Targeted Detection for Attacks on the MIL-STD-1553 Bus

MATTHEW ROGERS
KASPER RASMUSSEN
University of Oxford, Oxford, U.K.

Over the last decade we have observed a renewed focus on weapon systems security. Particularly the MIL-STD-1553 protocol, which was designed for military aircraft. In it, computers known as remote terminals (RTs) share information across a common serial data bus. Similarly to the well researched controller area network (CAN) bus, MIL-STD-1553 features no authentication, such that an attacker can manipulate the system by spoofing the bus controller (BC) and transmitting a single malicious message. These malicious messages are particularly bad in the MIL-STD-1553 context, where a single message can disable an RT, or engage a weapon system. To address these issues, this article proposes an intrusion detection system (IDS). While previous work utilizes the same techniques as used on the CAN bus, this leads to unnecessary complexity, inaccuracy, and poor efficiency. We take advantage of the protocol to detect an attacker spoofing the BC with 100% accuracy. In addition, we use standardized error flags to detect an attacker spoofing RT responses. The result of this work is an accurate and easy to implement detection system for all MIL-STD-1553 systems.

Manuscript received 28 March 2022; revised 27 July 2022 and 22 May 2023; accepted 15 October 2023. Date of publication 19 October 2023; date of current version 9 February 2024.

DOI. No. 10.1109/TAES.2023.3325423

Refereeing of this contribution was handled by G. Fasano.

Authors' addresses: Matthew Rogers and Kasper Rasmussen are with the University of Oxford, OX1 2JD Oxford, U.K. E-mail: (matthew.rogers2197@gmail.com; kasper.rasmussen@cs.ox.ac.uk). *(Corresponding author: Matthew Rogers.)*

I. INTRODUCTION

The MIL-STD-1553 protocol [1] and its derivatives control the aircraft and weapon systems used by many global powers [2]. This protocol suffers from many of the issues that plague more commonly researched serial data bus protocols like controller area network (CAN), including a complete lack of authentication on the bus. This means that any attacker capable of transmitting a message across the bus, be it via an implant, a supply chain attack, or a remote exploit, can effectively control the aircraft [3], [4], [5]. But the problem is actually worse with MIL-STD-1553 than other protocols. MIL-STD-1553 contains multiple mechanisms for disabling remote terminals (RTs), which are the computers exchanging data on the bus. In addition, MIL-STD-1553 utilizes a bus controller (BC) which dictates all traffic on the bus but has no security controls to uniquely identify it.

An attacker managing to transmit data on a military aircraft may seem like fiction, but the US government has done nothing but stress their fear of exactly this. In 2018, a Government Accountability Office report stressed the insecurity of US weapons systems [6], and the US Department of Defense issued the first update to the MIL-STD-1553 protocol since 2008, calling for improved cyber security for RTs [1]. Most military aircraft designed since 1973 have used MIL-STD-1553, and are not replaceable or retrofittable with modern resources. Thus, we propose an intrusion detection system (IDS) and intrusion prevention system (IPS) tailor designed to MIL-STD-1553 to provide a cost effective security solution. The IDS uses targeted voltage fingerprinting and built-in error flags to detect attacks.

Our solution has two main requirements. It must be fast enough and accurate enough to work as an IPS. It must be able to detect an attacker sending error or user initiated one-off messages.

In this article, when we say prevention system we mean a system which jams a message as it is being transmitted to cause a message to be dropped. Jamming a message mid-transmission requires a detection system that can make a decision without processing the entire message. This restricts the computational and algorithmic complexity of the IDS. It needs to be computationally simple enough to alert quickly and algorithmically simple enough to not need the full message context. Existing compatible prevention research uses a simple IDS that lacks the attack coverage we need to detect one-off attacks [7].

We highlight attack coverage as several existing research papers discuss the difficulty of differentiating attacks messages from user initiated and protocol messages [8], [9]. If these types of messages are ignored then an attacker can use them with impunity. Given that these types of messages in MIL-STD-1553 can trivially fire a weapon system or disable an aircraft, any IDS which does not include them is leaving their system at the mercy of the attacker.

The prevention and one-off attack coverage requirements result in two implied requirements. A prevention

system must be accurate or else it becomes a risk to the system. Based on existing detection systems' false positive rates we would expect an average of one erroneously prevented message every few minutes [5], [10], [11]. Preventing a weapon system from firing or a legitimate shutdown command from fixing the system is unacceptable. Our IDS must have higher accuracy. The second implied requirement is that any detection logic needs to be based on an individual message rather than an aggregated probabilistic approach. Aggregated approaches risk a single message being ignored and an impactful attack getting through.

To meet these requirements this article proposes a prevention-compatible IDS which has greater coverage and accuracy than existing work. We achieve these results on air worthiness certified MIL-STD-1553 computers by extracting the full security potential from the design of the protocol. Specifically, we utilize the architectural consequences of the bus controller, the hardware required by the stub connection to safely connect to such a high voltage bus, and the error detection built into standard transceivers for resiliency. Building around the protocol allows us to simplify our detection system such that it can be a better security system than existing work while simultaneously being faster and better suited to preventing attacks.

We summarize our contributions as follows.

1) An intrusion detection and prevention system for MIL-STD-1553 which can detect an attacker initiating a new message with 100% accuracy by using voltage fingerprinting on the BC.
2) A collision detection system which utilizes standard MIL-STD-1553 errors to detect any attacker interrupting messages on the bus or spoofing another active RT.

The rest of this article is organized as follows. The MIL-STD-1553 is thoroughly introduced in Section II, as well as an examination of related IDS research for MIL-STD-1553 and other vehicle protocols in Section III. Our system and adversary models which lay out how our IDS is connected to the system and what exactly an attacker can do is described Section IV. The full IDS and its three components are described in Section V. We examine the security model of our security analysis in Section VI before verifying that our system works on airworthiness certified MIL-STD-1553 computers in our experiment in Section VII. We discuss intrusion prevention systems and the assumptions of our system model in Sections VIII and IX. Finally, Section X concludes this article.

