Shape of the Cloak: Formal Analysis of Clock Skew-Based Intrusion Detection System in Controller Area Networks

Xuhang Ying, Sang Uk Sagong, *Student Member, IEEE,* Andrew Clark, *Member, IEEE,* Linda Bushnell, *Fellow, IEEE,* and Radha Poovendran, *Fellow, IEEE*

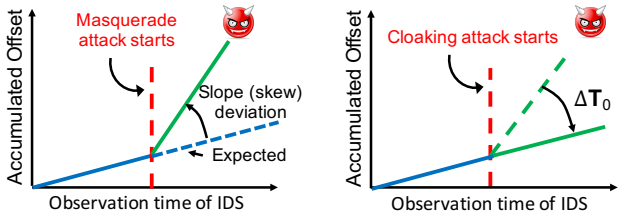
***Abstract*—This paper presents a new masquerade attack called the cloaking attack and provides formal analyses for clock skew-based Intrusion Detection Systems (IDSs) that detect masquerade attacks in the Controller Area Network (CAN) in automobiles. In the cloaking attack, the adversary manipulates the message inter-transmission times of spoofed messages by adding delays so as to emulate a desired clock skew and avoid detection. In order to predict and characterize the impact of the cloaking attack in terms of the attack success probability on a given CAN bus and IDS, we develop formal models for two clock skew-based IDSs, i.e., the state-of-the-art (SOTA) IDS and its adaptation to the widely used Network Time Protocol (NTP), using parameters of the attacker, the detector, and the hardware platform. To the best of our knowledge, this is the first paper that provides formal analyses of clock skew-based IDSs in automotive CAN. We implement the cloaking attack on two hardware testbeds, a prototype and a real vehicle (the University of Washington (UW) EcoCAR), and demonstrate its effectiveness against both the SOTA and NTP-based IDSs. By comparing each predicted attack success probability curve against its experimental curve, we find that the average prediction error is within 3.0% for the SOTA IDS and 5.7% for the NTP-based IDS.**

***Index Terms*—CPS Security, Formal Analysis, Controller Area Network, Intrusion Detection System, Cloaking Attack**

I. Introduction

Recent studies have identified security vulnerabilities in networked automobiles, in which attackers have compromised in-vehicle Electronic Control Units (ECUs), and disabled brakes [2], remotely controlled steering [3], and disabled vehicles on a highway [4]. Such exploits of ECUs are feasible because in-vehicle network protocols, such as the Controller Area Network (CAN) [5], were designed for closed systems and do not have security mechanisms such as message authentication. Networked automobiles, however, contain externally accessible ECUs that can be compromised by remote adversaries [2], [6], [7]. Since the CAN bus is a broadcast medium and there is no message authentication, a compromised ECU can be used to inject spoofed messages with faked message IDs and masquerade as a targeted ECU (masquerade attack) [2].

Given that CAN has a preset tight bit budget for messages and resource-constrained ECUs have real-time requirements, it has not been a practical option to incorporate cryptographic primitives as in [8]–[10] into CAN. As an alternative, Intrusion Detection Systems (IDSs) have been proposed that exploit physical properties such as message periodicity and network entropy without modifying the CAN protocol [11]–[14].



(a) (b)

Fig. 1: Clock skew estimated by the IDS at the receiver. (a) An IDS tracks the clock skew of the transmitter and detects deviations due to masquerade attacks. (b) A cloaking adversary adds a delay ΔT_0 to the message inter-transmission times to emulate the targeted ECU's clock skew and bypass the IDS.

One state-of-the-art (SOTA) IDS was proposed in USENIX 2016 [12] based on two key observations: 1) almost all CAN messages are periodic, and 2) periodically received messages can be used to estimate the *clock skew* of the transmitter, a unique physical invariant of each ECU due to variations in the clock's hardware crystal. Therefore, a change in estimated clock skew at the receiver implies an anomaly in the transmitter's clock characteristics, which indicates the presence of a masquerade attack with high probability (Fig. 1(a)). The novelty of the SOTA IDS is the use of the clock skew for detecting a masquerade attack without requiring any synchronization and identifying the compromised ECU that mounts the attack.

In our preliminary work [1], we investigated IDSs that use the clock skew for detecting masquerade attacks. Our key observation is that an adversary, who realizes that the IDS at the receiver ECU computes the clock skew using message inter-arrival times, can manipulate the inter-transmission times by adding delays to emulate the clock skew of the targeted ECU and avoid detection. We refer to masquerade attacks of this kind as the *cloaking attack* (Fig. 1(b)). We experimentally obtained the attack success probability curves (attack success probability as a function of the added inter-transmission delay) and noticed that they have a consistent bell-shaped structure across different hardware platforms, which may be captured by

X. Ying, S. U. Sagong, L. Bushnell and R. Poovendran are with the Department of Electrical and Computer Engineering, University of Washington, Seattle, WA, 98195-2500.). {xhying,sagong,lb2,rp3}@uw.edu

A. Clark is with the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA, 01609. aclark@wpi.edu

Part of this work was presented at ACM/IEEE ICCPS 2018 [1].

a formal model. In this paper, we provide such formal models that accurately predict and characterize the attack success probability curves for the SOTA IDS and its adaptation to the Network Time Protocol (NTP), using parameters of the attacker, the detector, and the hardware platform. Moreover, we collect additional 16+ hours of CAN data from the UW EcoCAR testbed for six representation messages with different periods, message ID levels, and transmitting ECUs to validate our formal models. To the best of our knowledge, this is the first paper that provides formal analyses of clock skew-based IDSs in automotive CAN. Throughout this paper, we make the following specific contributions:

- We propose the cloaking attack, in which an adversary adjusts message inter-transmission times and cloaks its clock to match the targeted ECU's clock skew and avoid detection.
- We analyze and formally model the attack success probability of the proposed attack on both the SOTA and NTP-based IDSs.
- We evaluate the proposed attack on hardware testbeds, including a CAN bus prototype and a real vehicle (the UW EcoCAR). Our results show that while the NTP-based IDS is more effective than the SOTA IDS in detecting masquerade attacks, the cloaking attack is successful against both IDSs during all hardware trials.
- We validate our formal analyses using the data collected from the UW EcoCAR. We define a metric called the *Area Deviation Error* (ADE) to measure the modeling accuracy, which is the ratio of the absolute difference of the areas under the predicted and experimental attack success probability curves to the area under the experimental curve. Our results show that the average ADEs of the proposed formal models are within 3.0% for the SOTA IDS and 5.7% for the NTP-based IDS.

The remainder of this paper is organized as follows. Section II reviews the related work. Sections III presents our system and adversary models. Section IV reviews the SOTA IDS and presents the proposed NTP-based IDS. The cloaking attack is proposed in Section V. Section VI presents formal models for the SOTA and NTP-based IDSs. Section VII presents the experimental evaluation. Section VIII concludes this paper.

TABLE I: Frequently used notations.

Notation	Description
$a_{k,i}$	Arrival time of i-th message in k-th batch
$\eta_{k,i}$	Noise in arrival time of i-th message in k-th batch
μ	Mean of all inter-arrival times before the attack
$\mu[k]$	Mean of inter-arrival times in k-th batch
σ	Standard deviation of all inter-arrival times
σ_η	Standard deviation of noise in arrival times
N	Batch size
O	(Constant) clock offset in each period T
$O_{avg}[k]$	Average offset in k-th batch
$O_{acc}[k]$	Accumulated offset up to k-th batch
$S[k]$	Clock skew estimate in k-th batch
$t[k]$	Elapsed time up to last message in k-th batch
$e[k]$	(Unnormalized) identification error in k-th batch
μ_{CUSUM}	Mean of reference identification errors
σ_{CUSUM}	Standard deviation of reference identification errors
$e_n[k]$	Normalized identification error in k-th batch
$e_{ref}[k]$	Identification error used as reference in CUSUM
$L^+[k]$, $L^-[k]$	Upper and lower control limits in k-th batch
Γ	CUSUM detection threshold
γ	CUSUM update threshold
κ	CUSUM sensitivity parameter
ΔT_0	Inter-transmission delay added by adversary that exactly achieves the targeted ECU's clock skew
ΔT	Difference between the total added delay and ΔT_0
P_s	Probability of a successful cloaking attack
τ	Rate of decrease of normalized identification error after an attack occurs (for the SOTA IDS)
$\hat{S}[k], \hat{t}[k]$ $\hat{O}_{acc}[k], \hat{e}[k]$	Expected value of $S[k], t[k], O_{acc}[k], e[k]$ (for the NTP-based IDS)

II. Related Work

Recent experimental studies have shown that automobiles are vulnerable to cyber attacks with potentially life-threatening consequences such as disabling brakes or overriding steering [2], [6], [7], [15]–[17], most of which are caused by the lack of security protections in CAN [2], [8]. Hence, there is an urgent need for securing CAN buses.

Security solutions for CAN can be broadly classified into schemes that add cryptographic measures to the CAN bus [8]–[10], [18] and anomaly-based IDSs that 1) analyze the traffic on the CAN bus including message contents [19]–[21], timing/frequency [15], [22]–[25], entropy [26], and survival rates [27], 2) exploit the physical characteristics of ECUs extracted from in-vehicle sensing data [28]–[30] or measurements [11], [13], [14], [31], [32], and 3) exploit the characteristics of the CAN protocol, such as the remote frame [33]. Compared to the CAN traffic, it is more difficult for adversaries to imitate the physical characteristics of ECUs, such as the mean squared error of voltage measurements [11]. In [13], Cho and Shin proposed an IDS called Viden that constructs voltage profiles to identify the attacker. In [32], Choi *et al.* proposed VoltageIDS that leverages the time and frequency domain features of the electrical CAN signals to fingerprint ECUs. In [34], Kneib and Huth proposed Scission that exploits physical characteristics from analog values of CAN frames to determines if whether was transmitted by the legitimate ECU. However, real-time sensing/measurement and processing can be challenging for ECUs with limited resource, which may hinder the deployment of the existing schemes in practice. In addition, it has been shown in [35] that the extra wires required by voltage-based IDSs may introduce new attack surfaces for various voltage-based attacks.

A novel IDS that uses the clock skew to fingerprint ECUs was proposed in [12]. As a physical invariant, the clock skew can be estimated from the timestamps of periodically received CAN messages and used for detecting masquerade attacks. In this paper, we propose the cloaking attack, in which the adversary alters the message inter-transmission times to match the clock skew of the targeted ECU and evade detection with a high probability. We further propose formal models that predict the attack success probability for a given CAN bus and IDS with high accuracy.

III. System Model

In this section, we provide brief background on the CAN protocol, review clock-related concepts as defined in NTP, and

present our timing model for the CAN bus. A list of frequently used notations is provided in Table I.

A. CAN Background

The CAN protocol [36], [37] is one of the most widely used in-vehicle network standards. It allows in-vehicle ECUs to broadcast messages, and almost all CAN messages are periodic. In particular, CAN messages do not have transmit timestamps and do not support encryption or authentication.

B. Clock-Related Concepts in NTP

Let $C_A(t)$ denote the time kept by clock A, and $C_{true}(t) = t$ be the true time. According to the NTP [38], [39], the *clock offset* of clock A is given by

$$O_A(t) = C_A(t) - C_{true}(t), \tag{1}$$

which is the difference between the time reported by C_A and the true time. The *frequency* of C_A at time t, denoted $C'_A(t)$, is the first derivative of $C_A(t)$, while the *clock skew* $S_A(t)$ is the first derivative of the clock offset $O_A(t)$. A positive clock skew means that C_A runs faster than C_{true}. The unit of clock skew is microseconds per second (μs/s) or parts per million (ppm). For example, if C_A is faster by 2 μs every 20 ms w.r.t. C_{true}, then its clock skew relative to C_{true} is 100 ppm.

