Anomaly Detection for the MIL-STD-1553B Multiplex Data Bus Using an LSTM Autoencoder

Alec Harlow, Brian Lachine and Vincent Roberge
Royal Military College of Canada, Kingston, Canada

alec.harlow@forces.gc.ca
brian.lachine@rmc.ca
vincent.roberge@rmc.ca

Abstract: Due to the modernization of commercial and military aircraft, real-time systems and their connectivity to ground based networks, including the Internet, that were thought to be "air-gapped", are becoming more susceptible to cyber-attack. Most real-time systems that communicate using the Military Standard 1553B Multiplex data bus (MIL-STD-1553B) protocol do not have the ability to detect cyber-attacks. These systems were originally developed with safety and redundancy in mind, not security. These two factors introduce attack vectors to MIL-STD-1553B communication buses and expose associated avionics systems to exploitation. Recent approaches to anomaly detection for the MIL-STD-1553B data bus have leveraged statistical analysis, Markov Chain modelling, remote terminal fingerprinting and signature-based detection. However, their comparative effectiveness is unknown. Regarding the statistical analysis technique, the lack of accuracy and precision in detecting the start and stop time of anomalous events are not ideal for conducting investigations due to the sheer volume of messages still required to be manually analysed. Deep learning techniques offer an effective means of anomaly detection and applying these techniques to the MIL-STD-1553B data bus could provide more accurate and precise detection times when anomalies or attacks are present, when compared to known statistical analysis, leading to more efficient forensic investigations of anomalous events. The aim of this research is to improve the time-related performance metrics when detecting attacks on the MIL-STD-1553B data bus traffic using a Long Short-Term Memory (LSTM) autoencoder. In order to accomplish this aim, an LSTM autoencoder detector was developed and tested on two separate datasets from different MIL-STD-1553B network architectures, totalling 15 threat instances over 5 scenarios. The detector was then compared to the MIL-STD-1553B Anomaly-Based Intrusion Detection System (MAIDENS) detector, a statistical-based intrusion detection system. The LSTM autoencoder detected every threat instance with no false positive or false negative results and significantly improved the time-related performance metrics when compared to the MAIDENS detector. The results demonstrated this deep learning technique as an effective method for identifying anomalies on a MIL-STD-1553B data bus and significantly reducing the overall number of frames to be analysed during the investigation of identified anomalies.

Keywords: MIL-STD-1553B, Anomaly Detection, Deep Learning, LSTM Autoencoder, Aviation Cybersecurity

1. Introduction

Current MIL-STD-1553B anomaly detection techniques are able to detect a number of attack techniques that include, but are not limited to: Denial of Service (DoS) attacks, Remote Terminal (RT) spoofing attacks, Bus Controller (BC) spoofing attacks and attacks that manipulate RT messages (Stan et al, 2017). The MIL-STD-1553B Anomaly-Based Intrusion Detection System (MAIDENS), a statistics-based intrusion detection system purposed by Généreux et al (2020) is able to detect all of the aforementioned attacks. Other detectors such as a Markov chain model proposed by Stan et al (2017) and a fingerprinting method proposed Stan et al (2019) were able to detect a smaller subset of attack types. The issue faced with a detector like MAIDENS, even though it has an impressive zero false positive rate, is that the detection time accuracy and precision achieved when indicating the start and stop time of an attack is not ideal for conducting forensic analysis on a detected attack. Having a detector that could reduce the detection window and thereby the number of messages associated with potential attack traffic, would significantly improve analysis efforts post attack detection.

To address the need to improve the time-related performance metrics when detecting attacks on the MIL-STD-1553B Data Bus, this paper presents research into the detection of anomalies on the MIL-STD-1553B Data Bus using a Long Short-Term Memory (LSTM) autoencoder deep learning (DL) technique. It identifies the improvement in time-related performance metrics of an LSTM autoencoder detector in identifying attacks in MIL-STD-1553B bus traffic when compared to the MAIDENS detector purposed by Généreux et al (2020).

The statistics-based intrusion detection system, MAIDENS, can detect every attack event or threat occurrence that it has been tested against with perfect event classification accuracy and precision. However, if you consider the time-related performance metrics MAIDENS can only detect an anomaly event within a range of 10 ±8 seconds (Généreux et al, 2020). This equates to about 10,000 ±8000 messages of traffic, based on the data transfer rate from the MAIDENS dataset, and would take an expert a significant amount of time to analyse each message to find the actual start and stop time of an attack. This deficiency stems from the

inability to label each individual message through available MIL-STD-1553B recording method which relies on statistical metrics for event classification instead of the more traditional confusion matrix-based approach, that would identify each message as a true/false positive or negative. Therefore, being able to improve the time-related performance metrics, specifically the detection time accuracy and detection time precision of the start and stop time of an attack, would drastically decrease the number of messages needed to be analysed during an investigation. In this context, accuracy is defined as the closeness of agreement between a test result and the accepted reference value and precision is defined as the agreement between independent test results obtained under stipulated conditions of detection (ISO 5725-1, 1994).