II. INTRODUCTION TO MIL-STD-1553

MIL-STD-1553 is a serial data bus protocol developed in the 1970s for military aircraft. It has since received major updates in 1975 and 1978 for increased standardization, and 2018 to put a focus on adding cyber security to devices connected to the bus. The core design philosophy was providing a common message format for avionics computers to exchange data, with support for a redundant bus in-case one of the avionics computers is destroyed. These avionics computers are referred to as RTs. Rather than having an arbitration process, where different RTs compete to exchange data, MIL-STD-1553 has a BC which sends or requests data from each terminal. In this section, we will describe the physical layer requirements of the bus, what MIL-STD-1553 messages looks like, the typical structure for how those messages are sent between the BC and RTs, and then some special characteristics of the protocol that are relevant for a security analysis.

A. Physical Layer Details

MIL-STD-1553 uses shielded twisted pair wiring for the bus with termination resistors and stub connections for each RT. All RTs and the BC are connected to all redundant buses, though only one bus should be active at a given time. Transmissions use the Manchester encoding, with a differential voltage of 18–27 V. The bus operates at a bit rate of 1 Mbit/s.

B. MIL-STD-1553 Messages

An MIL-STD-1553 message is composed of command, status, and data words. These words are depicted in Fig. 1. In brief, a command word tells the system what to do, a status word allows an RT to self-report any errors, and data words are the actual message payload for what the system is doing. Each word is 20 b with a 3-s sync waveform to let everyone on the bus know that a new word is starting, 2 B of actual information, and then an odd parity bit. We use message to mean the encapsulation of the call and response between the BC and RT, where each message ends as a new command word is received. A command word always come from the BC and contains the address of the RT it is addressing, whether that RT should expect to receive or transmit data, how much data are going to be sent (if any), and what subaddress (RT function) the BC is wanting to use. Status words are only ever sent by the RT and are a way for the addressed RT to make it clear it is ready to receive or transmit data; the status word is simply a collection of error flags.

Data words and RT addresses are defined by an interface control document (ICD) which tells every RT what each data word means, how much data to expect, and how that data changes based on what subaddress the BC sends in the command word. Every message has at least one command word, zero or one status words, and up to 31 data words. No status word will be sent if the BC chooses a broadcast address and some messages send no data.

An important note for MIL-STD-1553 messages is what makes a message valid. RTs determine message validity by checking for four things: the sync waveform at the beginning of every word, the Manchester encoding, that each word is 16 b long, and a valid odd parity bit. These mechanisms are useful for detecting transceiver failures and bit errors, especially for MIL-STD-1553 which has no defined bus arbitration process. It is always assumed that the BC initiates communication and decides who replies. The

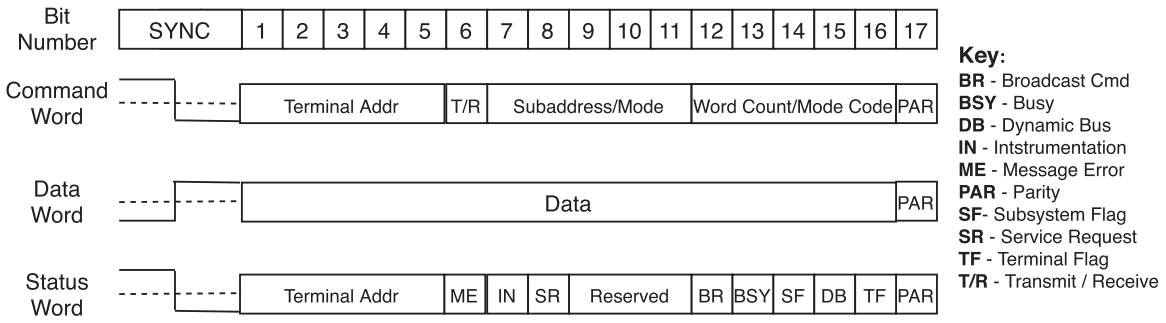

Fig. 1. Three "words" which make up MIL-STD-1553 messages. Command words start messages and indicate what the message will be about, status words are sent by RTs to indicate errors, and data words communicate the actual aircraft data.

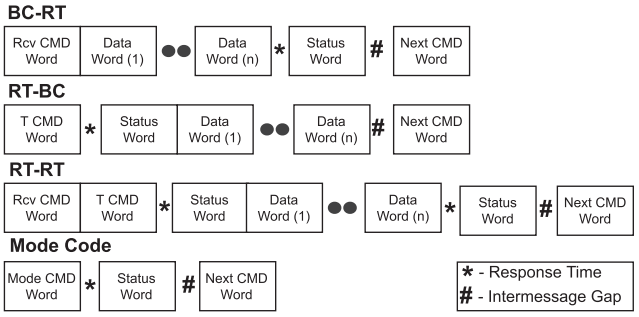

Fig. 2. Pattern of traffic for MIL-STD-1553 messages. Each message begins with a command word initiated by the BC, then varies by whether the BC is transmitting or receiving data. The dots represent up to n data words being transmitted.

consequence of this structure is that there is no collision avoidance.

C. Traffic Patterns and Standard Communication

Roughly speaking, communication follows four different message patterns in MIL-STD-1553: BC to RT, RT to BC, RT to RT, and mode code (BC to RT command). Fig. 2 depicts these patterns. Notably the BC always starts the communication with a command word, the difference between these options is simply whether the BC tells the RT to transmit or receive data. RT to RT is a special case where the BC tells one RT to receive data, then another to transmit data. In response to the opening command word, an RT has approximately 14 μs to respond with a status word before it times out and generates an error. Between messages there is a minimum of 4 μs.