In-vehicle ECUs typically have constant clock skews [12]. Suppose that C_A has a constant clock skew S_A. If Δt is the time duration measured by C_{true}, the amount of time that has passed according to C_A is $\Delta t_A = (1 + S_A)\Delta t$, and $\Delta t = \Delta t_A/(1 + S_A)$. Similarly, if there is a second non-true clock B with a constant clock skew S_B that reports a time duration of Δt_B, we have $\Delta t_B = (1 + S_B)\Delta t$. Then the clock skew of C_B relative to C_A, denoted as S_{BA}, is given by

$$S_{BA} = \frac{\Delta t_B - \Delta t_A}{\Delta t_A} = \frac{S_B - S_A}{1 + S_A} \tag{2}$$

and the relationship between S_{BA} and S_{AB} is given by

$$S_{AB} = \frac{-S_{BA}}{1 + S_{BA}}. \tag{3}$$

In the absence of a true clock, the *relative clock offset* and *relative clock skew* can be defined with respect to a reference clock. Two clocks are said to be *synchronized* at time t if both the relative clock offset and relative clock skew are zero.

C. Timing Model

We now discuss our timing model in Fig. 2, in which the receiving ECU R timestamps messages that arrive periodically. We consider R's clock as the reference clock and refer to the relative offset and relative skew of the transmitter's clock as offset and skew, respectively.

Consider an ECU that transmits a message every T seconds as per its local clock. If the two clocks are synchronized, the i-th message will be transmitted at $t_i = iT$ in R's clock. However, due to the transmitter's clock skew, there exists an *accumulated offset* O_i between the transmitter's clock that reports time iT and R's clock that reports time t_i since the transmission of message 0, which means $O_i = iT - t_i$

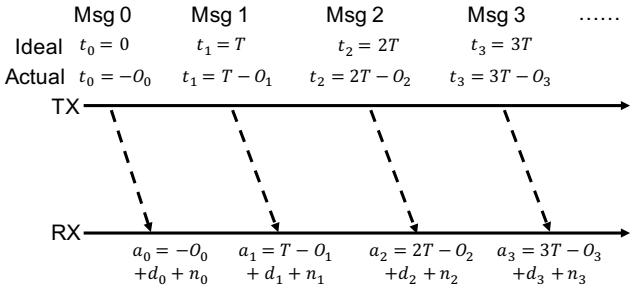


Fig. 2: Timing model of message arrivals on CAN bus.

according to Eq. (1). Therefore, the actual transmission time is $t_i = iT - O_i$ in R's clock. While the clock skew may be slowly varying due to factors like temperature, it is almost constant over short durations. Hence, we model the accumulated offset as a random variable $O_i = iO + \epsilon_i$, where O is the clock offset induced in one period T given the constant clock skew, and ϵ_i is the offset deviation due to jitters in the transmitter. We assume that the ϵ_i's are independent and identically distributed zero-mean random variables. After a network delay of d_i (due to message transmission, propagation, and reception), the message arrives at R's incoming buffer and has a timestamp

$$a_i = iT - iO - \epsilon_i + d_i + n_i, \tag{4}$$

where n_i is the zero-mean noise introduced by R's timestamp quantization process [40].

Let $\eta_i = -\epsilon_i + d_i + n_i$ and thus $a_i = iT - iO + \eta_i$. Since the data lengths of periodic CAN messages are constant over time, it is reasonable to assume constant-mean network delays, i.e., $\mathbb{E}[d_i] = d$. Hence, we model the η_i's as i.i.d. Gaussian random variables with $\eta_i \sim N(d, \sigma_\eta^2)$.

The inter-arrival time between the $(i-1)$-th message and the i-th message is $T_{rx,i} = a_i - a_{i-1} = (T - O) + (\eta_i - \eta_{i-1})$. Hence, the inter-arrival times have a mean $\mu \triangleq \mathbb{E}[T_{rx,i}] = T - O$, and a variance $\sigma^2 \triangleq Var(T_{rx,i}) = 2\sigma_\eta^2$.

D. Adversary Model

We consider adversaries who gain access to the CAN bus of an automobile by compromising one or more ECUs. We adopt the following two adversary models [12], [17]:

- *Weak adversary* – A weak adversary who compromises an ECU is able to eavesdrop on all the CAN traffic and can block outgoing messages from the compromised ECU. The weak adversary, however, cannot send messages from the compromised ECU.
- *Strong adversary* – A strong adversary who compromises an ECU has complete control over the compromised ECU, including eavesdropping on all messages, blocking outgoing messages, and transmitting messages with the timing and content of the adversary's choosing.

We consider adversaries who attempt to mount *masquerade attacks*. Fig. 3 illustrates a masquerade attack that is mounted by a weak adversary and a strong adversary acting in coordination. The strong adversary has compromised ECU A, while the weak adversary has compromised ECU B. The goal of the attack is to inject false messages from ECU A, so as to

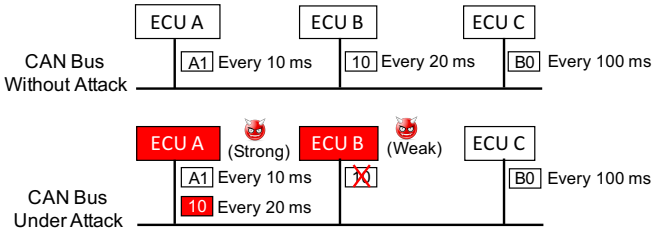


Fig. 3: Illustration of a masquerade attack. Without a masquerade attack, ECU A transmits message 0xA1 every 10 ms, and ECU B transmits message 0x10 every 20 ms. During the masquerade attack, ECU B is weakly compromised and its transmission of 0x10 is blocked. Meanwhile, ECU A is strongly compromised and is used to inject the false messages 0x10 every 20 ms in addition to its original message 0xA1.

degrade the safety, performance, and/or functionality of the vehicle. This attack enables an adversary who compromises a low-priority[1] ECU to effectively impersonate a higher-priority ECU, thus maximizing the impact of the attack.

We observe that, if ECU B were compromised by a strong adversary, the attack would be trivial. On the other hand, when ECU B is compromised by a weak adversary, the adversary cannot directly inject messages from ECU B itself. Instead, the weak adversary blocks the targeted messages from ECU B. The strong adversary then uses the compromised ECU A to inject false messages that are claimed to be from ECU B.

This attack exploits two vulnerabilities of CAN that have been identified in the related literature [2], [12]. First, all ECUs have access to the same broadcast medium, allowing easily-compromised, low-priority ECUs (ECU A in Fig. 3) to listen to and impersonate higher-priority ECUs. Second, the lack of integrity checks means that spoofed messages from ECU A are not detected as long as the normal formatting and error-correction checks of CAN messages are passed.

IV. Clock Skew-Based IDS

Clock skew-based IDSs leverage the clock skew to uniquely fingerprint each ECU and detect masquerade attacks. Since CAN messages do not have transmit timestamps, approaches that require transmit timestamps for clock skew estimation such as [40]–[42] are not applicable. Similar to [43], clock skew-based IDSs on CAN buses instead exploit traffic periodicity [12]. Since almost all messages are transmitted periodically, the receiving IDS can monitor the inter-arrival times of a target message and estimate the clock skew of the transmitting ECU accordingly. We note that this approach is only viable for periodic message traffic. In the rest of this section, we will review the SOTA IDS and propose an NTP-based IDS.

A. Review of SOTA IDS

The SOTA IDS in [12] consists of a clock skew estimator and a CUSUM (Cumulative Sum [44])-based detector. The estimator tracks the clock skew from message inter-arrival times and feeds identification errors to the CUSUM for detection. We now describe the two components in more detail.

1) Clock Skew Estimator: Incoming periodic messages are processed in batches of size N to mitigate undesired impacts of quantization and other sources of noise in receive timestamps. Let $a_{k,i}$ be the arrival time of the i-th message in the k-th batch. The average offset of the k-th batch is given by

$$O_{avg}[k] = \frac{1}{N-1}\sum_{i=2}^{N}[a_{k,i} - (a_{k,1} + (i-1)\mu[k-1])], \quad (5)$$

where $\mu[k-1]$ is the mean inter-arrival time of the previous $((k-1)$-th) batch.

The absolute value of $O_{avg}[k]$ is added to the previous accumulated offset to compute the updated value,

$$O_{acc}[k] = O_{acc}[k-1] + |O_{avg}[k]|, \quad (6)$$

which is modeled as $O_{acc}[k] = S[k]t[k] + e[k]$, where $S[k]$, $t[k]$, and $e[k]$ denote the clock skew estimate in batch k, the elapsed time until the last message of the k-th batch, and the (unnormalized) identification error in batch k, respectively.

The estimated clock skew $S[k]$ is the output of the Recursive Least Squares (RLS) algorithm. Ideally, the identification error would converge to zero if clock skew is correctly estimated. Hence, a change in the identification error indicates a change in the clock skew. Besides, the rate of convergence is governed by a parameter $\lambda < 1$ (e.g., $\lambda = 0.9995$) that exponentially weighs past samples. More details are available in [12].

2) CUSUM-Based Detector: The detector tracks the mean μ_{CUSUM} and the standard deviation σ_{CUSUM} of identification errors that are used as reference (denoted as $\{e_{ref}[k]\}$). In batch k, $e[k]$ is first normalized as $e_n[k] = (e[k] - \mu_{\text{CUSUM}}[k-1])/\sigma_{\text{CUSUM}}[k-1]$. To mitigate the undesired impact of outliers, $e[k]$ will be considered as a reference error sample for updating μ_{CUSUM} and σ_{CUSUM} only if $e_n[k]$ is less than the preset update threshold γ (e.g., $\gamma = 4$), as noted in [12].

The detector then uses $e_n[k]$ to update the upper control limit L^+ and the lower control limit L^- in batch k as follows

$$L^+[k] = \max[0, L^+[k-1] + e_n[k] - \kappa], \quad (7)$$
$$L^-[k] = \max[0, L^-[k-1] - e_n[k] - \kappa], \quad (8)$$

where κ is a sensitivity parameter that reflects the number of standard deviations to be detected. The detector declares an attack if either the control limit, L^+ or L^-, exceeds the preset detection threshold Γ, which implies a sudden positive or negative shift in value, respectively. As the general rule of thumb for CUSUM, Γ is usually set to 4 or 5 [45], and the SOTA IDS chooses $\Gamma = 5$.

B. Proposed NTP-based IDS

We now present an adapted IDS that computes clock offset and clock skew as per the NTP specifications, which is referred to as the NTP-based IDS. The motivation for our NTP-based IDS is two-fold. First, we note that the metric in Eq. (5) is not consistent with the NTP definition in Eq. (1), since it does not calculate the time difference between the transmitter's clock and the reference clock. In addition, it is assumed that O_i is a

[1]On the CAN bus, messages with smaller ID levels (i.e., higher priorities) will be transmitted earlier in the event of collisions through a process called arbitration. A larger ID indicates a lower priority. See [37] for more details.

random variable and $\mathbb{E}[O_i - O_{i-1}] = 0$. It implies that $\mathbb{E}[O_i] = \mathbb{E}[O_j]$ for $i \neq j$, which does not hold in general since offsets accumulate over time (if $i \gg j$, $\mathbb{E}[O_i] \gg \mathbb{E}[O_j]$). Our second motivation is the widespread use and acceptance of NTP as a timing mechanism for real-time systems, which raises the question of whether NTP definitions of clocks can be used for intrusion detection as well. While both the SOTA IDS [12] and the proposed NTP-based IDS estimate the clock skew via the RLS and detect an attack via the CUSUM, they update average and accumulated offsets differently, as explained below.

