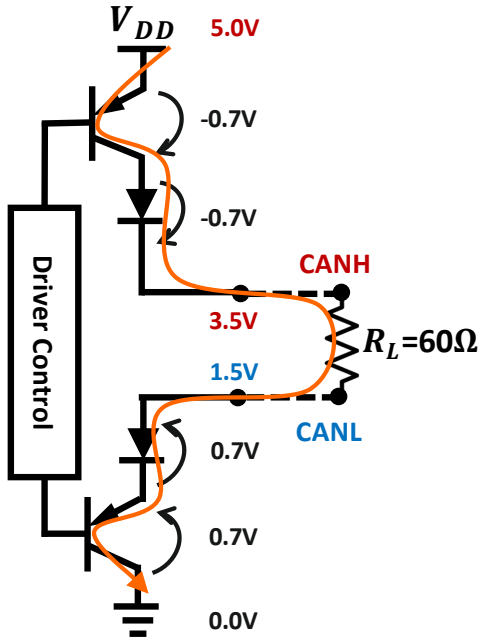
Summary of Timing-based Attack & Detection

- Developed an **NTP-based IDS** that computes the clock skew as defined in the NTP specification.
- Developed the **cloaking attack** that can bypass both the CIDS and the NTP-based IDS by manipulating the inter-transmission time of the spoofed message.
- Verified the performance of the NTP-based IDS and the cloaking attack on the **CAN bus testbed** and the **UW EcoCAR.**

Voltage-based Attack & Detection

Motivation

- Voltage-based IDS measure CAN-H and CAN-L voltages of each ECU using an extra set of wires.
- This introduces new attack surfaces to the CAN bus.
- CAN transceiver (MCP2551) turns on/off two bipolar junction transistors (BJTs) for transmitting bits 0 and 1.
- Transmitting bit 1:
  - CANH and CANL: 2.5V
  - BJTs off; no current through R_L
- Transmitting bit 0:
  - CANH: 3.5V, CANL: 1.5V
  - BJTs on; voltage drop between CANH and CANL due to current flow through R_L



V_{DD}: supply voltage
R_L: termination resistors

Current flow and voltage drop when transmitting bit 0

ELECTRICAL & COMPUTER ENGINEERING

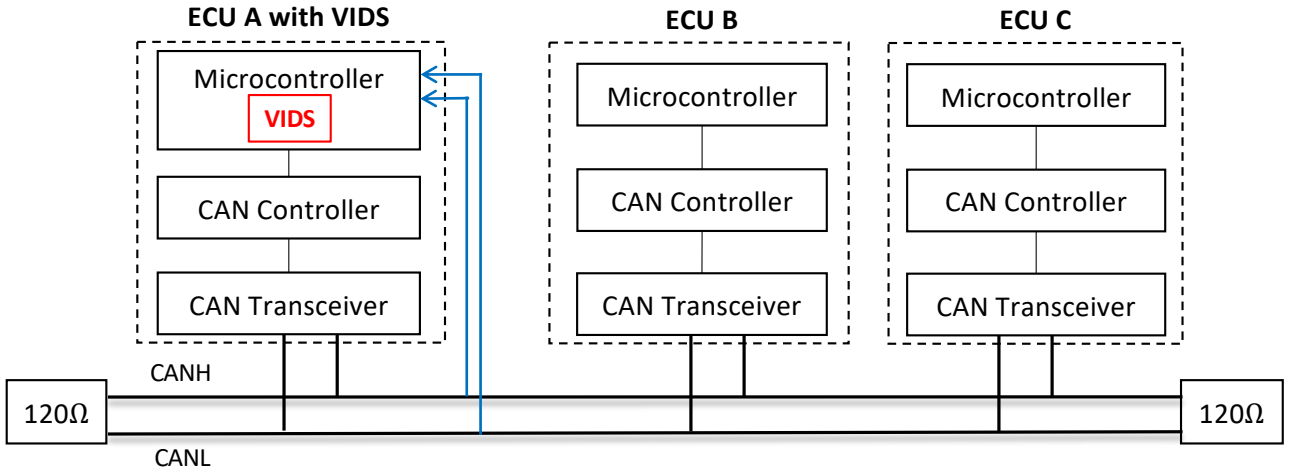
Voltage-based Attack & Detection

Contribution

- Develop three new *voltage-based attacks* using the extra wires to damage the compromised ECU, stopping message transmission.
- Develop a hardware-based *IRS* that detects the new voltage-based attacks and performs mitigation by isolating the compromised ECU.
- Verify the new voltage-based attacks and IRS on the *CAN bus testbed.*