D. Mode Codes

A mode code is a message, usually with no data, which orders an RT to do some action. Mode codes are identified by the subaddress field of the command word being set to 0 or 31. Four mode codes are required for every MIL-STD-1553 computers: 1) transmit [the last] status word [received], 2) reset remote terminal, 3) transmitter shutdown, and 4) override transmitter shutdown. The first is used for diagnosing errors, the rest are used to turn OFFor ON RTs that are in a failure state such that they do not monopolize the bus or risk the safety of the system by transmitting erroneously. These being implemented in every system means they are a helpful tool for an attacker. They are a single message, sometimes sent legitimately, which can disable any computer.

III. RELATED WORK

For related work this article pulls from existing work for MIL-STD-1553, as well as the related CAN bus research. The detection systems implemented on these serial data bus protocols follow three main patterns: 1) monitoring the physical layer signals (voltage), 2) the timing between messages, and 3) a contextual understanding of what data "makes sense" for a given scenario. Alternatively, researchers modify the transceiver to append a cryptographic authentication scheme [12], [13]. While strong cryptography for authentication is a good long term solution, the cost to modify each installed computer for decades of existing systems is prohibitve. This is particularly true for MIL-STD-1553, where systems are used for decades and regulations limit any hardware or software changes.

Voltage monitoring is typically used to fingerprint a computer. Researchers have shown that by analyzing the characteristics of a particular RT's signal, it is possible to identify which device is transmitting [11]. By identifying which device is transmitting, it becomes clear which device is performing the attack. This technique inherently detects malicious devices implanted onto the network, as they have a completely new fingerprint. Stan et al. [11] used RT fingerprinting, determining if a signal observed over the bus is the same as previously seen over the bus. Equivalent work in CAN requires constant retraining when outside of a lab environment to account for environmental factors, and can be prone to false positives [10], [14].

Message timing works by identifying anomalies in the sequence of frequency of messages. This technique relies on the assumption that when a message appears is predictable. MIL-STD-1553 theoretically uses a repeating set of messages, with a dedicated time frame for one-off messages. In practice, multiple implementations exist, though all have some predictable periodic messages. Stan et al. [11] and Onodueze et al. [15] used several machine learning techniques to predict the sequence of messages across the bus, with Stan having a separate model for periodic and

aperiodic messages. For the aperiodic model Stan et al. tried to predict what aperiodic message would come next based on aperiodic message history. By definition, the transmission of an aperiodic message does not influence the next aperiodic message sent. For example, a mode code which disables one computer, has no bearing on then firing a weapon system. That said, the periodic model presented in this article is effective, particularly on an MIL-STD-1553 bus implementing a major/minor frame timing scheme. A key problem with a message timing technique is that if an attacker corrupts an existing device then the technique is unable to detect the corrupted device transmitting periodic messages that it would normally transmit.

The final technique is analyzing the data words going over the system and determining when an attacker's message puts the overall vehicle into an anomalous state. For example, given a certain altitude, heading, and speed, is a specific maneuver or sensor output anomalous? This detection technique is capable of detecting corrupted RTs responding with malicious data words, though this approach has accuracy and computational power issues as it effectively simulates the system. Data word analysis is unable to determine when a one-off message is valid. This technique is not popular for academic researchers focusing on MIL-STD-1553 as the documentation for what each data word means is often confidential.

Our focus on detecting illegitimate mode codes means that existing message-timing-based approaches are not sufficient for detecting messages that could happen at any time. Our goal of a safe automated prevention system, and the associated speed requirements, means that data word analysis is impractical. In this article, we show that a simplified MIL-STD-1553-specific voltage monitoring approach can achieve greater accuracy than existing timing and data-based solutions. Most importantly, our solution covers the dangerous mode codes and weapon system messages which are not monitored by previous work.

IV. SYSTEM AND ADVERSARY MODEL

In this section, we define our MIL-STD-1553 bus and our corresponding assumptions about how our detection system and the adversary's device are connected to that bus. In addition, we define the capabilities of the adversary. Fig. 3 illustrates our system and adversary model, where the red RT is corrupted or an implant and our IDS is an addition to the system.

A. System Model

Our system model is an MIL-STD-1553 serial data bus utilizing a single BC, dual bus system. On the physical layer we assume each device is connected to the bus line using transformer coupling stub connections. The alternative, direct coupling, is banned from US Airforce and Army aircraft due to a max length of only 12 inches, no dc isolation, and a shorting of isolation resistors causing entire bus failure, among other reasons [16]. These groups also recommend not using dynamic bus control for sensitive systems, so

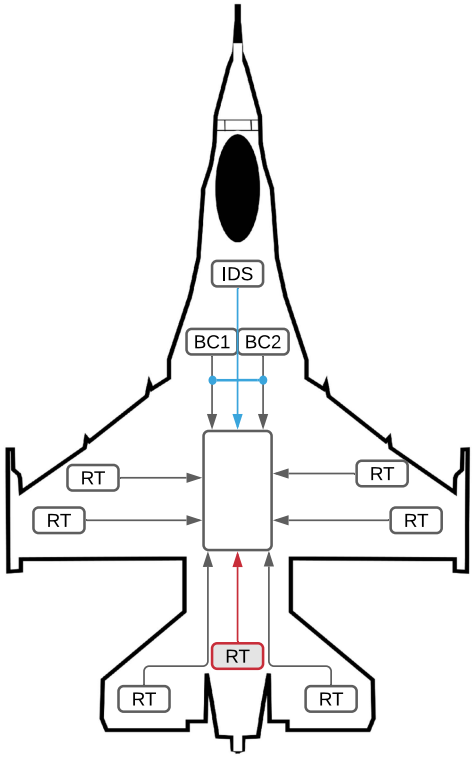

Fig. 3. System and adversary model depicting how our IDS is connected to the bus. The red outlined RT is taken over via a remote attacker. BC2 represents a backup BC common to these networks.