Let T be the message period and $\hat{O}_i$ be the clock offset of the i-th period observed by the receiver. According to the NTP clock definitions (Section III-B) and the timing model (Section III-C), $\hat{O}_i$ is equal to

$$\hat{O}_i = T - (a_i - a_{i-1}) = O - \Delta\eta_i, \tag{9}$$

where $\Delta\eta_i = \eta_i - \eta_{i-1}$. In batch k, the average offset is

$$O_{avg}[k] = \frac{1}{N}\sum_{i=1}^{N} \hat{O}_{k,i} = T - \frac{a_{k,N} - a_{k,0}}{N}, \tag{10}$$

where $a_{k,0} = a_{k-1,N}$ is the receive timestamp of the last message in the previous ($(k-1)$-th) batch. The accumulated offset of the k-th batch is updated as follows

$$O_{acc}[k] = O_{acc}[k-1] + NO_{avg}[k]. \tag{11}$$

Eq. (5) and (10) highlight the differences in how the average offset is updated by the SOTA and NTP-based IDSs, respectively. Similarly, Eq. (6) and (11) show how the SOTA and NTP-based IDSs update the accumulated offset, respectively. As we will show in Section VII, the NTP-based IDS is more effective in detecting masquerade attacks than the SOTA IDS.

V. Proposed Cloaking Attack

In this section, we propose a new masquerade attack called the *cloaking attack*, in which the adversary adjusts the inter-transmission times of the spoofed messages in order to manipulate the estimated clock skew and bypass an IDS.

Consider a message transmitted by the targeted ECU B every T seconds in its own clock, which corresponds to every $\hat{T} = T/(1+S_B)$ seconds in the receiver R's clock, where S_B is B's clock skew. For the ease of discussion, we ignore offset deviations and the noise in arrival timestamps due to network delay and quantization. Then B's clock skew as estimated by R is given by $\hat{S} = (T - \hat{T})/\hat{T} = S_B$.

In a masquerade attack, the weak adversary prevents ECU B from transmitting the targeted message, and the strong adversary controlling ECU A transmits the spoofed message every T seconds as per A's local clock C_A. Hence, ECU R receives messages every $\hat{T}' = T/(1+S_A)$ seconds, as measured by C_R, where S_A is A's clock skew. The clock skew measured by ECU R will then be $\hat{S}' = (T - \hat{T}')/\hat{T}' = S_A$. Hence, if $S_A \neq S_B$, then the IDS will detect a change in the estimated clock skew after the adversary launches the attack.

The insight underlying our attack is that, while clock skew is a physical invariant, clock skew estimation in an IDS is based entirely on message inter-arrival times, which can be easily manipulated by the transmitter (i.e., the strong adversary controlling ECU A) adjusting the message inter-transmission times. Effectively, the adversary *cloaks* the skew of its hardware clock, thus motivating the term *cloaking attack*. Under the cloaking attack, instead of transmitting every T seconds, the compromised ECU A transmits every $\tilde{T} = T + \Delta T_0$ seconds, in order to match the clock skew observed at R.

We now discuss the choice of ΔT_0. Under the cloaking attack, the inter-arrival time observed by R is

$$\hat{T}'' = \frac{\tilde{T}}{1+S_A} = \frac{T + \Delta T_0}{1+S_A}$$

and the transmitter's clock skew estimated by R is

$$\hat{S}'' = \frac{T - \hat{T}''}{\hat{T}''} = \frac{S_A \cdot T - \Delta T_0}{T + \Delta T_0}. \tag{12}$$

Hence, to bypass the IDS, the adversary needs to choose ΔT_0 such that $\hat{S}'' = \hat{S}$, or equivalently $\hat{T}'' = \hat{T}$, which means

$$\Delta T_0 = \frac{(S_A - S_B)}{1+S_B} \cdot T = S_{AB} \cdot T = \frac{-S_{BA}}{1+S_{BA}} \cdot T, \tag{13}$$

where S_{AB} is A's clock skew relative to B's clock, and the last two equalities are due to Eq. (2) and Eq. (3), respectively.

Therefore, the message inter-transmission time $\tilde{T}$ would be

$$\tilde{T} = T + \Delta T_0 = T - \frac{S_{BA}}{1+S_{BA}}T = \frac{T}{1+S_{BA}},$$

which is the period of the message from B (weak adversary) measured by the local clock of A (strong adversary).

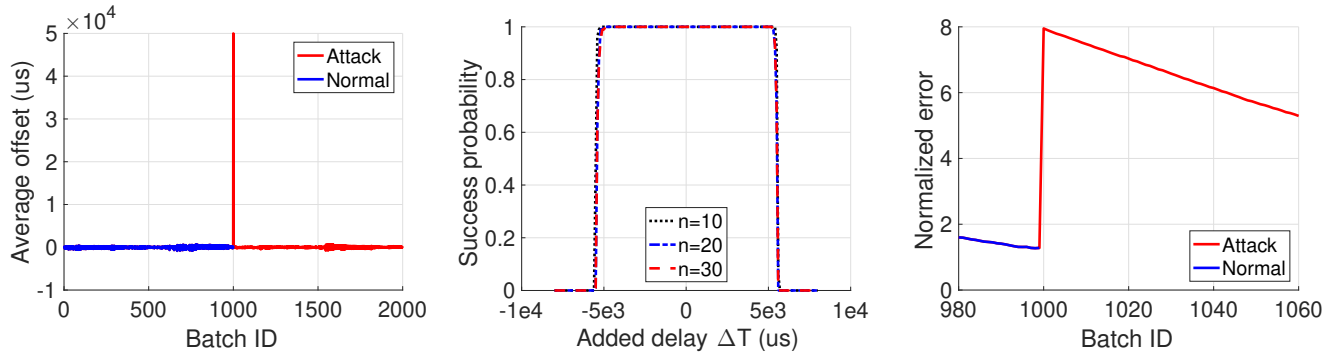
To summarize, the cloaking attack is performed as follows. After the adversary compromises two ECUs as strong and weak adversaries, the strong adversary estimates the period of the targeted message $\tilde{T}$ using its local clock. During the cloaking attack, the strong adversary transmits spoofed messages every $\tilde{T}$ seconds. While the preceding analysis ignores the noise in the system, our results in Section VII show that the cloaking attack is effective in a realistic environment.

In practice, however, the adversary may not be able to achieve the exact value of ΔT_0 due to hardware limitations and possible measurement inaccuracy. Let the total amount of the actual inter-transmission delay added by the adversary be $\Delta T + \Delta T_0$, where ΔT is the amount of deviation from ΔT_0. When ΔT is closer to zero, the attack will be successful with a higher probability. Hence, the attack success probability P_s is a function of ΔT (an attack parameter), parameters of the detector (e.g., λ, γ, and Γ), and the hardware platform. In order to predict and characterize the impact of the cloaking attack on a CAN bus and IDS without having to solely rely on extensive experiments, we aim to formally model P_s for both the SOTA and NTP-based IDSs, as presented below.

VI. Formal Analysis

A. Formal Analysis of SOTA IDS

In this section, we develop a formal model for the probability of a successful cloaking attack P_s as a function of parameters including the distribution of message inter-arrival times, the message period, the added inter-transmission delay, and the detection parameters of the IDS. We first present our modeling assumptions and observations. We then formulate our formal model and derive P_s for the SOTA IDS.



(a) (b) (c)

Fig. 4: Impact of the cloaking attack on the SOTA IDS. (a) Average offset as a function of batch ID. Only the first attack batch has a large average offset. (b) The attack success rates are roughly the same for $n = 10$, 20, and 30 attack batches. (c) The normalized identification error suddenly increases when the attack begins, and it then starts decreasing at an almost constant rate. Note that the figures are generated using the data for the 20 ms message 0x185 collected from the UW EcoCAR testbed. We set $N = 20$, $\gamma = 4$, $\Gamma = 5$, and $\kappa = 8$. The attack data is obtained by adding 5 ms to the inter-arrival times of the cloaking data collected from the UW EcoCAR testbed, and the attack starts from batch 1000.

1) Assumptions for SOTA IDS: For the SOTA IDS, the detection parameters including batch size N and CUSUM parameters Γ (the detection threshold) and κ (the sensitivity parameter) are known to the IDS. Since the IDS records all message arrival timestamps, it knows the message period T and can measure the mean μ and standard deviation σ of the message inter-arrival times.

Our analysis takes as input a "snapshot" of the IDS right before the attack that begins in the m-th batch. This means that the following parameters maintained by the IDS are readily available: the mean μ_{CUSUM} and standard deviation σ_{CUSUM} of the reference identification errors in the CUSUM, the average inter-arrival time $\mu[m-1]$, the accumulated offset $O_{acc}[m-1]$, the estimated skew $S[m-1]$, and the elapsed time $t[m-1]$.

2) Observations: Our modeling and analysis of the SOTA IDS are based on the following observations. As shown in Fig. 4(a), the first batch after the attack begins is the only batch that has a large average offset, and all subsequent batches have small offsets. This is because the average offset of the current batch is computed from the mean inter-arrival time of the previous batch (Eq. (5)). The first attack batch has a very different mean inter-arrival time from the last normal batch due to ΔT, whereas adjacent batches before and after the attack have close mean inter-arrival time.

As a result, for an attack that begins in the m-th batch[2], the identification error will be larger due to the sudden change in the mean inter-arrival time and will decrease over time due to clock skew update. In fact, we observe that the attack is usually either detected during the first tens of batches following the attack, or is not detected at all (Fig. 4(b)).

If we take a closer look at the first tens of batches after the attack begins, we observe a linear decrease in the normalized identification error (Fig. 4(c)). These observations motivate the following model of the normalized identification error $e_n[k]$ at batch $k \geq m$

$$e_n[k] \approx e_n[m] - \tau(k-m), \tag{14}$$

where $\tau > 0$ is a constant slope representing the rate of decrease of the normalized identification error.

3) Attack Success Probability: Based on the observations of Section VI-A2, we divide our formal analysis into three stages: 1) modeling the distribution of the normalized identification error in the first attack batch $e_n[m]$, 2) estimating the rate of decrease τ of the normalized identification error, and 3) computing the attack success probability from estimated distributions of $\{e_n[k] : k \geq m\}$. Each stage is described as follows.

Distribution of the normalized identification error in the first attack batch. We now examine the identification error $e[m]$ at the first attack batch m, which is

$$e[m] = O_{acc}[m] - S[m-1]t[m].$$

The clock skew value $S[m-1]$ is known, but the parameters $O_{acc}[m]$ and $t[m]$ are to be modeled. From the definitions of accumulated offset and elapsed time, we have

$$\begin{aligned} e[m] = {} & O_{acc}[m-1] + |O_{avg}[m]| \\ & - S[m-1](t[m-1] + T_{m,0} + a_{m,N} - a_{m,1}), \end{aligned} \tag{15}$$

where $T_{m,0}$ is the inter-arrival time between the last message of the previous ($(m-1)$-th) batch and the first message of the current (m-th) batch. Next, we will compute the mean and standard deviation of $e[m]$.

Based on our timing model (Section III-C), the average offset under an attack with a delay of ΔT (i.e., the equivalent

[2]We assume that the first attack message appears as the 1st message of the m-th batch.

total amount of added delay is $\Delta T + \Delta T_0$) is

$$\begin{aligned} O_{avg}[m] &= \frac{1}{N-1}\sum_{i=2}^{N}[(i(T+\Delta T-O)+\eta_{m,i}) \\ &\quad -((T+\Delta T-O)+\eta_{m,1}+(i-1)\mu[m-1])] \\ &= \frac{1}{N-1}\sum_{i=2}^{N}[(i-1)(\mu+\Delta T-\mu[m-1]) \\ &\quad +(\eta_{m,i}-\eta_{m,1})], \end{aligned}$$

where $\mu = T - O$ is the mean inter-arrival time before an attack[3]. Although the statistics of η after the attack may be different from those before the attack due to different characteristics of transmitting ECUs, such information is not available at batch $(m-1)$. Therefore, we assume the same statistics of η before and after the attack, namely, $\eta_{m,i} \sim N(d, \sigma_\eta^2)$ for $1 \le i \le N$, which yields

$$O_{avg}[m] \sim N\left(\frac{N}{2}(\mu+\Delta T-\mu[m-1]), \frac{N}{N-1}\sigma_\eta^2\right). \quad (16)$$

Since $\sigma_\eta^2 = \sigma^2/2$ (Section III-C), the variance of $O_{avg}[m]$ is also equal to $\frac{N}{2(N-1)}\sigma^2$, where σ is the standard deviation of inter-arrival times. For ΔT sufficiently large, the ΔT term will dominate $(\eta_{m,i} - \eta_{m,1})$, and hence we have

$$\begin{aligned} |O_{avg}[m]| \approx \frac{1}{N-1}\sum_{i=2}^{N}[(i-1)\cdot|\mu+\Delta T-\mu[m-1]| \\ +(\eta_{m,i}-\eta_{m,1})]. \end{aligned} \quad (17)$$

Next, we can substitute the $|O_{avg}[m]|$ term in Eq. (15) with Eq. (17) and compute the mean and standard deviation of $e[m]$, as described in the following lemma.