Voltage-based IDS

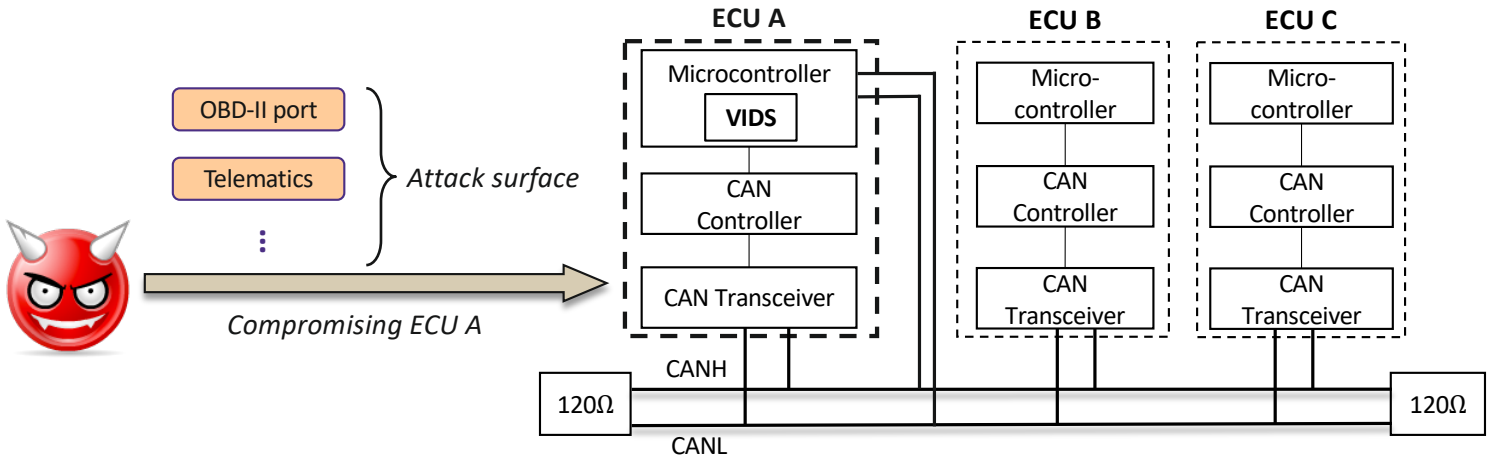
- VIDS detects an intrusion by tracking abnormal deviations in the voltage characteristics of ECUs.[3]-[6]
  - VIDS measures the voltage of the CAN bus using extra wires.
  - Voltage characteristics are difficult to mimic.



[3] M. Kneib *et al.*, CCS, 2018. [4] W. Choi *et al.*, IEEE TIFS, 2018. [5] K. Cho *et al.*, CCS, 2017. [6] P. Murvay *et al.*, IEEE SP Letter, 2014.