our system model will assume a singular, static, BC. The RTs vary from system to system, but examples include engine controllers, navigation, communication systems, and sensors. Each RT is capable of receiving and decoding commands from the BC. We assume the physical bus has not been damaged or spliced to include in line modifiers. While it is possible for advanced RTs to detect a plethora of system errors, and illegal messages, we do not assume our system contains any of these RTs. Our system's RTs follow the baseline standard, meaning they ignore any message with invalid Manchester encoding, more than 16 data bits in a word, an invalid sync, or an invalid odd parity. We assume these errors are rare on an attacker free system. We make no assumptions about how frequently or in what order messages are transmitted.

Our detection system is connected to both channels of the data bus, and can read MIL-STD-1553 messages. In addition, it is connected to the BC stub connection, enabling a Boolean indicator for when the BC is transmitting. The technical details of this indicator are described in Section V. We connect a high speed analog-to-digital converter to the BC stub and main bus line to monitor voltages across the stub and bus. We assume our detection system is secure from wireless threats and can indicate if uninstalled from a system.

B. Adversary Model

Our adversary is connected to our system as if they are another RT or BC on the system. We make no assumptions

about how the attacker got onto the system, though examples may include an implant, a remote compromise, or a supply chain attack. They can transmit and spoof any MIL-STD-1553 message. An attacker can also invalidate MIL-STD-1553 messages. Examples of this include adding extra status or data words, or intentionally colliding with a message. An attack is successful if they are capable of transmitting or cancelling a message/word on the MIL-STD-1553 bus without being detected.

An attacker targeting an MIL-STD-1553 system is likely seeking a military advantage, causing system errors at strategic times. This could manifest in navigation systems misleading a pilot, engaging and firing weapon systems around allies, or simply turning computers OFF to completely disable the system.

For the purpose of this article, we assume the BC is a trusted entity and cannot be compromised by the attacker. Our security system bolsters this assumption by making it more difficult to pivot from a foothold to the BC. We further examine this assumption in Section IX. Due to the high voltage and strict size and weight requirements on an aircraft we assume an attacker cannot override bits on the bus without producing an encoding error. The attacker has full knowledge of all of the devices on the system. Passive attackers are outside of the scope of this article.

V. INTRUSION DETECTION FOR MIL-STD-1553

In this section, we propose an MIL-STD-1553-specific intrusion detection system with three components: 1) a voltage monitor on the BC stub connection, 2) a voltage monitor on the main bus line, and 3) an error detection system which relies on standard MIL-STD-1553 transceiver error flags. We decide on voltage monitoring as our core technology as it is well suited to differentiating legitimate messages from spoofed messages. This is particularly important in MIL-STD-1553 as any mode code or weapon system message could result in catastrophic consequences. Timing-based and data-based detection systems do not have the ability to differentiate these one-off messages with the 100% accuracy required to potentially prevent attack impact.

Voltage-based detection systems in existing work have similar accuracy issues [14]; however, the bus controller in MIL-STD-1553 dramatically simplifies (and correspondingly increases the accuracy of) our voltage fingerprinting algorithm in two ways. The BC initiates all messages, including the previously mentioned one-off messages. It follows that the detection system only needs to determine if a message is from the BC or not to determine its validity in our system model. Fingerprinting one system is inherently easier. In addition, the BC is connected to the bus via a transformer stub connection. The result is a large voltage difference on the stub when the BC is transmitting or receiving data. The rest of this section details how the three components of the detection system function individually and as a full system.

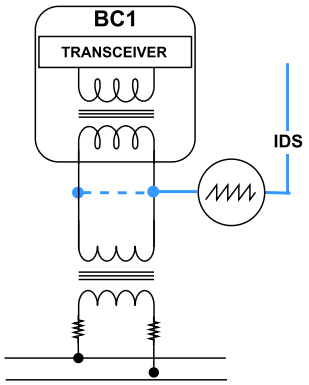

Fig. 4. Transformer coupling stub connection with our IDS connected between two transformers. The isolation transformer after the transceiver along with the coupling transformer and isolation resisters connected to the bus line ensure that the voltage at our IDS connection is significantly more when transmitting than receiving.

A. Fingerprinting the Bus Controller

Every message beginning with a command word started by the BC means that we do not need to fully fingerprint the system to detect spoofed command words (new messages). Full fingerprinting would allow us to know which device is spoofing a command word, but this capability comes with the cost of high complexity and retraining difficulties [14]. Risking false positives and creating additional attack windows runs counter to our goal of an intrusion prevention system and knowing which device is executing the attack is inconsequential for preventing attacks as they are transmitted. Detecting spoofed command words still leaves spoofed data words and status words, we argue in the next subsection that spoofing another RT while that RT is still online inevitably raises an error flag in our receiver, and thus, full fingerprinting is not necessary for detecting spoofed words. We propose only fingerprinting the BC. With this we maintain our ability to detect all messages initiated on the bus without overcomplicating the detection system.