Lemma 1. *Under the assumption (17), the identification error $e[m]$ of the first attack batch is Gaussian with mean*

$$\begin{aligned} \mu_e &= O_{acc}[m-1] + \frac{N}{2}(|\mu+\Delta T-\mu[m-1]|) \\ &\quad - S[m-1](t[m-1]+T_{m,0}+(N-1)(\mu+\Delta T)) \end{aligned} \quad (18)$$

and variance

$$\sigma_e^2 = \frac{1}{2}\left(\frac{N-2S[m-1]}{N-1} + 2S[m-1]^2 - 2S[m-1]\right)\sigma^2.$$

A proof can be found in Appendix A.

The distribution of the normalized identification error in the first attack batch is Gaussian and satisfies

$$e_n[m] = \frac{e[m]-\mu_{\text{CUSUM}}}{\sigma_{\text{CUSUM}}} \sim N\left(\frac{\mu_e-\mu_{\text{CUSUM}}}{\sigma_{\text{CUSUM}}}, \frac{\sigma_e^2}{\sigma_{\text{CUSUM}}^2}\right).$$

After obtaining the distribution of the normalized identification error in the first attack batch, our next task is to model the rate of decrease τ of the normalized identification error $e_n[k]$ in Eq. (14), which will give us an approximation of $e_n[k]$ for $k \ge m+1$.

[3]Strictly speaking, the resulting offset due to the added delay of ΔT is $O' = (T+\Delta T)/T \cdot O$. However, ΔT is usually much smaller than T, and thus we can approximate O' as O.

Rate of decrease of the normalized identification error. According to Eq. (15), the identification error $e[k+1]$ after an attack begins (i.e., $k \ge m$) is given by

$$\begin{aligned} e[k+1] &= O_{acc}[k+1] - S[k]t[k+1], \\ &= O_{acc}[k] + |O_{avg}[k+1]| \\ &\quad - S[k](t[k] + T_{k+1,0} + (a_{k+1,N} - a_{k+1,1})), \end{aligned}$$

where $T_{k+1,0} \approx \mu + \Delta T$ is the inter-arrival time between the last message in the k-th batch and the first message of the $(k+1)$-th batch during the attack.

Since skew updating is slow in the first tens of batches due to the slow convergence of the RLS algorithm, we may assume that $S[k] = S$ is a constant. Then we have

$$\begin{aligned} e[k+1] &= O_{acc}[k] + |O_{avg}[k+1]| - St[k] \\ &\quad - S((\mu+\Delta T) + (a_{k+1,N} - a_{k+1,1})) \\ &= e[k] + |O_{avg}[k+1]| \\ &\quad - S(N(\mu+\Delta T) + (\eta_{k+1,N} - \eta_{k+1,1})). \end{aligned}$$

According to Eq. (16), the average offset $O_{avg}[k+1]$ is Gaussian with mean $\frac{N}{2}(\mu+\Delta T-\mu[k])$ and variance $\frac{N}{2(N-1)}\sigma^2$. Although the value of $\mu[k]$ for $k \ge m$ is not available at batch $(m-1)$, we have $\mathbb{E}(\mu[k]) = \mu+\Delta T$, which means $O_{avg}[k+1]$ can be approximated as zero.

Therefore, we can derive a linear approximation to $e[k]$ by taking the expectation of $(e[k+1]-e[k])$. Since $|O_{avg}[k+1]|$ is the absolute value of a Gaussian random variable with mean zero and variance $\frac{N}{2(N-1)}\sigma^2$, we have

$$\mathbb{E}(|O_{avg}[k+1]|) = \sqrt{\frac{N}{2(N-1)}\sigma^2}\cdot\sqrt{\frac{2}{\pi}} = \sigma\sqrt{\frac{N}{\pi(N-1)}}.$$

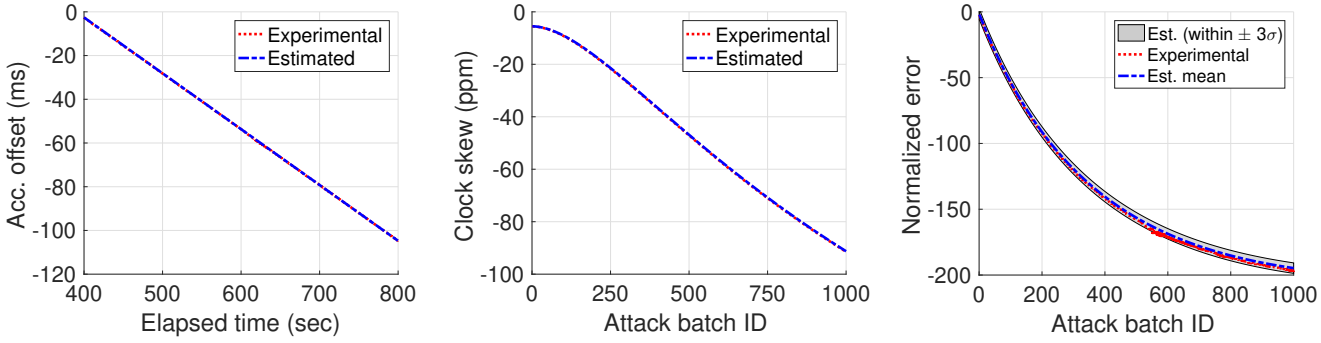
Since the normalized identification error is computed as $e_n[k] = (e[k] - \mu_{\text{CUSUM}})/\sigma_{\text{CUSUM}}$, the rate of decrease τ of $e_n[k]$ can be approximated as

$$\begin{aligned} \tau &\approx |\mathbb{E}(e_n[k+1]-e_n[k])| = \left|\mathbb{E}\left(\frac{e[k+1]-e[k]}{\sigma_{\text{CUSUM}}}\right)\right| \\ &= \left|\frac{1}{\sigma_{\text{CUSUM}}}\left(\sigma\sqrt{\frac{N}{\pi(N-1)}} - S(N(\mu+\Delta T))\right)\right|. \end{aligned}$$

Note that the fixed σ_{CUSUM} is used, since $e_n[k]$ is usually larger than γ and thus σ_{CUSUM} will not be updated.

Now that we have distributions of normalized identification errors $\{e_n[k] : k \ge m\}$, we can compute the distribution of the maximum value of control limits L^+ and L^-, and derive the attack success probability.

Computation of the attack success probability. In order to derive the attack success probability, let us take a closer look at how the control limits are updated. Without loss of generality, we consider positive ΔT and assume that the upper control limit L^+ is zero before the attack. From Eq. (7), we can see that if $e_n[m] \ge \kappa+\Gamma$, the attack will be detected immediately in the first batch; if $e_n[m] \le \kappa$, it will not be detected at all. If $e_n[m]$ lies in $(\kappa, \kappa+\Gamma)$ and $L^+[k] = \sum_{k \ge m}(e_n[l]-\kappa)$ is greater than Γ for some k, the attack will still be detected after several batches. Hence, we can first compute the maximum value of L^+, which depends on $e_n[m]$ and τ, and then



(a) (b) (c)

Fig. 5: Experimental versus estimated (a) accumulated offset and elapsed time, (b) clock skew, and (c) normalized identification error. The estimated values match closely with the experimental values. Note that figures are generated using data for the 20ms message 0x185 collected from our testbed with $N = 20$, $\gamma = 4$, $\Gamma = 5$, $\kappa = 8$, and $\Delta T = 5\ \mu s$.

relate the attack success probability $Pr(L^{+}_{\max} \leq \Gamma)$ to the distribution of $e_n[m]$, as shown in the following theorem.

Theorem 1. *The attack success probability satisfies*

$$P_s = Pr\left(\frac{\tau - \sqrt{\tau^2 + 8\tau\Gamma}}{2} - \kappa \leq e_n[m] \leq \frac{-\tau + \sqrt{\tau^2 + 8\tau\Gamma}}{2} + \kappa\right). \quad (19)$$

The proof can be found in Appendix B.

By Theorem 1, we can see that the attack success probability can be computed by evaluating the cumulative density function of a Gaussian random variable.

B. Formal Analysis of NTP-Based IDS

We then formally analyze the probability of a successful cloaking attack for the NTP-based IDS, given the system parameters immediately before the attack.

1) Assumptions for NTP-Based IDS: For the NTP-based IDS, the batch size N and CUSUM parameters including γ (the update threshold), Γ (the detection threshold), κ (the sensitivity parameter) and λ (the parameter in the RLS), are known to the IDS. Since the IDS records the receive timestamps of the target message, it knows the period T and can also measure the mean μ and standard deviation σ of inter-arrival times.

As mentioned in Section IV, the NTP-based IDS tracks the accumulated offset $O_{acc}[k]$ and elapsed time $t[k]$ in each batch k, and maintains the reference identification errors. Hence, it is reasonable to assume that the values of $\{O_{acc}[k] : k < m\}$, $\{t[k] : k < m\}$, and $\{e_{ref}[k] : k < m\}$ are known to the NTP-based IDS prior to the attack.

2) Observations: Our modeling and analysis of the NTP-based IDS are based on the following observations. First, if the attack with an added delay of ΔT starts in the k-th batch, the resulting $O_{acc}[k]$, $t[k]$, and $e[k]$ can be estimated from μ, T, $S[k-1]$, and ΔT. Second, although the IDS keeps track of the slowly changing clock skew via the RLS based on newly obtained $t[k]$ and $O_{acc}[k]$, the output of the RLS converges to that of a non-RLS estimator that minimizes the weighted mean squared error. Third, with the estimated value of $e[k]$, the IDS can further estimate the CUSUM statistics following its updating rule, as well as the mean value and distribution of normalized errors.

3) Attack Success Probability: Based on the observations in Section VI-B2, we divide our formal analysis into four stages: 1) estimating the accumulated offset $O_{acc}[k]$ and the elapsed time $t[k]$ after the attack begins at batch m, 2) approximating the clock skew $S[k]$ estimated by the RLS, 3) modeling the distributions of normalized identification errors $\{e_n[k] : k \geq m\}$, and 4) computing the probability of control limits exceeding Γ to obtain the attack success probability.