In order to accomplish the aim, the MIL-STD-1553B Data Bus traffic dataset from the work conducted in MAIDENS was used and includes five threat scenarios and baseline traffic. The time related performance metrics of both MAIDENS and the LSTM autoencoder methods to detect these anomalies were then evaluated in terms of improved detection time accuracy and detection time precision.

2. Background

MIL-STD-1553B is a military standard bus communications protocol published in 1973 by the United States Department of Defense (DoD). The standard uses a multipoint topology of remote terminals (RTs) connected by a dual redundancy data bus as depicted in Figure 1. One terminal is designated as the bus controller (BC) and initiates and directs all communication on the bus. A bus monitor (BM) can be attached to the bus, but traditionally fills the role of a data historian, analogous to a “black box” on an aircraft.

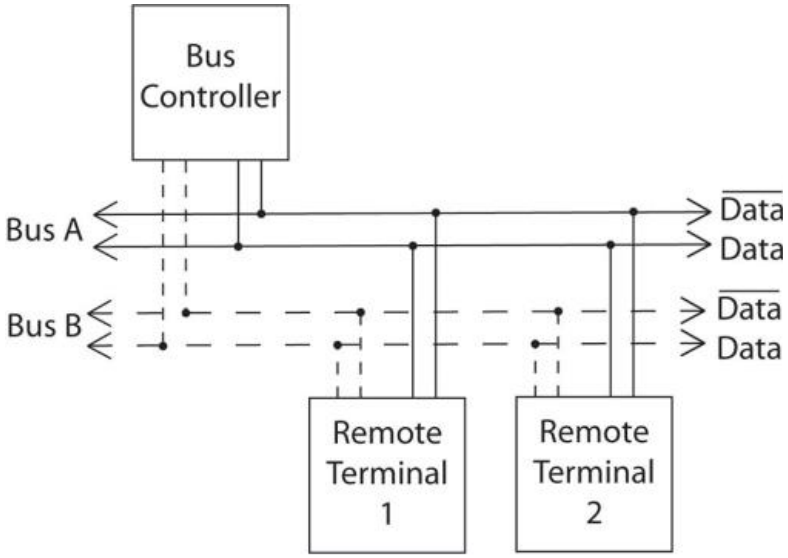

Figure 1: Example of a MIL-STD-1553B Bus Topology Consisting of a BC and Two RTs Connected by a Dual Redundancy Data Bus (Généreux et al, 2020)

There are a total of 32 addresses (0 – 31) on a MIL-STD-1553B bus network, one address (31) is reserved as a broadcast address and the remaining 31 addresses are for potential RTs that can be connected to the data bus. Each address has up to 32 sub-addresses or addressable data buffers that are used to read and write data to and from, with sub-addresses 0 and 31 being reserved for mode codes. Information is transferred through three 20-bit messages: a command word, a status word and a data word. The per-bit-breakdown of each word is shown in Figure 2. The protocol uses these three types of words for all communication on the bus and in order to initiate communication a command word must be sent by the BC (MIL-STD-1553 Designer's Guide, 1998). This ensures that all traffic on the bus is scheduled and reliable.

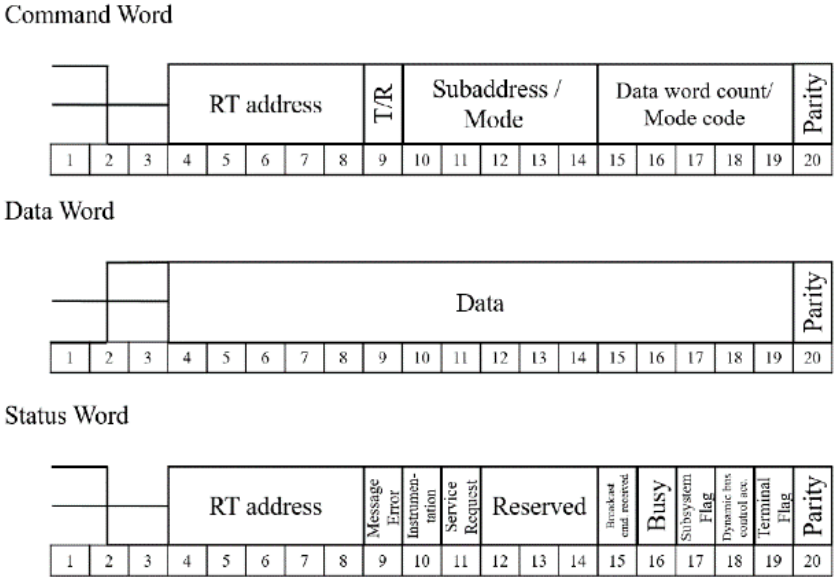

Figure 2: Per Bit Breakdown of Command, Data and Status Words for MIL-STD-1553B (Stan et al, 2019)