A typical fingerprinting solution would monitor the main bus line, then train on labeled data to learn which RT is which. Instead, we can once again use the MIL-STD-1553 standard to our advantage. Each RT, including the BC, is connected to the bus via a transformer coupling stub connection, as shown in Fig. 4. Monitoring this connection reveals that a transmitting device has voltage transformed up, resulting in a higher voltage. A receiving device has voltage transformed down, resulting in a lower voltage. By connecting our IDS to the BC stub, as depicted in Fig. 4, we can determine when the BC is transmitting or receiving based on if the voltage is high or low. It follows that if our IDS monitors for a command word sync starting a message and sees that the voltage is low, then an attacker must be spoofing the BC. This solution requires no training and no high fidelity waveform analyzers. Our IDS only requires a connection to the stub connecting the BC and bus line and the ability to read the magnitude of the voltage. The proximity of the attacker to the BC on the main bus line has

no consequence on this technique due to the voltage drop-off being behind the transformer and the high differential voltage of MIL-STD-1553 (up to 27 V) keeping the impact from any errors or electromagnetic interference low.

B. Monitoring the Main Bus Line

Our IDS must account for an attacker intentionally circumventing our BC fingerprinting by operating the bus at a higher voltage than normal. Transmitting the attacker message at a higher voltage than normal makes it appear, from the perspective of our BC stub connection, as if the BC is transmitting the message. Our IDS then fails to detect this attack. To address this technique, we must also monitor the voltage of the main bus line. The principal of the detection strategy is that our IDS is exclusively fingerprinting the BC. This means that any change in voltage on the main bus line from one command word to the next is indicative of either the BC malfunctioning or malicious interference. If this solution is too restrictive it could risk false positives, however, the magnitude of the difference in differential voltage between transmitting and receiving is approximately 4 V, 8 V peaks to peak. By permitting any fluctuations up to 4 V peak to peak, we are safe from normal voltage fluctuations. The differential nature of reading voltage from an MIL-STD-1553 line helps ensure our system is not prone to environmental errors. Our IDS works without training, as we only need to alert on deviations from a single source. Any attempts to circumvent this technique by slowly raising the voltage over subsequent command word messages will generate alerts as the non-BC command words are transmitted. Raising the voltage significantly results in voltage deviation alerts, leaving the attacker with no options for transmitting a command word undetected.

C. Collisions and Spoofed Status and Data Words

The focus of our IDS thus far is on detecting command words, which occur whenever someone initiates a message. This section focuses on the rest of the attack space which is summarized by an attacker spoofing an RT's response or intentionally colliding with a word to invalidate it. Collisions might occur if an attacker creates a duplicate RT address or transmits at the same time. The collided with message will almost certainly be undecodable nonsense with Manchester encoding errors or sync waveform errors.

For spoofing there is nuance between spoofing status and data words. Spoofing data words is the case more likely to cause a collision as data are always sent immediately after an RT or BC has started transmitting a different word. To avoid a collision an attacker must send extra data words on the end. These extra words may be ignored or may be interpreted in a way that results in an exploit.

Spoofing status words has three cases: an attacker can transmit before, during, or after an RT responds. Transmitting before the RT responds does not result in a collision [4], as the legitimate RT thinks the spoofed message is a command word because the sync waveforms are the same and a status word at that time would make no sense. The attacker is using the same address as the command word it is responding to, otherwise a BC error occurs from a mismatched command and status word address. At this point two things can happen. Either the legitimate RT sees an illegal command word and does not transmit further or the legitimate RT transmits a status word response to the attacker's message. Our system model assumes illegal command functionality is not enabled, however, we believe it is a fair assumption that if illegal functionality is enabled then any security system can trivially observe an illegal message on the bus. If the legitimate RT responds as a status word, then our security system now thinks that a command word was sent, a status word responded, and now views the legitimate RTs status word as an illegal command word. This is detected by our BC fingerprinting security mechanism. We do not consider alerting on the legitimate RT a false positive considering that it is clear that one of the two messages sent right after each other is an attack. A defender could differentiate these cases by monitoring for fast status word responses. The attacker transmitting concurrently triggers a collision. The attacker transmitting after the RT is sending a command word as far as our security system is concerned, meaning our BC fingerprinting detection can detect the attack.

The above cases can be summarized as collisions, extra command words, and extra data words. Our BC fingerprinting mechanism handles any command words while the collisions and extra data words are handled by error flags built into standard MIL-STD-1553 transceivers, specifically invalid word, word count, and response timeout errors. Through these flags we can detect any collision, as long as that collision does not nullify a message such that no encoding error is observed. We believe this is not plausible with commercial off-the-shelf MIL-STD-1553 transceivers. We describe the relevant errors as follows.

Invalid Word Errors: indicate a sync field error, Manchester encoding error, parity error, and/or bit count error. This is useful for indicating collisions.

Word Count Errors: are triggered by an attacker transmitting a number of data words inequal to the value set in the command word. They might wish to do this to exploit a parsing error and cause some malicious effect but this error flag makes it simple to detect.

Response Timeout Errors: are registered whenever an RT does not respond or responds after a configurable amount of time (between 18.5 and 128 μm). This is useful for detecting an attacker holding the bus high such that no messages are processed.

In aggregate, invalid word errors ensure we detect a collision, word count errors detect extra data sent on the bus, and response timeout detects a denial of service from the bus being held high. Now an attacker needs to takeover a device to transmit as it and remain undetected. However, the attacker's target is not always vulnerable to supply chain or remote attacks. The attacker may need to pivot. We describe how our detection system restricts pivoting in our security analysis.

D. Detection Summary

Our IDS has three components. The BC fingerprinting component connects to the stub connection and monitors for command word observed at a low voltage to indicate an attacker spoofing the BC. The main bus line component monitor for voltage jumps from command word to command word to ensure an attacker is not bypassing the BC fingerprinting component. Finally, built-in MIL-STD-1553 error flags detect spoofed status and data words, extra words, and collisions. Taken together our adversary cannot accomplish their goal of transmitting or cancelling a message/word without being detected.