Accumulated offset and elapsed time. For the NTP-based IDS, the accumulated offset before the attack is

$$O_{acc}[k] = \sum_{i=1}^{k} NO_{avg}[i] = \sum_{i=1}^{k} N\left(T - \frac{a_{i,N} - a_{i-1,N}}{N}\right) = kNO - (\eta_{k,N} - \eta_{0,N}), \quad (20)$$

where $a_{0,N}$ is the arrival timestamp of the last message in the initialization batch, and $O = T - \mu$ is the average offset in each period T. The elapsed time is

$$t[k] = a_{k,N} - a_{0,N} = kN(T - O) + \eta_{k,N} - \eta_{0,N}. \quad (21)$$

We assume that the attack starts from the first message of batch m, and the inter-arrival time between the last normal message and the first attack message is roughly equal to $\mu + \Delta T$. Then for $k \geq m$, we have

$$\begin{aligned} t[k] &= kN(T - O) + (k - m + 1)N\Delta T + \eta_{k,N} - \eta_{0,N} \\ &= t[m-1] + (k - m + 1)N(T - O + \Delta T) \\ &\quad + \eta_{k,N} - \eta_{m-1,N}. \end{aligned} \quad (22)$$

Since $O_{acc}[k] = kNT - t[k]$, we also have

$$\begin{aligned} O_{acc}[k] &= O_{acc}[m-1] - (k - m + 1)N(-O + \Delta T) \\ &\quad - (\eta_{k,N} - \eta_{m-1,N}). \end{aligned} \quad (23)$$

Note that in the above equations, the amount of network delay and noise as captured by $\eta_{m-1,N}$ is given at batch $(m-1)$, and thus $\eta_{k,N}$ is the only random variable.

With more attack batches arriving, the estimated clock skew will gradually change over time. Hence, it is important to model the process of clock skew updating, which is our next step of modeling.

Approximation of the estimated clock skew. While the RLS is an online algorithm that recursively updates the clock skew estimate with non-linear equations, it has been shown in [46] that the clock skew estimated via the RLS would converge to the value S that minimizes the following quadratic function,

$$J_k(S) = \sum_{i=1}^{k} \lambda^{k-i}(O_{acc}[i] - S \cdot t[i])^2, \tag{24}$$

where $\lambda < 1$ is the parameter in the RLS, and the optimal value is given by

$$\hat{S}[k] = \arg\min_S J_k(S) = \frac{\sum_{i=1}^{k} \lambda^{k-i} O_{acc}[i] \cdot t[i]}{\sum_{i=1}^{k} \lambda^{k-i} t^2[i]}. \tag{25}$$

Let the mean of $t[k]$ in Eq. (22) and $O_{acc}[k]$ in Eq. (23) be $\hat{t}[k]$ and $\hat{O}_{acc}[k]$, respectively. Given $\hat{t}[k]$ and $\hat{O}_{acc}[k]$, we can estimate the output of RLS as $\hat{S}[k]$ based on Eq. (25).

As shown in Fig. 5(a) and Fig. 5(b), the estimated values of accumulated offset, elapsed time, and clock skew are closely matched with the experimental values.

Distribution of the normalized identification errors. With the estimated clock skew values $\{\hat{S}[k]\}$, the identification error $e[k]$ is given as

$$\begin{aligned} e[k] &= O_{acc}[k] - \hat{S}[k-1]t[k] \\ &= (\hat{O}_{acc}[k] - \eta_{k,N}) - \hat{S}[k-1](\hat{t}[k] + \eta_{k,N}) \\ &= \hat{e}[k] - (1 + \hat{S}[k-1])\eta_{k,N}, \end{aligned}$$

where $\hat{e}[k] = \hat{O}_{acc}[k] - \hat{S}[k-1]\hat{t}[k]$. Since $\eta_{k,N}$ is Gaussian, the identification error $e[k]$ is also Gaussian with mean $\hat{e}[k]$ and variance $(1 + \hat{S}[k-1])^2\sigma_\eta^2$.

In order to estimate the distribution of $e_n[k]$, we need to model the updating process of CUSUM statistics, i.e., $\hat{\mu}_{\text{CUSUM}}$ and $\hat{\sigma}_{\text{CUSUM}}$. Hence, given $\{\hat{e}[k]\}$, we can compute $\hat{e}_n[k] = (\hat{e}[k] - \hat{\mu}_{\text{CUSUM}}[k-1])/\hat{\sigma}_{\text{CUSUM}}[k-1]$. If $|\hat{e}_n[k]| \leq \gamma$, we add $\hat{e}[k]$ to $\{e_{ref}[k]\}$ and re-compute $\hat{\mu}_{\text{CUSUM}}[k]$ and $\hat{\sigma}_{\text{CUSUM}}[k]$ from $\{e_{ref}[k]\}$. Then we increment k by 1 and repeat the above steps.

Since $e_n[k] = (e[k] - \hat{\mu}_{\text{CUSUM}}[k-1])/\hat{\sigma}_{\text{CUSUM}}[k-1]$, it implies

$$e_n[k] \sim N\left(\frac{\hat{e}[k] - \hat{\mu}_{\text{CUSUM}}[k-1]}{\hat{\sigma}_{\text{CUSUM}}[k-1]}, \frac{(1 + \hat{S}[k-1])^2\sigma_\eta^2}{\hat{\sigma}_{\text{CUSUM}}[k-1]^2}\right).$$

As shown in Fig. 5(c), the estimated mean values of $e_n[k]$ match closely with the experimental values. Based on the distributions of $\{e_n[k] : k \geq m\}$ derived above, we can now compute the attack success probability.

CUSUM analysis. Let the probability density function of $e_n[k]$ be f_k, and the number of attack batches used for detection be n. We assume that $\kappa \geq \Gamma$, which is consistent with the NTP-based IDS and our simulations. A detection takes place in the k-th attack batch if $L^+[k] > \Gamma$ or $L^-[k] > \Gamma$. Let $\alpha = \min\{k : \max\{L^+[k], L^-[k]\} > \Gamma\}$, which is the attack batch ID when control limits first exceed the detection threshold. In other words, if $\alpha > n$, it means that the attack is not detected within n batches. Hence, the attack success probability is equal to $Pr(\alpha > n)$, and the following lemma shows how to compute

$$g_{n,k}(z^+, z^-) \triangleq Pr(\alpha > n | L^+[k] = z^+, L^-[k] = z^-).$$

Lemma 2. *The probability of a successful cloaking attack for the CUSUM-based detector satisfies*

$$\begin{aligned} g_{n,k}(z^+, z^-) = &\int_{z^- - \kappa - \Gamma}^{z^- - \kappa} g_{n,k+1}(0, z^- - r - \kappa) f_k(r)\, dr \\ &+ g_{n,k+1}(0,0) Pr(e_n[k] \in [z^- - \kappa, \kappa - z^+]) \\ &+ \int_{\kappa - z^+}^{\kappa - z^+ + \Gamma} g_{n,k+1}(z^+ + r - \kappa, 0) f_k(r)\, dr. \end{aligned}$$

From Lemma 2, we can take a discrete approximation of $g_{n,k}(z^+, z^-)$ as

$$\begin{aligned} g_{n,k}\left(\frac{i\Gamma}{m}, \frac{j\Gamma}{m}\right) \approx &\frac{\Gamma}{m} \sum_{l=0}^{m} g_{n,k+1}\left(0, \frac{l\Gamma}{m}\right) f_k\left(\frac{(j-l)\Gamma}{m} - \kappa\right) \\ &+ g_{n,k+1}(0,0) Pr(e_n[k] \in [z^- - \kappa, \kappa - z^+]) \\ &+ \frac{\Gamma}{m} \sum_{l=0}^{m} g_{n,k+1}\left(\frac{l\Gamma}{m}, 0\right) f_k\left(\frac{(l-i)\Gamma}{m} + \kappa\right). \end{aligned}$$

A proof can be found in Appendix C.

Therefore, the value of $g_{n,k}$, that is, the probability of a successful cloaking attack within n attack batches predicted at the k-th attack batch ($k \leq n$), can be computed as a linear function of the values of $g_{n,k+1}$. The attack success probability is equal to $g_{n,0}(0,0)$.

VII. Evaluation

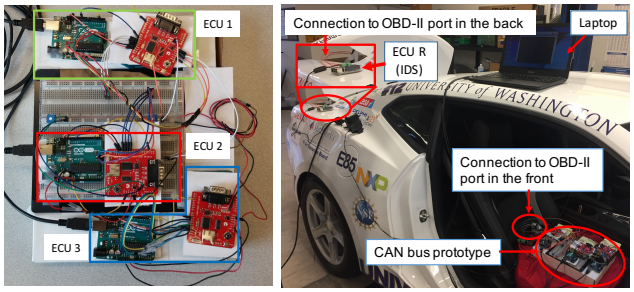
In this section, we evaluate the proposed cloaking attack on two CAN bus testbeds and demonstrate that the cloaking attack is able to bypass both the SOTA and NTP-based IDSs. We then validate our formal analysis through extensive experiments.

A. Testbeds

We build two CAN bus testbeds: a CAN bus prototype and a CAN testbed on a real vehicle (the UW EcoCAR, a 2016 Chevrolet Camaro [47]). Compared with the prototype that consists of three ECUs, the UW EcoCAR hosts 8 stock ECUs and two experimental ECUs. A total of 2500+ messages with 89 different IDs are being exchanged every second.

1) CAN Bus Prototype: As shown in Fig. 6(a), each ECU on the CAN bus prototype consists of an Arduino UNO board and a Sparkfun CAN bus shield that uses a Microchip MCP2515 CAN controller with a MCP2551 CAN transceiver. The bus speed is set to 500 Kbps as in typical CAN buses.

2) UW EcoCAR testbed: The CAN bus prototype is connected to the CAN bus of the UW EcoCAR via the On-Board Diagnostics (OBD-II) port to build the UW EcoCar testbed (Fig. 6(b)). During our experiments, the UW EcoCAR was in the park mode in an isolated and controlled environment for safety purposes, but all ECUs were functional and actively exchange CAN messages. We noticed that ECUs in the park mode had very close clock skews as in the drive mode.



(a) CAN bus prototype (b) UW EcoCAR testbed

Fig. 6: Setup of CAN bus testbeds. (a) The CAN bus prototype consists of three testbed ECUs, each of which consists of an Arduino board and a Sparkfun CAN bus shield. (b) The CAN bus prototype and Raspberry Pi-based ECUs are connected to the CAN bus of the UW EcoCAR via the OBD-II ports to build the UW EcoCAR testbed.

Due to the large CAN traffic and limited computing capability, Arduino-based ECUs are not able to log all CAN messages on the bus or transmit high frequency messages. Therefore, we build additional ECUs that consist of a Raspberry Pi 3 and a PiCAN 2 board and used SocketCAN [48] to enable the interaction between the added ECUs and the UW EcoCAR.

B. Evaluation of Cloaking Attack

We first demonstrate and evaluate the cloaking attack on both the CAN bus prototype and the UW EcoCAR testbed.

1) Setup: On the CAN bus prototype, ECU 1 acts as the IDS that logs all messages, ECU 2 is the targeted ECU that transmits message 0x11 every 100 ms (10 Hz), and ECU 3 is the strong adversary that impersonates ECU 2. On the UW EcoCAR testbed, a stock ECU that transmits message 0x184 every 100 ms is treated as the targeted ECU and the same ECU 3 acts as the strong adversary that injects spoofed messages.

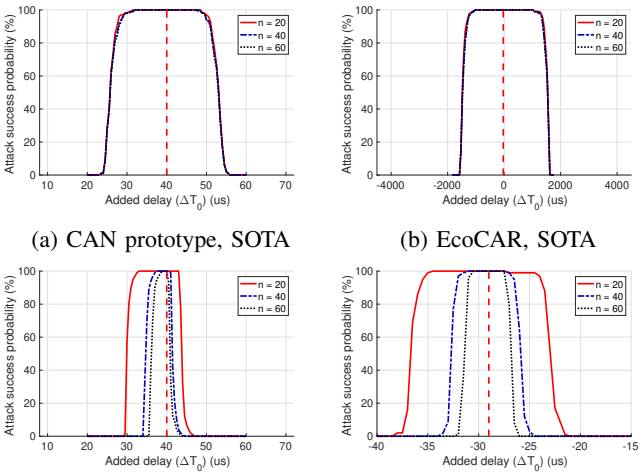
When launching the cloaking attack, the impersonating ECU 3 transmits every 100040 μs ($\Delta T_0 = 40$ μs) to spoof message 0x11 on the CAN bus prototype and every 99971 μs ($\Delta T_0 = -29$ μs[4]) to spoof message 0x184 on the UW EcoCAR testbed. During our experiments, we collected a total of 8.5 hours of attack data from the CAN bus prototype and the UW EcoCAR testbed separately.