2.1 MIL-STD-1553B Vulnerabilities

The MIL-STD-1553B Protocol was designed with safety and reliability, not security in mind. Every communication exchange is preprogrammed and follows a cyclic schedule controlled by the BC. Manufacturers of RTs are expected to follow the guidelines defined by the MIL-STD-1553B protocol (MIL-STD-1553 Designer's Guide, 1998). However, attackers are not limited by these guidelines set by the design documentation and can use its flaws to achieve their desired effect.

Stan et al (2017) showcase two main attack categories: message manipulation and behaviour manipulation. Message manipulation refers to modification of legitimate words (command, data, or status) that are transmitted over the bus. Behaviour modification refers to altering how a component would normally operate. Using these two types of attacks methods, Stan et al (2017) identify 3 main types of threats to the MIL-STD-1553B communications protocol: Denial of Service (DoS), data leakage, and data integrity violation. These attacks can be carried out by two means. The first is by a rogue RT: a device that is not intended to be connected to the data bus and was not part of the original design. The second is by a compromised RT: a RT that is part of the original design, but has been maliciously modified by some means. In addition to the types of attacks described in Stan et al (2017), a study by Lounis et al (2022) reviews and analyses the attack vectors on the MIL-STD-1553B data bus as well. They identify 4 types of attacks that can occur: fabrication attacks, interception attacks, interruption attacks, and modification attacks. These attacks have some overlap with the types of attacks in Stan et al (2017) and contain attack vectors relating to RT components, not just the MIL-STD-1553B protocol. In this paper, the attack taxonomy from Stan et al (2017) will be followed.

2.2 Existing MIL-STD-1553B Detection Methods

The MAIDENS detector purposed by Généreux et al (2020) is a statistical anomaly-based intrusion detection system that uses a time-based, histogram comparison method. The time-based features of the MIL-STD-1553B bus are used to create a baseline histogram representation of known, intrusion-free data. By plotting the frequency of the values of a given feature, they are able to compare run-time data to the baseline data and if it is a certain percentage from the baseline, it is considered anomalous. Based on the results of Généreux et al (2020), the presented MAIDENS detector was able to identify the start and stop times in which a threat occurrence took place across five scenarios with no false positive or negative results. The five scenarios included four DoS type attacks: DoS using Command words, DoS using Status words, RT specific DoS and RT specific sub-address DOS. It also included one data manipulation attack, RT Hijacking, which is a form of data integrity violation and uses a compromised or rogue RT to take over or control a legitimate RT. The detection times ranged from within 0.38 seconds to 47.11 seconds of an attacks start and stop time. This equates to detecting an attack within 380 to 47,110 messages, assuming a bus averages one message every 0.001 seconds, which is not ideal when investigating an attack or threat instance given the potential for a significant number of bus frames to analyse that may not be related to the attack. Overall, the MADIENS detector is

effective at detecting attack threat occurrences, however the accuracy of the detection times could be improved upon to assist in the investigation of detected threat occurrences.

2.3 LSTM Autoencoders

An LSTM autoencoder uses LSTM layers to learn the compressed representations of a dataset (Provotar et al, 2019) and combines the signal reconstruction ability of an autoencoder with the ability of an LSTM to learn and model a time series dataset. The main differences of a LSTM autoencoder and a regular autoencoder is that the main blocks of the network architecture are LSTM cells as shown in Figure 3. The effectiveness of a LSTM in reconstructing the input signal or input values is indicated by its reconstruction error. The higher the reconstruction error value the more inaccurate the predicted values were compared to the input values, the opposite being true for a lower value. This is useful when determining whether data contains anomalies, as the more anomalies contained in the data, the more difficult it will be to recreate the input data resulting in a higher reconstruction error value for each recreated value. The reconstruction error can then be used as a method for setting a threshold value, where any recreated value that is higher than the threshold can be considered anomalous.