System and Adversary Models

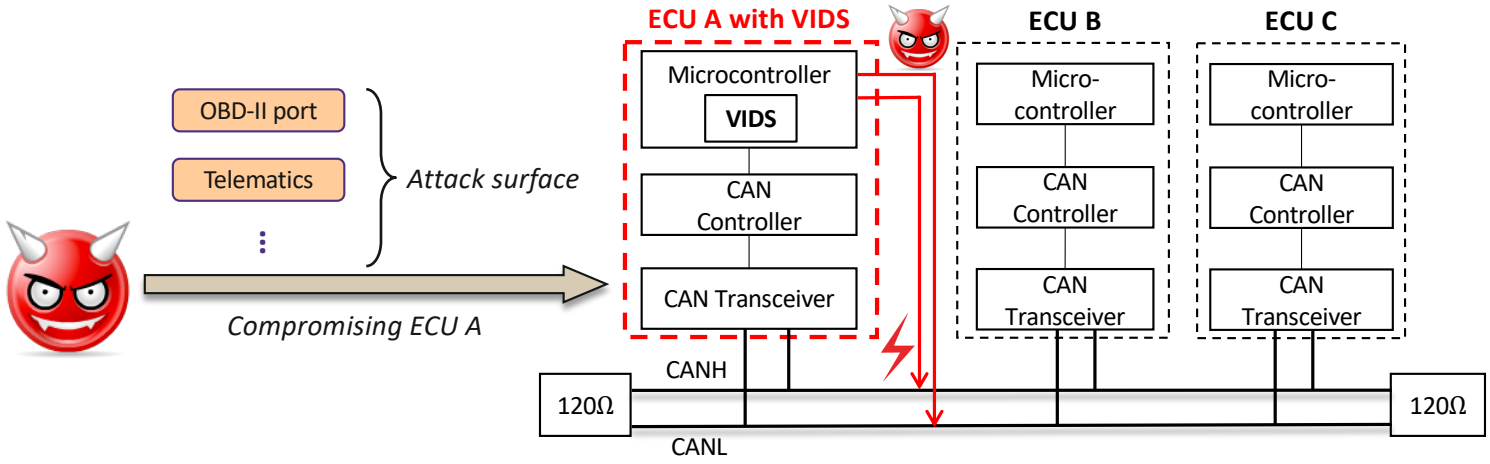
- ECU A with VIDS is compromised using an attack surface such as OBD-II port.[7]
- Adversary can now manipulate voltage levels of analog pins on ECU A.
  - Impedes the CAN transceiver's operation and disables VIDS.



[7] S. Checkoway *et al.*, USENIX, 2011.

System and Adversary Models

- ECU A with VIDS is compromised using an attack surface such as OBD-II port.[7]
- Adversary can now manipulate voltage levels of analog pins on ECU A.
  - Impedes the CAN transceiver's operation and disables VIDS.



[7] S. Checkoway *et al.*, USENIX, 2011.

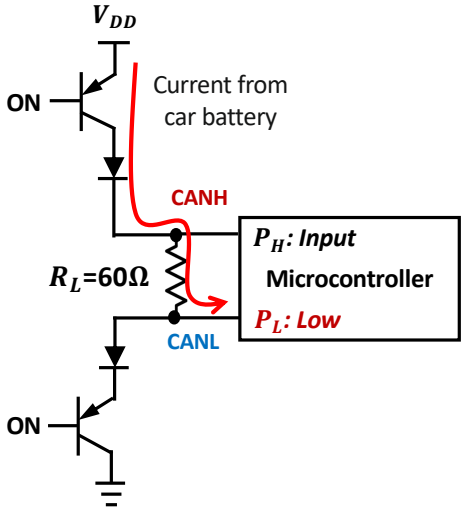
Voltage-based Attacks

- Normal operation: VIDS uses wires to measure the voltage of the ECUs.
- Attack: compromised microcontroller sets the wires to *low and high* voltage to mount an attack.
  - Overcurrent attack
  - Denial-of-Service (DoS) attack
  - Forced retransmission attack

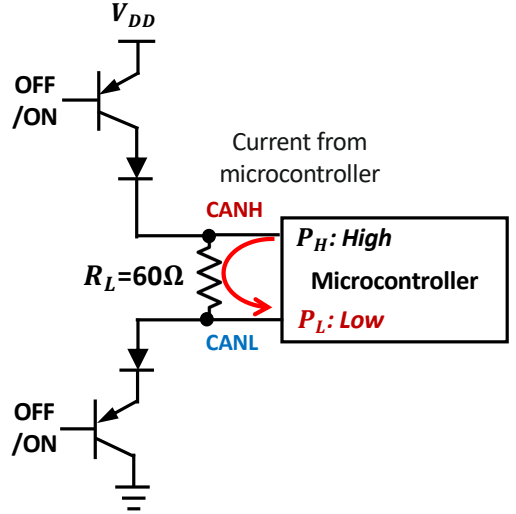
Pin connected to CANH (P_H)	Pin connected to CANL (P_L)	Is it an attack?
Input/measurement	Input/measurement	Normal measurement of voltage
Input/measurement	**High**	**DoS attack**
Input/measurement	**Low**	**Passive overcurrent attack**
High	**Input/measurement**	**Forced retransmission attack**
High	High	Not valid setting
High	**Low**	**Active overcurrent attack**
Low	Input/measurement	DoS or passive overcurrent attack
Low	High	DoS or active overcurrent attack
Low	Low	DoS or passive overcurrent attack