We set batch size $N = 20$ for both the SOTA and the NTP-based IDSs. For the SOTA IDS, the update threshold γ is 3 and the detection threshold Γ is 5, which is consistent with [12]. For the NTP-based IDS, we use $\gamma = 4$ and $\Gamma = 5$. For the data collected from the CAN bus prototype, the sensitivity parameter κ is set to 5 for both IDSs, while it is set to 8 for the UW EcoCAR data to avoid false alarms.

To simulate the cloaking attack, the IDS is fed with 1000 batches of normal data, followed by n batches of attack data in each experiment[5]. An attack is successful if it is undetected by the IDS and fails otherwise. A total of 100 independent experiments are performed to compute the attack success probability P_s.

[4]While Arduino's time resolution is 4 μs , we set ΔT_0 to -28 μs and changed it to -32 μs every five messages so that $\Delta T_0 \approx -29$ μs on average.

[5]We assume perfect timing for the cloaking attack, that is, the first attack message is received at the next expected time instant of the targeted message.



(a) CAN prototype, SOTA (b) EcoCAR, SOTA

(c) CAN prototype, NTP-based (d) EcoCAR, NTP-based

Fig. 7: Attack success probability on the SOTA IDS and the NTP-based IDS on the CAN bus prototype and the UW EcoCAR testbed with message period 100 ms. For the ΔT_0 values achieved in our hardware experiments (red dashed line), the cloaking attack was successful in all test cases.

2) Results: For the ΔT_0 values achieved in our evaluation, P_s is 1 against both the SOTA and NTP-based IDSs (Fig. 7, dashed line). In order to gain additional insight into the performance of each IDS under cloaking attack, we generate additional datasets by adding different values of ΔT_0 to the message inter-arrival times and then analyze both IDSs using the new datasets.

In order to quantify the effectiveness of an IDS against the masquerade (cloaking) attack, we define a metric called ϵ-*Maximum Slackness Index* (MSI), which measures the interval of ΔT_0 that an adversary can introduce while remaining undetected with a probability of $(1-\epsilon)$. We first let $P_s(\Delta T_0)$ be the attack success probability when the added delay is ΔT_0. We define the upper and lower limits of ΔT_0 for a successful attack as $(\Delta T_0)_{\max}(\epsilon) = \max\{\Delta T_0 : P_s(\Delta T_0) > 1-\epsilon\}$ and $(\Delta T_0)_{\min}(\epsilon) = \min\{\Delta T_0 : P_s(\Delta T_0) > 1-\epsilon\}$, respectively. We then define ϵ-MSI $= (\Delta T_0)_{\max}(\epsilon) - (\Delta T_0)_{\min}(\epsilon)$. Intuitively, a smaller value of ϵ-MSI signifies a more effective detector and less freedom for the adversary, since the adversary's clock skew must closely match with that of the targeted ECU in order to remain undetected.

On the CAN bus prototype, with $n = 20$ and $\epsilon = 0.05$, the ϵ-MSI value for the SOTA IDS is 22.5 μs (Fig. 7(a)), but only 11.5 μs for the NTP-based IDS (Fig. 7(c)). Hence, it is much easier for the cloaking attack to bypass the SOTA IDS than the NTP-based IDS. We also found that increasing n has little impact on ϵ-MSI for the SOTA IDS, which is 20.5 μs for $n = 40$ or 60, but significantly impacts ϵ-MSI of the NTP-based IDS, which varies from 11.5 μs to 2.5 μs as n is increased from 20 to 60. This result suggests that the performance of the NTP-based IDS improves over the attack duration. Another interesting observation is that the P_s curves

TABLE II: Selected subset of representative messages from Electronic Brake Control Module (EBCM), Electronic Power Steering (EPS), Electronic Parking Brake (EPB), and Body Control Module (BCM).

Message ID	Period (ms)	Transmitter	Data Size (hours)
0x0D1	10	EBCM	0.53 hours
0x185	20	EBCM	1.01 hours
0x1FC	50	EBCM	2.01 hours
0x184	100	EPS	4.44 hours
0x22A	100	EPB	3.84 hours
0x3C9	100	BCM	4.48 hours

are skewed instead of symmetric. This is because when the Arduino-based ECU starts operating, its clock skew slowly decreases due to the temperature change in hardware. As a result, the IDS tends to overestimate the clock skew, and is more sensitive to a larger positive delay (that would further decrease the clock skew).

ϵ-MSI for the SOTA IDS increases significantly for a real vehicle, as shown in Fig. 7(b), due to the significantly heavier CAN traffic compared to the prototype, which reduces the effectiveness of the detection. As an example, a cloaking attack with ΔT_0 between -1029 μs and 1021 μs can bypass the SOTA IDS with 100% probability regardless of n. For the NTP-based IDS with $\epsilon = 0.01$, ϵ-MSI is 10.5 μs for $n = 20$ and 3 μs for $n = 60$. Hence, in the real vehicle, as in the CAN prototype, the NTP-based IDS is more effective in detecting masquerade attacks than the SOTA IDS. The proposed cloaking attack, however, is still able to thwart both detection schemes when ΔT_0 is chosen to be within the interval $[(\Delta T_0)_{min}(\epsilon), (\Delta T_0)_{max}(\epsilon)]$.

C. Evaluation of Formal Analysis

We now validate the proposed formal models using the data collected from the UW EcoCAR testbed.

Data Collection. Since it is both labor and time intensive to collect data for all periodic messages, we select a subset of 6 representative messages with different periods, message ID levels, and transmitting ECUs, which are listed in Table II.

When collecting the cloaking attack data for a targeted message of period T, the strong adversary (a Raspberry Pi-based ECU A that is connected to the OBD-II port) transmits messages every T seconds, using a non-conflicting message ID to avoid any undesirable impact on the vehicle. The IDS at ECU R records the timestamps of all received messages. The targeted and spoofed messages will be filtered and later used as the normal and attack data, respectively.

In order to determine a suitable amount of added delay for the cloaking attack, we first set ΔT_0 to be the difference between the average inter-arrival time of the targeted message observed by ECU A and the nominal period T (Section V). We then experimentally tune ΔT_0 so that the observed clock skews at the IDS residing in ECU R become the same. Hence, the collected data corresponds to the cloaking attack with $\Delta T = 0$, where ΔT is the difference between the actual delay and the ideal delay ΔT_0 (Section V).

Post-processing. Due to the limited capability of the receiver to capture all messages in the vehicle, some messages are missed sporadically. To maintain message periodicity, missing messages are inserted during post-processing.

Setup. To obtain the predicted attack success probability curve, we feed 1000 batches of normal data to the IDS and computed the attack success probability P_s for different ΔT using equations in Sections VI-A and VI-B. For consistency, we set $N = 20$, $\gamma = 4$, $\Gamma = 5$, and $\kappa = 8$ for the IDS in all experiments, and there are no false alarms.

To obtain the experimental attack success probability curve, the same normal data is fed to the IDS, followed by n batches of attack data in each experiment. A total of 100 independent experiments are performed, and the ratio of experiments where the attack is successful (undetected) is computed as P_s for $\Delta T = 0$.

Since collecting data for each ΔT value would be prohibitively time-consuming, we generate attack data for other ΔT values by adding a fixed offset equal to ΔT to the inter-arrival times of the collected data, as in Section VII-B. By repeating the previous process, we obtain the experimental attack success probability curve.

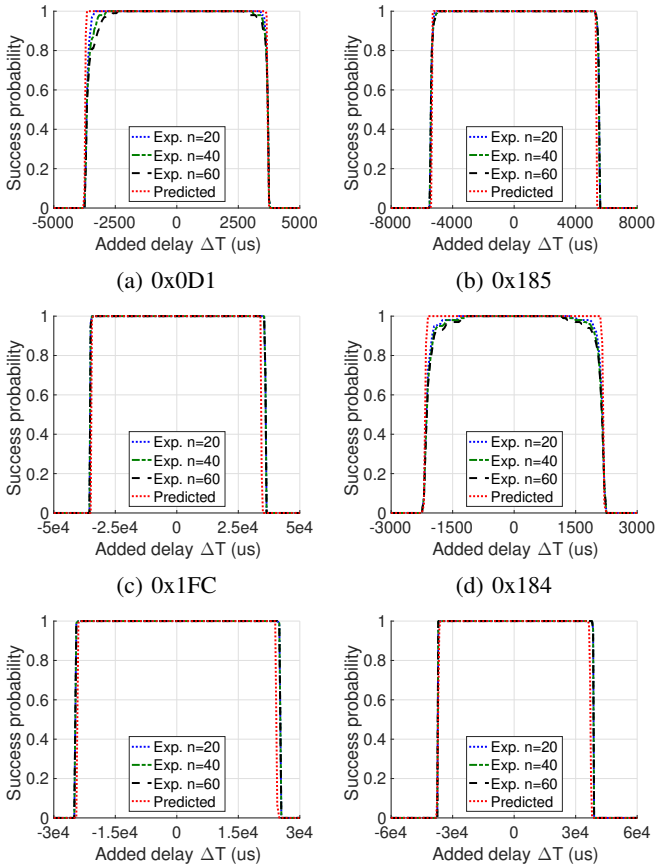
Metric for quantifying the prediction error. In order to quantify the prediction error of the proposed models, we define a metric called *Area Deviation Error (ADE)* as

$$\text{ADE} = \frac{\int_{-\infty}^{\infty} |P_{s,pred.}(\Delta T) - P_{s,exp.}(\Delta T)| d\Delta T}{\int_{-\infty}^{\infty} P_{s,exp.}(\Delta T) d\Delta T} \times 100\%, \tag{26}$$

where $P_{s,pred.}(\Delta T)$ and $P_{s,exp.}(\Delta T)$ are the predicted and experimental attack success probabilities, respectively. In other words, ADE is the ratio of the absolute difference of the areas under the predicted and experimental attack success probability curves to the area under the experimental curve. Hence, a smaller ADE value implies a smaller deviation from the experimental curve (the ground truth) and better prediction accuracy. Note that the ADE can be larger than 100%, when the area under the experimental curve is small.

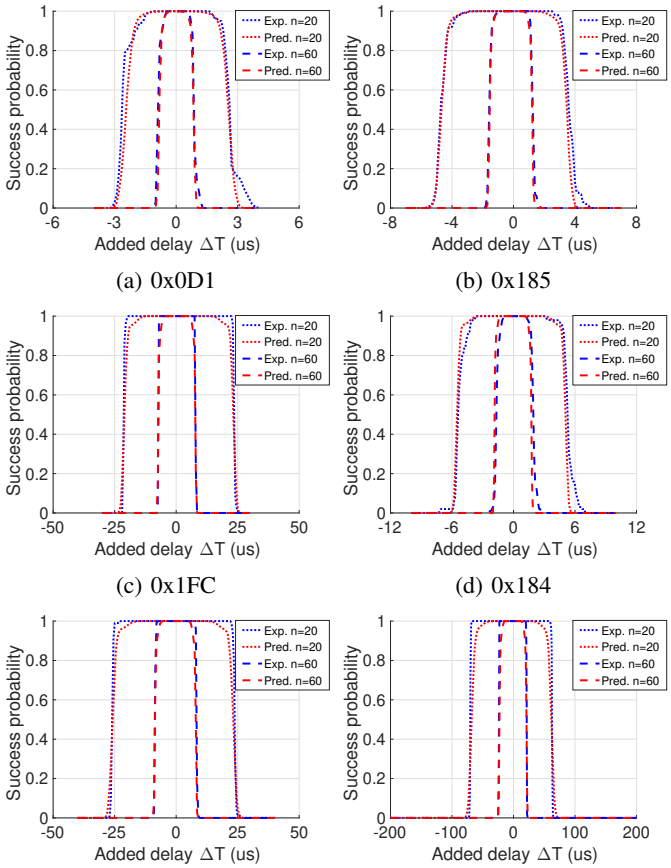
Evaluation of SOTA IDS analysis. As shown in Fig. 8, we can see a close match between the predicted and experimental curves. For a given ΔT, the proposed model provides the same attack success probability for different n. This is because our analysis focuses on modeling the normalized identification error in the first attack batch and its rate of decrease using the system parameters, which are independent of n. In fact, the closeness of the curves agrees with the observation that the SOTA IDS is insensitive to n.