VI. SECURITY ANALYSIS

In this security analysis, we examine if an attacker can still achieve their goals while being fully aware of our intrusion detection system and how it works. The attacker's goal is injecting a command, status, or data word onto the bus, pivoting from one RT to another, or interrupting messages on the bus without being detected. Let us begin with them trying to initiate a message onto the bus.

Initiating a message requires the attacker to transmit a command word over the bus. Our detection system recognizes the initial command word sync waveform within the first 3 μm of the message. The attacker's message when received by our BC stub connection is at less than 10 differential volts, and so is alerted on as a non-BC initiated command word and detected. To get around this the attacker can transmit at a higher voltage across the main bus, such that the transformed voltage going into the BC is greater than 10 differential volts. However, the BC operates within a typical voltage window of approximately 4 V, making the substantial leap of the main bus line immediately anomalous in comparison to the BC. The nature of differential signalling makes this is a reliable metric, as any interference that would significantly alter the voltage of the bus will have little impact on the differential voltage. This leaves the attacker unable to transmit a command word undetected without taking over the BC, which is outside our system model.

If the attacker cannot initiate messages via a command word, then they can still affect the system by transmitting status or data words in response to the BC's command words. After a command word the the attacker responds before, during, or after the legitimate RT's response. As described in Section V, the attacker generates extra command words (which are detected by the previously described detection system), generates a collision, or generates extra data words. The attacker cannot disable the legitimate RT to avoid collisions as the built-in mechanism for doing this, mode codes, is detected by our command word spoofing detection. A collision raises an invalid word error due to: the sync or Manchester encoding being mangled, a parity error, or extra bits. Extra data words result in a word count error. These errors flags are generated on our receiver, and thus, the attacker has no way of preventing them.

The combination of these two security mechanisms, command word spoofing detection and collision and extra data detection, ensures that the attacker must take over the BC to transmit command words and must take over a specific RT to transmit status or data words as that RT. We do not detect malicious status or data words transmitted by a corrupted RT as long as they transmit in that corrupted RT's time slot. The only remaining option for an attacker not in this position is pivoting from one system to another. Pivoting to the BC requires the attacker's RT to be addressed with an RT to BC message and pivoting from one RT to another requires an RT to RT message, where the attacker is the transmitting RT. This provides a limitation on the attacker, as they must corrupt an RT that speaks to their intended pivot target. Furthermore, alerting on invalid word counts limits the attacker to a small payload for their any data exploit.

In summary, our BC detection system flags on any spoofed command words and spoofed status and data words are detected through standardized MIL-STD-1553 error flags. With this, we achieve the goals outlined at the start of the article: a detection system which rapidly alerts on attacks and can differentiate between legitimate rare messages, and attacker generated ones. Having full attack coverage for command words is a significant improvement on existing work, as it ensures an attacker cannot disable an RT or spoof any message from the BC without being detected. In addition, removing the ability to disable an RT undetected allows us to safely rely on standardized error flags to detect spoofing attacks.

VII. EXPERIMENTAL RESULTS

In this section, we describe our testbench, experiments, and results. These demonstrate the practical security benefits of this research and how they can be rapidly integrated into an MIL-STD-1553 system.

A. Testbench

Our test bed consists of two MIL-STD-1553 devices connected via transformer coupled stubs to the same bus. The first is an ALTA computer, effectively an FPGA with an MIL-STD-1553 transceiver. The second is a DDC MIL-STD-1553 chapter-10-compliant computer which is running our analysis software. We use each device's API to send MIL-STD-1553 messages across the bus, collect them with our detection system, and analyze the data using Python. For monitoring voltages, we use a single oscilloscope to maintain consistency in the resolution of measurements.

B. Experiment

We can test the efficacy of the detection system by having multiple MIL-STD-1553 devices transmit command words. If our IDS only alerts when a non-BC is transmitting, then we can confirm our ability to detect when the BC is transmitting. For this experiment, we use the two computers in our test bench, using different manufacturers to ensure our detection mechanism works on multiple MIL-STD-1553-compliant devices. First, we have the DDC computer take

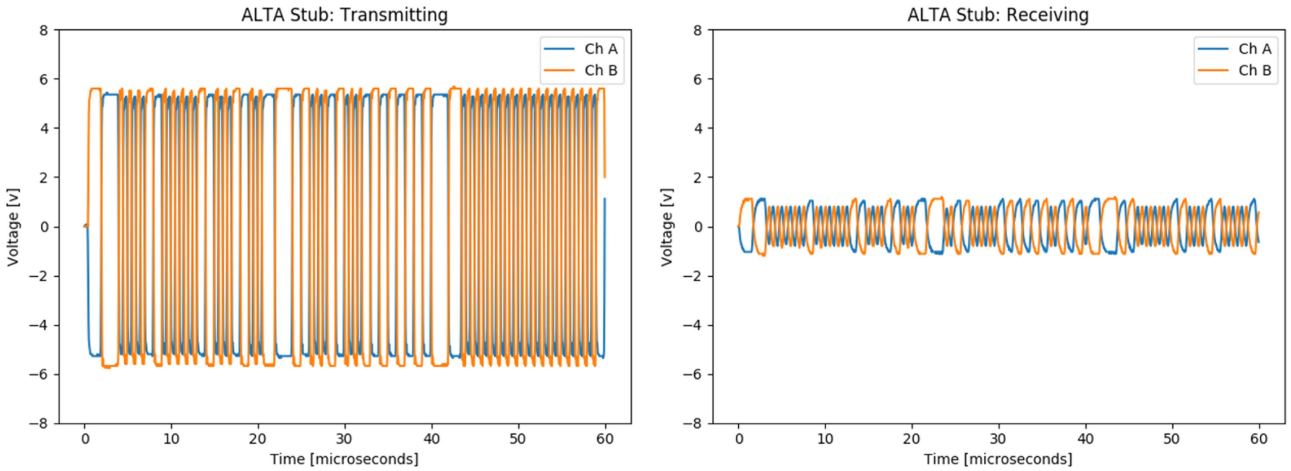

Fig. 5. Results of BC transmission detection experiment. These graphs show the voltage of each channel, depicting a clear divide in the magnitude of the peak to peak voltage observed by a transmitting device versus a receiving device from the perspective of the BC stub.

the role of the BC, connecting an oscilloscope to its stub connection, then having it transmit a set of test messages. The ALTA computer then replays those test messages as our RT to determine if the observed voltage was substantially less than observed when the DDC computer transmitted. We then swap our setup such that the ALTA computer is the BC, and the DDC computer is replaying messages as an RT. This experiment is successful if we can distinguish between the BC and other devices on the MIL-STD-1553 bus.