Overcurrent Attack

- Attack: make the current flowing into the microcontroller, P_L, exceed the hardware limit, I_{max} $(20 - 40$ mA range$)$.
- Passive overcurrent attack: P_H in input, P_L in low voltage output.
- Active overcurrent attack: P_H in high voltage output, P_L in low voltage output.



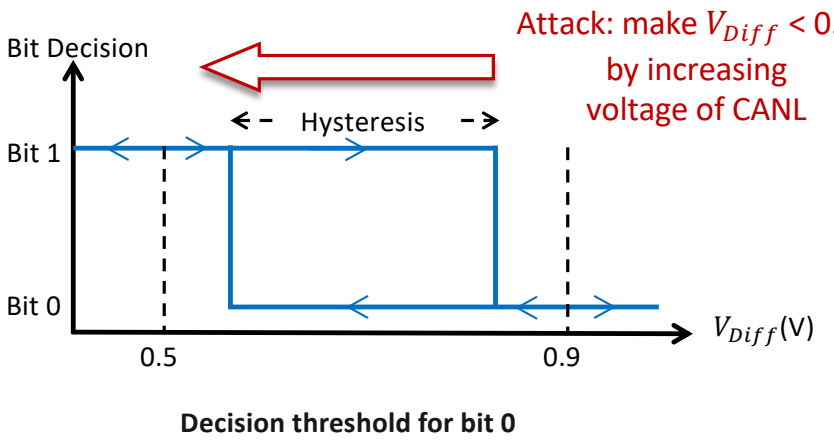
Current flow at passive overcurrent attack



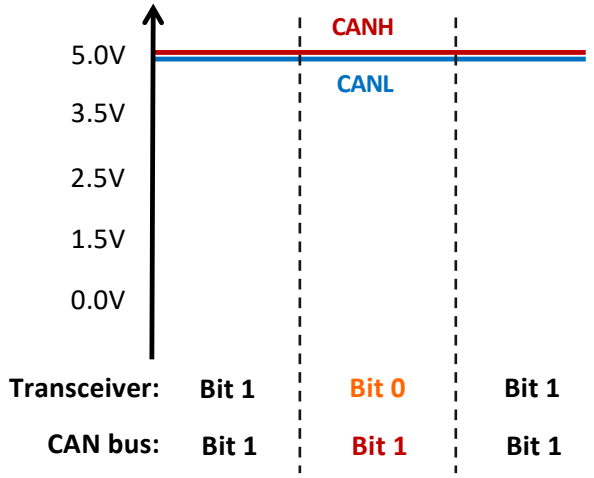
Current flow at active overcurrent attack

DoS Attack

- Attack: keep CAN bus in an idle state by holding the voltages of the CAN bus lines at one level.
  - Increase CANL to make the differential voltage less than the decision threshold for bit 0.
  - Bit 0 cannot be transmitted.
  - P_H in input, P_L in high voltage output.