In addition, for messages like 0x0D1 and 0x184, we observe small discrepancies at the corners of the curves, which may caused by outliers in collected data. In the meanwhile, the assumption of linearly decreasing normalized identification error may also cause the proposed model to overestimate the attack success probability. We observe that for some messages with less noise in timestamps, the normalized identification error of the SOTA IDS may not strictly decrease. In this case, the error lasts for a longer duration and causes the attack to be detected at a later time, which could explain why the experimental attack success probability is smaller than the predicted value. Improving our formal model for the SOTA IDS for messages with less noise is left as future work.



(a) 0x0D1 (b) 0x185 (c) 0x1FC (d) 0x184 (e) 0x22A (f) 0x3C9

Fig. 8: Experimental versus predicted attack success probability curves for the SOTA IDS with different numbers of attack batches n for different messages. The predicted curves are closely matched with the experimental curves for all messages. Note that the proposed model for the SOTA IDS is insensitive to n, thus providing the same prediction for different n values.



(a) 0x0D1 (b) 0x185 (c) 0x1FC (d) 0x184 (e) 0x22A (f) 0x3C9

Fig. 9: Experimental versus predicted attack success probability curves for the NTP-based IDS with $n = 20$ and 60 batches of attack data for different messages. The predicted curves match well in general with the experimental curves for all messages and have a closer agreement for larger n.

TABLE III: ADE (%) between the predicted and experimental attack success probability curves for the SOTA and NTP-based IDSs with different numbers of attack batches.

Msg ID	SOTA IDS			NTP-based IDS		
	20	40	60	20	40	60
0x0D1	1.5	2.3	3.1	6.1	7.7	6.9
0x185	2.1	2.2	2.2	3.3	4.0	3.8
0x1FC	3.2	3.1	3.1	3.4	3.6	2.8
0x184	2.9	3.8	4.5	5.3	9.3	11.9
0x22A	2.9	2.9	2.9	3.7	3.9	3.9
0x3C9	2.4	2.4	2.4	5.6	5.2	5.1
Mean	2.5	2.8	3.0	4.6	5.6	5.7

Evaluation of NTP-based IDS analysis. As shown in Fig. 9, we can also see a close agreement in shape between the predicted and experimental attack success probability curves. The fact that the attack success probability decreases as n increases for the NTP-based IDS can also be captured by the proposed model. Although there are discrepancies due to outliers, the gap between the predicted and experimental curves becomes smaller when n is increased from 20 to 60.

Prediction accuracy. As shown in Table III, the prediction error in terms of ADE is message-dependent for both IDSs. For the SOTA IDS, the average ADE is within 3.0% for the SOTA IDS, and it is within 5.7% for the NTP-based IDS. We also note that there is no explicit relationship between ADE and n for both IDSs.

VIII. Conclusions

In this paper, we proposed the cloaking attack and provided formal analyses of the attack for two clock skew-based IDSs, i.e., the SOTA IDS and the NTP-based IDS. We incorporated parameters of the attacker, the detector, and the hardware platform and derived attack success probabilities for both IDSs. We demonstrated the cloaking attack on hardware testbeds and validated the proposed models through extensive experiments using the data collected from the UW EcoCAR testbed. Our results illustrate the feasibility of developing formal analysis and models for other variants of CAN used in vehicles and applications such as computer-integrated manufacturing.

Acknowledgment

We would like to thank Drs. Sukarno Mertoguno and David Corman for discussions on this problem. This work was sup-

ported by NSF grants CNS-1446866 and CNS-1656981, ONR grants N00014-16-1-2710 and N00014-17-1-2946, and ARO grant W911NF-16-1-0485. Views and conclusions expressed are that of the authors and not be interpreted as that of the NSF, ONR or ARO.

REFERENCES

[1] S. U. Sagong, X. Ying, A. Clark, L. Bushnell, and R. Poovendran, "Cloaking the clock: emulating clock skew in controller area networks," in *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*. IEEE Press, 2018, pp. 32–42.

[2] S. Checkoway *et al.*, "Comprehensive experimental analyses of automotive attack surfaces," in *Proceedings of the 20th USENIX Conference on Security*, ser. SEC'11. Berkeley, CA, USA: USENIX Association, 2011, pp. 77–92.

[3] "Team of hackers take remote control of Tesla model S from 12 miles away," https://www.theguardian.com, accessed: 2018-07-10.

[4] "Hackers remotely kill a jeep on the highway - with me in it." https://www.wired.com, 2015, accessed: 2018-07-10.

[5] "CAN Specification Version 2.0," Robert Bosch GmbH, 1991.

[6] K. Koscher *et al.*, "Experimental security analysis of a modern automobile," in *Proceedings of the 2010 IEEE Symposium on Security and Privacy*, ser. SP '10, 2010, pp. 447–462.

[7] C. Miller and C. Valasek, "A survey of remote automotive attack surfaces," in *Black Hat USA*, 2014.

[8] C. W. Lin and A. Sangiovanni-Vincentelli, "Cyber-security for the controller area network (CAN) communication protocol," in *2012 International Conference on Cyber Security*, Dec 2012, pp. 1–7.

[9] D. K. Nilsson *et al.*, "Efficient in-vehicle delayed data authentication based on compound message authentication codes," in *2008 IEEE 68th Vehicular Technology Conference*, Sept 2008, pp. 1202–1206.

[10] A. Van Herrewege, D. Singelee, and I. Verbauwhede, "CANAuth - a simple, backward compatible broadcast authentication protocol for CAN bus," in *ECRYPT Workshop on Lightweight Cryptography*, 2011.

[11] P. S. Murvay and B. Groza, "Source identification using signal characteristics in controller area networks," in *IEEE Signal Processing Letters*, vol. 21, no. 4, April 2014, pp. 395–399.

[12] K.-T. Cho and K. G. Shin, "Fingerprinting electronic control units for vehicle intrusion detection," in *25th USENIX Security Symposium (USENIX Security 16)*, Austin, TX, 2016, pp. 911–927.

[13] ——, "Viden: Attacker identification on in-vehicle networks," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '17, 2017, pp. 1109–1123.

[14] W. Choi, H. J. Jo, S. Woo, J. Y. Chun, J. Park, and D. H. Lee, "Identifying ecus using inimitable characteristics of signals in controller area networks." *IEEE Trans. Vehicular Technology*, vol. 67, no. 6, pp. 4757–4770, 2018.

[15] C. Miller and C. Valasek, "Adventures in automotive networks and control units," in *DEF CON21*, 2013.

[16] ——, "Remote exploitation of an unaltered passenger vehicle," in *Black Hat USA*, 2015.

[17] I. Foster, A. Prudhomme, K. Koscher, and S. Savage, "Fast and vulnerable: A story of telematic failures," in *9th USENIX Workshop on Offensive Technologies (WOOT 15)*, Washington, D.C., 2015.

[18] X. Ying, G. Bernieri, M. Conti, and R. Poovendran, "TACAN: Transmitter authentication through covert channels in Controller Area Networks," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems (accepted)*, 2019.

[19] M.-J. Kang and J.-W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," in *PLOS ONE*, vol. 11, no. 6. Public Library of Science, June 2016, pp. 1–17.

[20] S. Dario, M. Mirco, and C. Michele, "Detecting attacks to internal vehicle networks through Hamming distance," in *2017 AEIT International Annual Conference*, Sep 2017, pp. 1–6.

[21] M. Marchetti and D. Stabili, "Anomaly detection of CAN bus messages through analysis of ID sequences," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, June 2017, pp. 1577–1583.

[22] H. M. Song *et al.*, "Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network," in *2016 International Conference on Information Networking (ICOIN)*, Jan 2016.

[23] M. R. Moore *et al.*, "Modeling inter-signal arrival times for accurate detection of CAN bus signal injection attacks: A data-driven approach to in-vehicle intrusion detection," in *Proc. of the 12th Annual Conference on Cyber and Information Security Research*, ser. CISRC '17, 2017.

[24] A. Taylor, N. Japkowicz, and S. Leblanc, "Frequency-based anomaly detection for the automotive CAN bus," in *2015 World Congress on Industrial Control Systems Security (WCICSS)*, Dec 2015, pp. 45–49.

[25] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive CAN networks – practical examples and selected short-term countermeasures," in *Proceedings of the 27th International Conference on Computer Safety, Reliability, and Security*, ser. SAFECOMP '08, 2008, pp. 235–248.

[26] M. Müter and N. Asaj, "Entropy-based anomaly detection for in-vehicle networks," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, June 2011, pp. 1110–1115.

[27] M. L. Han, B. I. Kwak, and H. K. Kim, "Anomaly intrusion detection method for vehicular networks based on survival analysis," *Vehicular communications*, vol. 14, pp. 52–63, 2018.

[28] A. Ganesan, J. Rao, and K. Shin, "Exploiting consistency among heterogeneous sensors for vehicle anomaly detection," in *WCX 17: SAE World Congress Experience*. SAE International, March 2017.

[29] H. Li *et al.*, "Poster: Intrusion detection system for in-vehicle networks using sensor correlation and integration," in *Proc of the 2017 ACM SIGSAC Conference*, ser. CCS '17. ACM, 2017, pp. 2531–2533.

[30] M. Müter *et al.*, "A structured approach to anomaly detection for in-vehicle networks," in *2010 Sixth International Conference on Information Assurance and Security*, Aug 2010, pp. 92–98.

[31] K.-T. Cho *et al.*, "CPS approach to checking norm operation of a brake-by-wire system," in *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems (ICCPS'15)*, 2015, pp. 41–50.

[32] W. Choi, K. Joo, H. J. Jo, M. C. Park, and D. H. Lee, "Voltageids: Low-level communication characteristics for automotive intrusion detection system," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 2114–2129, 2018.

[33] H. Lee, S. H. Jeong, and H. K. Kim, "Otids: A novel intrusion detection system for in-vehicle network by using remote frame," in *2017 15th Annual Conference on Privacy, Security and Trust (PST)*. IEEE, 2017, pp. 57–5709.

[34] M. Kneib and C. Huth, "Scission: Signal characteristic-based sender identification and intrusion detection in automotive networks," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018, pp. 787–800.

[35] S. U. Sagong, X. Ying, R. Poovendran, and L. Bushnell, "Exploring attack surfaces of voltage-based intrusion detection systems in controller area networks," in *ESCAR Europe 2018*, 2018.

[36] ISO, *International Standard ISO 11898-1 Road Vehicles-Controller Area Network (CAN), Part 1 Data Link Layer and Physical Signaling*, 2015.

[37] Bosch, *CAN Specification Version 2.0*, 1991.

[38] S. B. Moon, P. Skelly, and D. Towsley, "Estimation and removal of clock skew from network delay measurements," in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, Mar 1999, pp. 227–234.

[39] D. L. Mills, "Network time protocol (version 3): Specification, Implementation and Analysis," RFC 1305, Tech. Rep., 1992.

[40] S. Zander and S. J. Murdoch, "An improved clock-skew measurement technique for revealing hidden services," in *Proceedings of the 17th Conference on Security Symposium*, 2008, pp. 211–225.