Our next experiment is intentionally introducing collisions and protocol errors onto the bus to determine what errors occur. We do this by having an RT attempt to transmit after our other computer starts transmitting a message. In addition, we add data words to the end of messages to simulate an attacker tagging onto an existing message to deliver an exploit. The experiment is a success if we can reliably detect collisions and extra words with standardized error flags.

C. Results

For our BC detection experiment we found we can successfully differentiate between when the BC is transmitting and receiving data. Fig. 5 demonstrates the results of our experiment with the ALTA device acting as the BC. We can see that when transmitting data our stub connection registered greater than 10 differential volts, while it registered less than 4 differential volts when receiving data. We observed the same results when the DDC device acted as the BC, and the ALTA device transmitted data. This gives us confidence in our detection mechanism's ability to have high accuracy with only the magnitude of the differential voltage, and a way of monitoring for a command word sync waveform.

Our collision test is colliding with a message as it is being sent. Our IDS transceiver registered an invalid word error in response every time we had two devices transmit at once. Notably, whenever a collision occurred our standard transceiver logged the error on the next message, dropping the message that was transmitted over. This complicates incident response as it is difficult to know what message the attacker decided to interrupt without using nonstandard collection hardware. It also means an attacker continuously interrupting the bus would ensure that our transceiver never processes an error frame up to the application layer IDS software. However, long periods of silence on an otherwise regular bus would make this attack trivial to detect. For messages with data words appended to the end, our transceiver raised word count errors. This ensures an attacker cannot arbitrarily force an RT or BC to read extra data without being detected.

VIII. Prevention for Illegitimate Command Words

An intrusion prevention system takes a detection system and adds the ability to cancel messages on the bus. In the case of serial data bus networks, this usually involves mangling the offending signal, causing each RT to drop the targeted message [7]. In order for an intrusion prevention system to be effective it must be accurate, and fast. An IPS must be accurate because preventing legitimate messages could cause safety issues, particularly on a moving aircraft, making false positives unacceptable. An IPS must be fast because it must process an attack, detect the attack, and then mangle the message. All of this must be done before the message is completed, or else the system's computers will process the attacking message. The fastest message on an MIL-STD-1553, a mode code with no data, takes 20 μm.

This is all to say our BC fingerprinting technique is perfect for an intrusion prevention system. Our evaluation gives us confidence in the accuracy of this technique. In terms of speed, if a command word sync is seen, and the observed voltage does not indicate the BC is transmitting, then our intrusion prevention system is capable of detecting an attack within the first 3 μm of a transmission. Our IPS is left with 17 μm to process this alert and create a collision which mangles the victim signal.

We are unable to provide prevention protections to collision-based attacks (those detected using standard transceiver errors), as we would simply further mangle the message. This allows an attacker to perform denial-of-service-style attacks, but they cannot do so undetected. We also cannot prevent extra data words. More targeted or nuanced attacks require transmitting a command word and we can prevent these attacks.

Another option for intrusion prevention in MIL-STD-1553 is using the shutdown mode code ourselves. This would disable an RT until the entire system is reset, or we use another mode code to restart it. While this could prevent future denial of service attacks, it requires a number of capabilities and assumptions to work. First and foremost, our IDS would need a way to determine what device sent the attack, which typically requires full scale voltage fingerprinting. Otherwise, an attacker could spoof the address of one RT, send an obvious attack, and then our IPS would shutdown a completely legitimate RT. In addition, this approach assumes the attacker reads the bus and perfectly follows the protocol. We do not believe these are reasonable assumptions.

IX. DISCUSSION

A. Denial of Service Attacks

Denial of service attacks prevent all communication on the bus, either by holding the line high, or causing collisions with individual messages. This is exactly how our intrusion prevention system works. Because our IPS uses collisions to prevent messages on the bus, it is incapable of preventing collisions on the bus. These collisions are still detected, but colliding with a collision is pointless as the message is already dropped. The result is that a denial of service attack can hold the bus hostage.

To that point, it is important to understand what occurs when bus communication stops. If no bus communication is happening and no mechanical backups occur, then no changes to the system can be made. But the consequences of that are not defined in the standard. Whether the vehicle turns OFF, an emergency protocol begins, or the last state continues until bus operation resumes is implementation specific.

B. Corrupted Bus Controller

Our system assumes that the BC is not taken over by the attacker. In terms of realism, the BC is unlikely to host any typical remote communication functionality and so would not be the initial foothold. This means our pivoting detection is useful against remote attacks. As for supply chain attacks, one would hope the computer that is in control of a very expensive aircraft would be the most hardened part of that aircraft, but it is difficult to generalize. That said, no existing MIL-STD-1553 research secures against a corrupted BC as they essentially control all of the data going over the bus. Timing-based systems only work if the corrupted BC breaks from standard traffic patterns. Fingerprinting is ineffective since they are transmitting as that device. The BC dictates data to RTs which limits how effective any physical modeling-based detection can work. Cryptography is ineffective since the corrupted device presumably has access to any secure keys to transmit data.