Decision threshold for bit 0

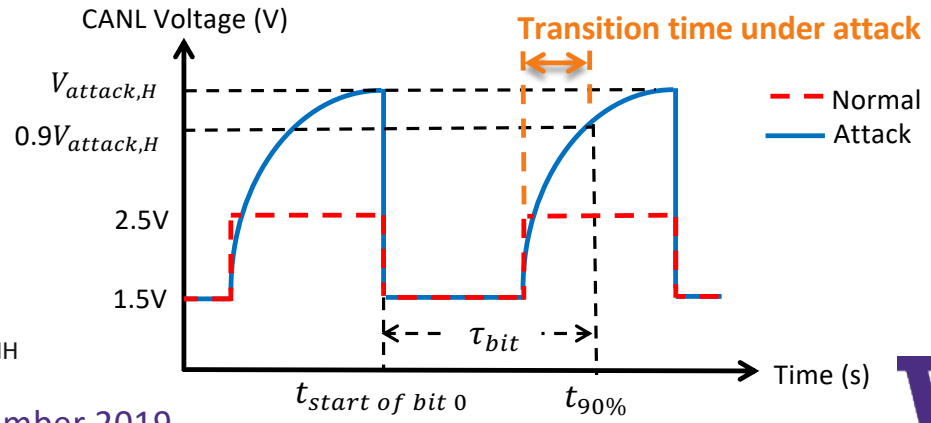


Voltage of the CAN bus when 5V is applied to CANL

ELECTRICAL & COMPUTER ENGINEERING

Forced Retransmission Attack

- Attack: manipulate the bit timing requirement of the CAN protocol to make an error, causing message to be retransmitted.
- Make transition time from bits 0 to 1 longer than the nominal transition time (70 - 130ns).
  - Bit 1 right after bit 0 cannot be transmitted properly.
  - P_H in high output, P_L in input mode.
- Bit length time $\boldsymbol{\tau_{bit} = t_{90\%} - t_{start\ of\ bit\ 0}}$
  - Sum of duration of bit 0 and transition time.



$V_{attack,H}$: voltage applied to CANH

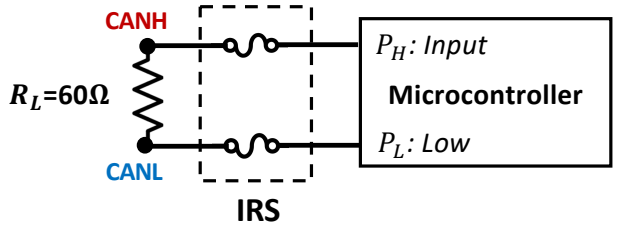
ELECTRICAL & COMPUTER ENGINEERING

Mitigating Voltage-based Attacks

	Active overcurrent attack	Passive overcurrent attack	DoS attack	Forced retransmission attack
Magnitude of current through analog pin	83.3mA flows from P_H to P_L	58.3mA flows to P_L	281mA flows from P_L†	58.3mA flows from P_H

† Experimentally measured value

By selecting a fuse with current rating smaller than the minimum current among the four attacks and I_{max}, VIDS is disconnected from the CAN bus when any one of the voltage-based attacks occur.



Experiments

Testbed

CAN bus testbed

- MCP2515 CAN controller & MCP2551 CAN transceiver.

ECU A

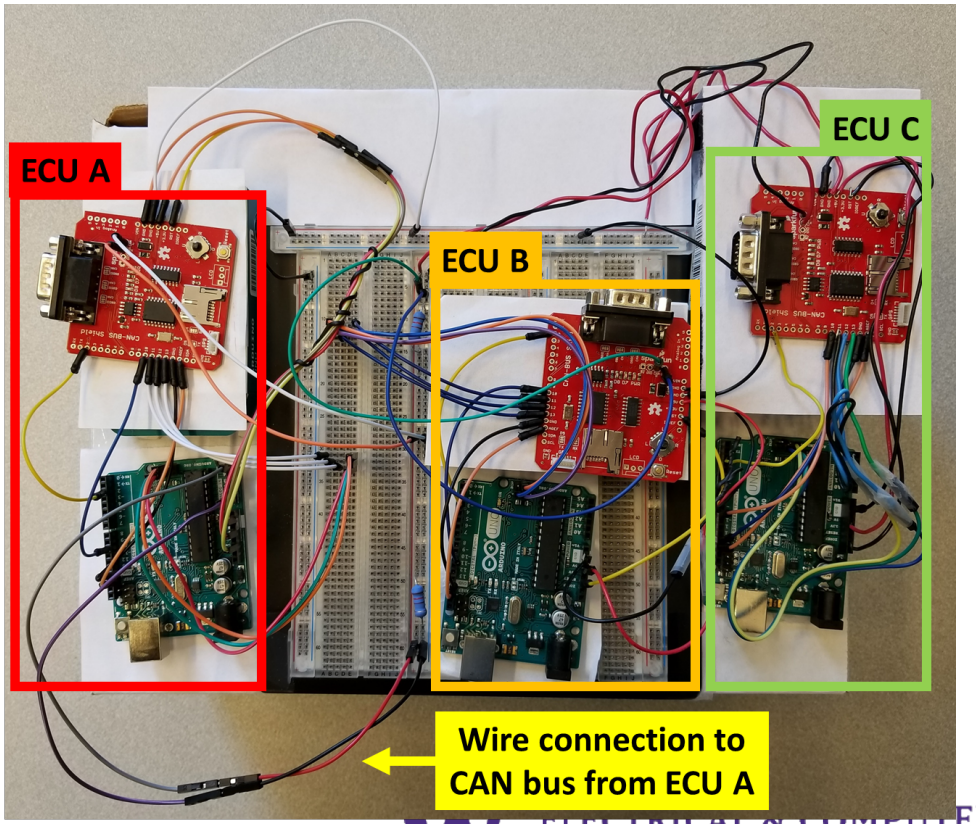
- Two extra wires connected to CAN bus.
- Launching voltage-based attacks.