[41] S. Jana and S. K. Kasera, "On fast and accurate detection of unauthorized wireless access points using clock skews," in *Proceedings of the 14th ACM MobiCom*, 2008, pp. 104–115.

[42] T. Kohno, A. Broido, and K. C. Claffy, "Remote physical device fingerprinting," in *Proceedings of the 2005 IEEE Symposium on Security and Privacy*, ser. SP '05, 2005, pp. 211–225.

[43] J. H. Novak, S. K. Kasera, and N. Patwari, "Preventing wireless network configuration errors in patient monitoring using device fingerprints," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*. IEEE, 2013, pp. 1–6.

[44] M. Basseville *et al.*, *Detection of abrupt changes: Theory and application*. Prentice Hall Informations and system science series, 1993.

[45] D. C. Montgomery, *Introduction to statistical quality control*. John Wiley & Sons, 2007.

[46] G. C. Goodwin and R. L. Payne, *Dynamic system identification: experiment design and data analysis*. Academic Press New York, 1977.

[47] "University of Washington EcoCAR 3," http://uwecocar.com/, accessed: 2017-09-26.

[48] "Linux-CAN/SocketCAN user space applictions," https://github.com/linux-can/can-utils, accessed: 2018-07-10.

APPENDIX

A. Proof of Lemma 1

Proof. To compute the mean of $e[m]$, the normalized identification error in the first attack batch (Eq. (15)), we first consider the mean of $|O_{avg}[m]|$, the distribution of the absolute value of the average offset in the m-th batch. Under the assumption in Eq. (17), we have

$$\begin{aligned}\mathbb{E}(|O_{avg}[m]|) &= \frac{1}{N-1}\sum_{i=2}^{N}\mathbb{E}[(i-1)(|\mu+\Delta T-\mu[m-1]|) \\ &\qquad + (\eta_{m,i}-\eta_{m,1})] \\ &= \frac{N}{2}(|\mu+\Delta T-\mu[m-1]|). \end{aligned} \tag{27}$$

Now we compute the mean of the third term in Eq. (15). Since $\mathbb{E}[a_{m,N}] = \mathbb{E}[a_{m,1}] + (N-1)(\mu+\Delta T)$, we have

$$\begin{aligned}&\mathbb{E}[S[m-1](t[m-1]+T_{m,0}+a_{m,N}-a_{m,1})] \\ &\quad = S[m-1](t[m-1]+T_{m,0}+(N-1)(\mu+\Delta T). \end{aligned} \tag{28}$$

Combining Eq. (27) and (28) with Eq. (15) yields (18).

The variance of $e[m]$ from Eq. (15) is

$$\begin{aligned}\sigma_e^2 &= \text{Var}\left(\frac{1}{N-1}\sum_{i=2}^{N}(\eta_{m,i}-\eta_{m,1}) - S[m-1](\eta_{m,N}-\eta_{m,1})\right) \\ &= \frac{1}{2}\left(\frac{N-2S[m-1]}{N-1} + 2S^2[m-1] - 2S[m-1]\right)\sigma^2,\end{aligned}$$

which completes our proof. □

B. Proof of Theorem 1

Proof. Let $e_n[k]$ be the normalized identification error in the k-th batch when the attack begins in the m-th batch, where $k \geq m$. Let τ be the decreasing rate of $e_n[k]$. Then we have $e_n[k] \approx e_n[m] - \tau(k-m)$. For the upper control limit L^+, its maximum is reached at the l-th batch, where $l = \max\{k : e_n[k] \geq \kappa, k \geq m\} = \lceil (e_n[m]-\kappa)/\tau \rceil + (m-1)$. We then have

$$\begin{aligned}L^+_{\max} &= \sum_{k=m}^{l}(e_n[k]-\kappa) = (l-m+1)\cdot e_n[m] \\ &\quad - \frac{(l-m+1)(l-m)}{2}\tau - (l-m+1)\cdot\kappa.\end{aligned}$$

Simplifying the above equation yields

$$L^+_{\max} = \frac{(e_n[m]-\kappa)^2}{2\tau} + \frac{e_n[m]-\kappa}{2}. \tag{29}$$

In order for the attack to be undetected, the condition that $L^+_{\max} \leq \Gamma$ needs to be met, or equivalently,

$$(e_n[m]-\kappa)^2 + \tau(e_n[m]-\kappa) - 2\tau\cdot\Gamma \leq 0. \tag{30}$$

Since $e_n[m] \sim N\left(\frac{\mu_e-\mu_{\text{CUSUM}}}{\sigma_{\text{CUSUM}}}, \frac{\sigma_e^2}{\sigma^2_{\text{CUSUM}}}\right)$, the probability of $L^+_{\max} \leq \Gamma$ is $Pr(e_n[m] \leq \frac{-\tau+\sqrt{\tau^2+8\tau\Gamma}}{2} + \kappa)$.

Similarly, for the lower control limit L^-, the probability of $L^+_{\min} \leq \Gamma$ is $Pr(e_n[m] \geq -\frac{-\tau+\sqrt{\tau^2+8\tau\Gamma}}{2} - \kappa)$. Combining these results yields (19). □

C. Proof of Lemma 2

Proof. The law of total probability implies

$$\begin{aligned}&Pr(\alpha > n | L^+[k] = z^+, L^-[k] = z^-) \\ &= \int_{-\infty}^{\infty} Pr(\alpha > n | L^+[k] = z^+, L^-[k] = z^-, e_n[k] = r) f_k(r)\, dr,\end{aligned}$$

and we have

$$L^+[k+1] = \begin{cases} 0, & r < \kappa - z^+ \\ z^+ + r - \kappa, & r \geq \kappa - z^+ \end{cases},$$

$$L^-[k+1] = \begin{cases} z^- - r - \kappa, & r < z^- - \kappa \\ 0, & r \geq z^- - \kappa \end{cases}.$$

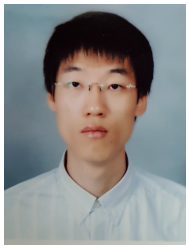
Now, first, suppose that $\kappa - z^+ < z^- - \kappa$. Then $z^+ + z^- > 2\kappa$, and hence by assumption $z^+ + z^- \geq 2\Gamma$. Thus either $z^+ \geq \Gamma$ or $z^- \geq \Gamma$, implying that $\tau = 0 < n$.

We can therefore write

$$\begin{aligned}&g_{n,k}(z^+, z^-) \\ &= \int_{-\infty}^{z^- - \kappa} Pr(\alpha > n | L^+[k] = z^+, L^-[k] = z^-, e_n[k] = r) f_k(r)\, dr \\ &+ \int_{z^- - \kappa}^{\kappa - z^+} Pr(\alpha > n | L^+[k] = z^+, L^-[k] = z^-, e_n[k] = r) f_k(r)\, dr \\ &+ \int_{\kappa - z^+}^{\infty} Pr(\alpha > n | L^+[k] = z^+, L^-[k] = z^-, e_n[k] = r) f_k(r)\, dr \\ &= \int_{z^- - \kappa - \Gamma}^{z^- - \kappa} g_{n,k+1}(0, z^- - r - \kappa) f_k(r)\, dr \\ &\quad + g_{n,k+1}(0,0) Pr(e_n[k] \in [z^- - \kappa, \kappa - z^+]) \\ &\quad + \int_{\kappa - z^+}^{\kappa - z^+ + \Gamma} g_{n,k+1}(z^+ + r - \kappa, 0) f_k(r)\, dr.\end{aligned}$$

This completes the proof. □

Xuhang Ying (S'15) is a Postdoctoral Research Associate at the Network Security Lab at University of Washington. He received his B.Eng. degree in Information Engineering from the Chinese University of Hong Kong, his M.S. and Ph.D. degrees in Electrical Engineering from the University of Washington in 2013, 2016, and 2018, respectively. His Ph.D. research focused on crowdsensing and resource allocation in shared spectrum. His research interests include Controller Area Networks (CAN) security and wireless security.



Sang Uk Sagong (S'11) is a Ph.D. candidate in the Electrical and Computer Engineering Department of the University of Washington. He received the B.S. degree and the M.S. degree in Electrical Engineering from Yonsei University, Korea in 2009 and 2011, respectively. His research interests include standard of the Controller Area Network (CAN) protocol and security of automobile.

Andrew Clark (M'15) is an Assistant Professor in the Department of Electrical and Computer Engineering at Worcester Polytechnic Institute. He received the B.S. degree in Electrical Engineering and the M.S. degree in Mathematics from the University of Michigan - Ann Arbor in 2007 and 2008, respectively. He received the Ph.D. degree from the Network Security Lab (NSL), Department of Electrical Engineering, at the University of Washington - Seattle in 2014. He is author or co-author of the IEEE/IFIP William C. Carter award-winning paper (2010), the WiOpt Best Paper (2012), and the WiOpt Student Best Paper (2014), and was a finalist for the IEEE CDC 2012 Best Student-Paper Award. He received the University of Washington Center for Information Assurance and Cybersecurity (CIAC) Distinguished Research Award (2012) and Distinguished Dissertation Award (2014). He holds a patent in privacy-preserving constant-time identification of RFID. His research interests include control and security of complex networks, submodular optimization, control-theoretic modeling of network security threats, and deception-based network defense mechanisms.

Linda Bushnell (F'17) is a Research Professor and the Director of the Networked Control Systems Lab in the Electrical and Computer Engineering Department at the University of Washington - Seattle. She received the B.S. degree and M.S. degree in Electrical Engineering from the University of Connecticut - Storrs in 1985 and 1987, respectively. She received the M.A. degree in Mathematics and the Ph.D. degree in Electrical Engineering from the University of California - Berkeley in 1989 and 1994, respectively. Her research interests include networked control systems, control of complex networks, and secure-control. She was elected a Fellow of the IEEE for her contributions to networked control systems. She is a recipient of the US Army Superior Civilian Service Award, NSF ADVANCE Fellowship, and IEEE Control Systems Society Distinguished Member Award. She is currently an Associate-Editor for *Automatica* and *IEEE Transactions on Control of Network Systems* and Series Editor for the Springer series *Advanced Textbooks in Control and Signal Processing*. She is currently Chair of the IEEE Control Systems Society (CSS) Women in Control Standing Committee, and Liaison to IEEE Women in Engineering (WIE). She is also the Treasurer of the American Automatic Control Council (AACC) and a Member of the International Federation of Automatic Control (IFAC) Technical Board.

Radha Poovendran (F'15) is a Professor and the Chair of the Electrical and Computer Engineering Department at the University of Washington (UW). He is the Director of the Network Security Lab (NSL) at the University of Washington. He is the Associate Director of Research of the UW Center for Excellence in Information Assurance Research and Education. He received the B.S. degree in Electrical Engineering and the M.S. degree in Electrical and Computer Engineering from the Indian Institute of Technology- Bombay and University of Michigan - Ann Arbor in 1988 and 1992, respectively. He received the Ph.D. degree in Electrical and Computer Engineering from the University of Maryland - College Park in 1999. His research interests are in the areas of wireless and sensor network security, control and security of cyber-physical systems, adversarial modeling, smart connected communities, control-security, games-security and information theoretic security in the context of wireless mobile networks. He is a Fellow of the IEEE for his contributions to security in cyber-physical systems. He is a recipient of the NSA LUCITE Rising Star Award (1999), National Science Foundation CAREER (2001), ARO YIP (2002), ONR YIP (2004), and PECASE (2005) for his research contributions to multi-user wireless security. He is also a recipient of the Outstanding Teaching Award and Outstanding Research Advisor Award from UW EE (2002), Graduate Mentor Award from Office of the Chancellor at University of California - San Diego (2006), and the University of Maryland ECE Distinguished Alumni Award (2016). He was co-author of award-winning papers including IEEE/IFIP William C. Carter Award Paper (2010) and WiOpt Best Paper Award (2012). He holds eight patents in wireless and aviation security.