We believe the best way to handle this is to have our detection system on the BC. Our system is far simpler when implemented on the BC directly as the BC knows when it is or is not transmitting. In addition, end point monitoring software on the BC itself would likely be effective at detecting any attackers on the system. Similarly to how Internet-of-Things research proposes monitoring CPU and memory usage to identify infected devices [17]. Avionics computers act regularly enough that it should be obvious when new activity is occurring on the device.

X. CONCLUSION

In this article, we present a novel IDS tailor designed for MIL-STD-1553. Our detection system detects any illegitimate messages started as if they were from the BC; this is regardless of if the attacker is sending standard traffic, engaging weapon systems, or transmitting legitimate error and diagnostic messages at inappropriate times. We detect these attacks with 100% accuracy through a simple BC fingerprinting solution which is resistant to attacker activity or environmental factors. In addition, we show that standardized MIL-STD-1553 receiver errors can be used to detect an attacker attempting to spoof RT responses or collide with traffic as it goes across the bus. The simplicity behind these approaches allows any MIL-STD-1553 system maintainer to implement our IDS without complicated training or advanced fingerprinting capabilities. By designing our IDS with the MIL-STD-1553 protocol in mind we created a more impactful detection system which can operate quickly enough and accurately enough to power an intrusion prevention system. This is desperately needed to protect the millions to billions of dollars represented by even a single MIL-STD-1553 system, much less the decades of aircraft and infrastructure relying on the protocol.

REFERENCES

[1] *Digital Time Division Command/Response Multiplex Data Bus*, US Department of Defense Department of Defense Interface Standard MIL-STD-1553 C, US Department of Defense, Arlington, VA, Standard, Feb. 2018.

[2] D. Young, "MIL-STD-1553's longevity is well deserved," 2010. [Online]. Available: https://militaryembedded.com/avionics/computers/mil-std-1553s-longevity-well-deserved

[3] D. De Santo, C. Malavenda, S. Romano, and C. Vecchio, "Exploiting the MIL-STD-1553 avionic data bus with an active cyber device," *Comput. Secur.*, vol. 100, 2021, Art. no. 102097.

[4] K. Lounis, Z. Mansour, M. Wrana, M. A. Elsayed, S. H. H. Ding, and M. Zulkernine, "A review and analysis of attack vectors on MIL-STD-1553 communication bus," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 6, pp. 5586–5606, Dec. 2022.

[5] S. J. Genereux, A. K. Lai, C. O. Fowles, V. R. Roberge, G. P. Vigeant, and J. R. Paquet, "MAIDENS: MIL-STD-1533: Anomaly-based intrusion detection system using time-based histogram comparison," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 1, pp. 276–284, Feb. 2020.

[6] C. Chaplain, "Weapon systems cybersecurity: DOD just beginning to grapple with scale of vulnerabilities," US Government Accountability Office, Washington, DC, USA, GAO-19-128, Oct. 2018.
[7] H. Giannopoulos, A. M. Wyglinski, and J. Chapman, "Securing vehicular controller area networks: An approach to active bus-level countermeasures," *IEEE Veh. Technol. Mag.*, vol. 12, no. 4, pp. 60–68, Dec. 2017.
[8] K. Cho and K. Shin, "Error handling of in-vehicle networks makes them vulnerable," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 1044–1055.
[9] K. Cho and K. Shin, "Fingerprinting electronic control units for vehicle intrusion detection," in *Proc. 25th USENIX Conf. Secur. Symp.*, 2016, pp. 911–927.
[10] 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 Trans. Inf. Forensics Secur.*, vol. 13, no. 8, pp. 2114–2129, Aug. 2018.
[11] O. Stan, A. Cohen, Y. Elovici, and A. Shabtai, "Intrusion detection system for the MIL-STD-1553 communication bus," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 4, pp. 3010–3027, Aug. 2020.
[12] P. Murvay and B. Groza, "Security shortcomings and countermeasures for the SAE J1939 commercial vehicle bus protocol," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4325–4339, May 2018.
[13] O. Pfeiffer, *Implementing Scalable CAN Security with CANcrypt: Authentication and Encryption for CANopen, J1939 and Other Controller Area Network Or CAN FD Protocols*. San Jose, CA, USA: Embedded Systems Academy Incorporated, 2017.
[14] M. Kneib and C. Huth, "Scission: Signal characteristic-based sender identification and intrusion detection in automotive networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2018, pp. 787–800.
[15] F. Onodueze and D. Josyula, "Anomaly detection on MIL-STD-1553 dataset using machine learning algorithms," in *Proc. IEEE 19th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, 2020, pp. 592–598.
[16] "Electrical and layout considerations for 1553 terminal design," Data Device Corporation, Bohemia, NY, USA, AN/B-27, 2012. [Online]. Available: https://www.milstd1553.com/wp-content/uploads/2012/12/MIL-STD-1553B.pdf
[17] S. Tedeschi, C. Emmanouilidis, J. Mehnen, and R. Roy, "A design approach to IoT endpoint security for production machinery monitoring," *Sensors*, vol. 19, no. 10, 2019, Art. no. 2355.

Matthew Rogers received the D.Phil. degree in cybersecurity from the University of Oxford, Oxford, U.K., in 2023, on a Rhodes Scholarship.

He is focused on serial vehicle and infrastructure cyber security. His research interests include CAN, MIL-STD-1553, and commercial avionics systems.

Kasper Rasmussen received the Ph.D. degree in computer science from ETH Zürich, Zürich, Switzerland.

He is joined the University of Oxford, Oxford, U.K., in 2013, where he is currently an Associate Professor with the Computer Science Department.

Dr. Rasmussen was the recipient of the University Research Fellowship from the Royal Society in London, 2015.