ECU B

- Receiving and logging messages.

ECU C

- Transmitting 1Hz message with ID 0x01 and data 0x01



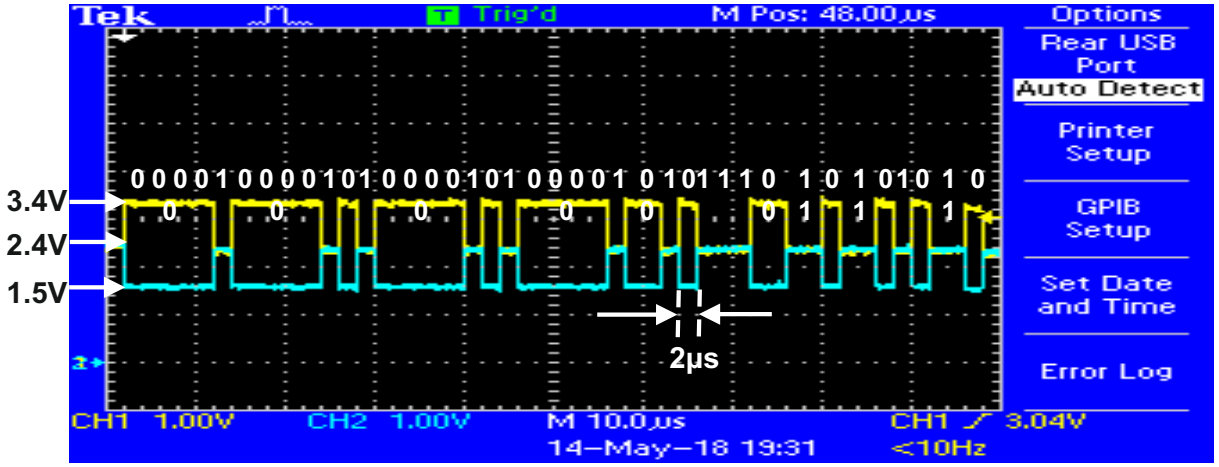
Experiments

Normal Transmission

Voltage of **CANH** and **CANL** in normal transmission measured on the CAN bus testbed.

- Bit 0: 3.4V for CANH and 1.5V for CANL
- Bit 1: 2.4V for CANH and CANL

One bit is 2μs long with transition time in order of 100ns under 500kbps.



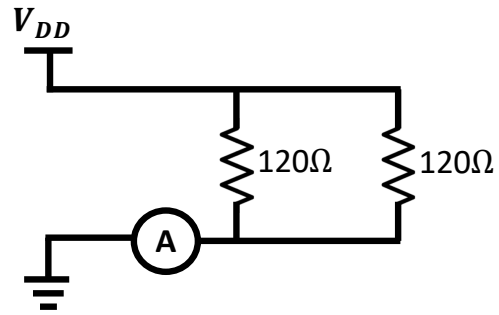
ELECTRICAL & COMPUTER ENGINEERING

Experiments

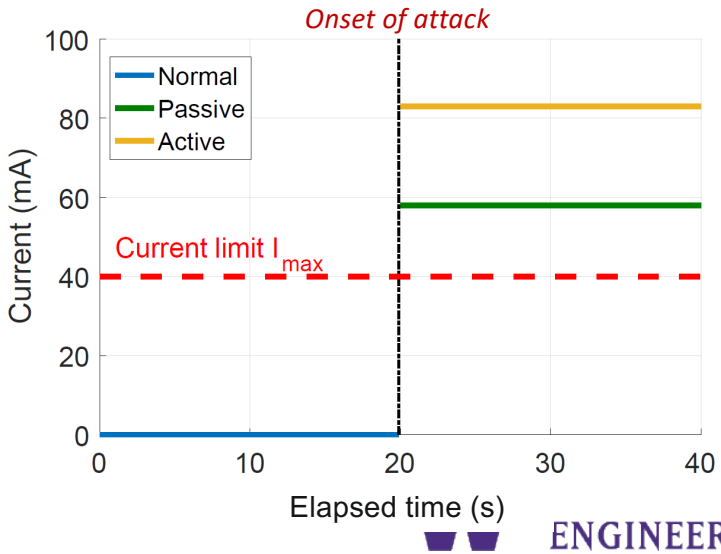
Overcurrent Attack

Current under overcurrent attack is larger than I_{max}, which damages a microcontroller.

- I_{max} = 40mA for Microchip ATmega328P, Renesas V850, and NXP MPC563
- Passive overcurrent attack: 58mA
- Active overcurrent attack: 83mA
- A fuse with current rating smaller than I_{max} can mitigate the overcurrent attack

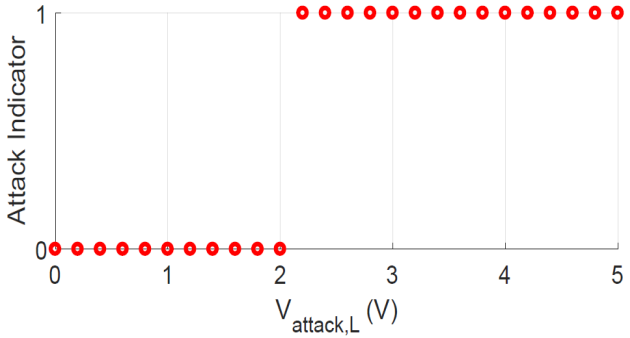


Test circuit emulating overcurrent attack

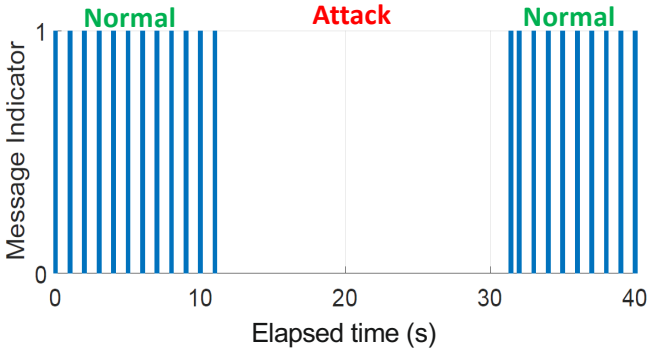


Experiments

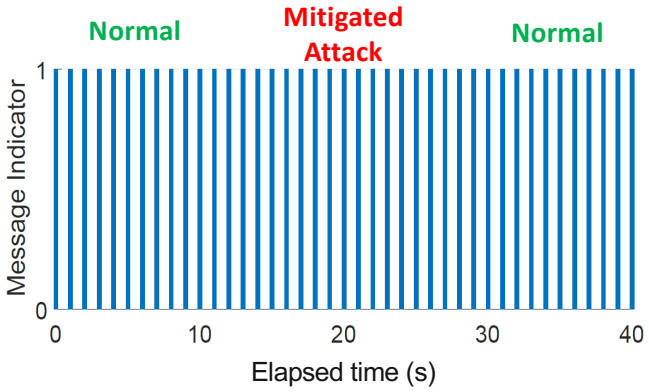
DoS Attack



DoS attack is successful when voltage applied to CANL, $V_{attack,L}$, is larger than 2.2V



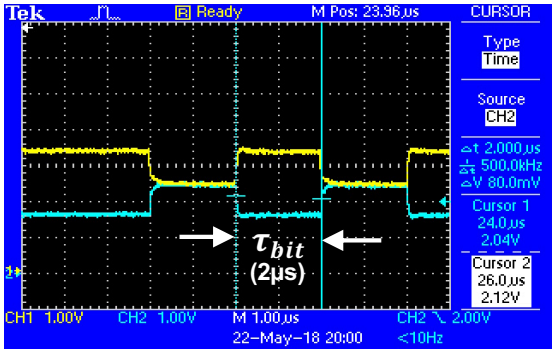
The attack is successfully launched between 11-31s on CAN bus testbed



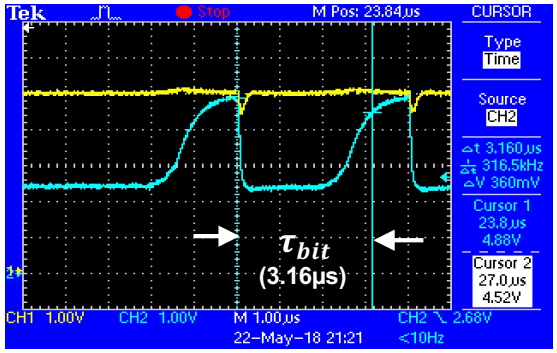
IRS mitigates the attack

Experiments

Forced Retransmission Attack

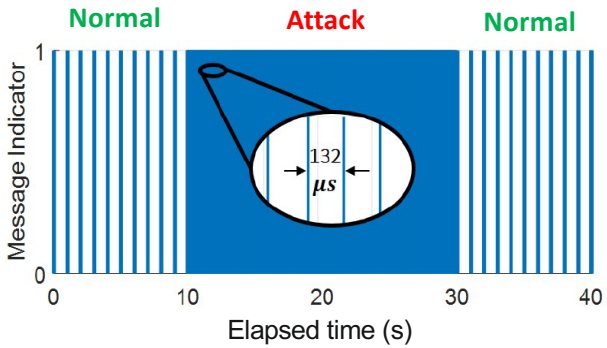


Normal

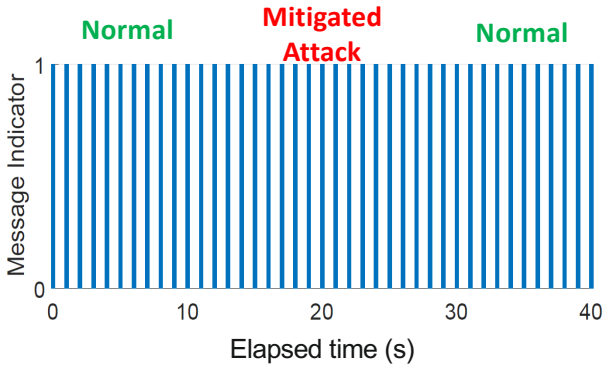


$V_{attack,H}$=5.0V

τ-bit increases



The attack is successfully launched between 10-30s on CAN bus testbed



IRS mitigates the attack

ELECTRICAL & COMPUTER ENGINEERING

Summary of Voltage-based Attack & Detection

- Developed the three new **voltage-based attacks**: overcurrent attack, DoS attack, and forced retransmission attack.
- Developed a **hardware-based IRS** using a fuse that isolates the compromised VIDS.
- Verified the voltage-based attacks and hardware-based IRS on the **CAN bus testbed.**

Thank You

This work was supported by ONR grants N00014-16-1-2710 and N00014-17-1-2946, NSF grants CNS-1446866 and CNS-1544173, and ARO grant W911NF-16-1-0485.

ELECTRICAL & COMPUTER ENGINEERING