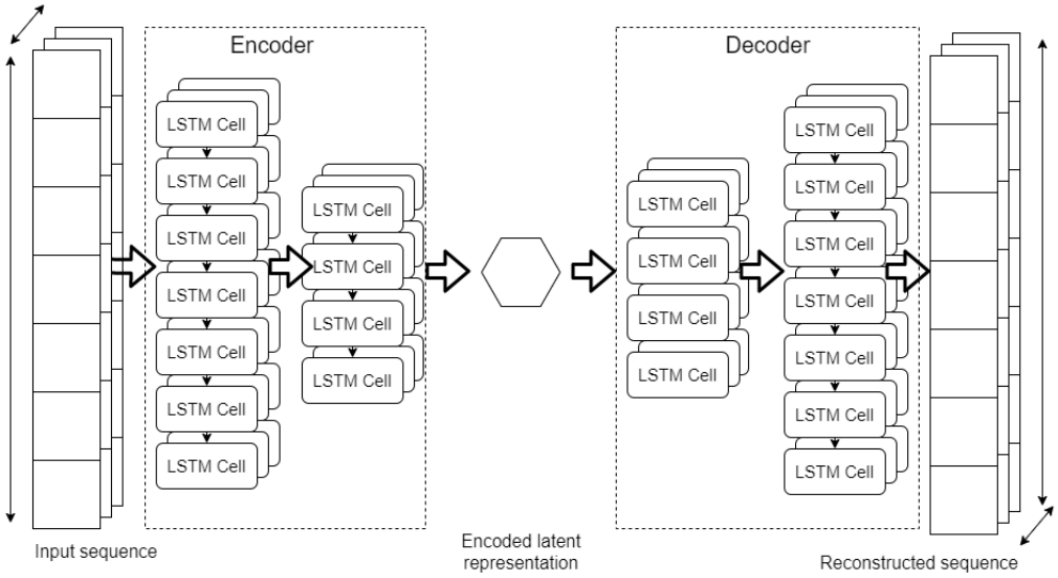

Figure 3: Architecture of a LSTM Autoencoder (Venskus et al, 2021)

2.4 Practical Applications of LSTM Networks

There are examples where LSTM networks either alone or in combination with other methods are used for anomaly detection. One such method was purposed by Taylor et al (2016), using a LSTM RNN to detect anomalies on the automotive Controller Area Network (CAN) bus. The proposed detector accepts raw CAN bus data words as an input and works by learning to predict the next data word originating from each sender on the bus. If there are bits in the next word that are not expected they are flagged as anomalies. This predictive type of modelling would be ideal for behaviour-based anomaly detection, as it is able "to detect data packet anomalies that are unusual only in the context of the rest of the sequence." With the added benefit that, "The LSTM based anomaly detector can be applied to many different vehicles without substantial modification." (Taylor et al, 2016).

As described in Provotar et al (2019), autoencoders that use deep encoders and deep decoders offer many advantages over the standard single layer encoders and single layer decoders. The depth can exponentially reduce the computational cost of representing some functions and also exponentially decrease the amount of training needed to learn some functions. As well, experimentally, deep autoencoders yield a better compression compared to linear autoencoders. In recent years, there has been additional research in the field of using LSTM Autoencoders for anomaly detection. Maleki et al (2021) apply a LSTM autoencoder to detect anomalies in CPU utilization, in a cloud computing environment. They concluded that an advantage of their applied method was the detection of both abrupt and gradually developing anomalies. Said Elsayed et al (2020) use a hybrid method of a LSTM Autoencoder combined with a OCSVM to detect anomalies in web traffic. Their method showed promising results, however due to the nature of web traffic, further real-world

testing was required to confirm the detectors viability. Mahmoud et al (2022) used an LSTM Autoencoder for intrusion detection in IoT systems, which showed impressive performance metrics, when compared to other Autoencoder models on the commonly used NSL-KDD dataset. Overall, the use of LSTM Autoencoder based anomaly detectors have demonstrated the ability to achieve positive anomaly detection results.

The success of the LSTM RNN model in the works of Taylor et al (2019) for anomaly detection on the CAN bus is a good indication that LSTMs will work well for anomaly detection on other bus like networks. Combined with the promising anomaly detection rates of the LSTM autoencoder presented in the works of Provotar et al (2019), Maleki et al (2021), and Mahmoud et al (2022), the MIL-STD-1553B data bus is a good test case for anomaly detection using a LSTM autoencoder DL method, leading the focus of this research to explore the effectiveness of the LSTM autoencoder DL method for anomaly detection on MIL-STD-1553B networks.

3. Methodology and Design

The following three phases were conducted in order to achieve the aim of this research:

1. Data Acquisition Pipeline:
    a. Generation and collection of MIL-STD-1553B baseline and anomalous traffic recordings.
    b. Extraction of features derived from MIL-STD-1553B message traffic.
2. LSTM Autoencoder Model Development:
    a. Creation of a feature extraction tool to ingest MIL-STD-1553B recordings and prepare a usable dataset for model creation.
    b. The creation and application of a LSTM Autoencoder detector to create a baseline model to evaluate subsequent traffic against for anomaly detection.
3. Validation of Detection and Time-Related Performance Metrics
    a. Validation of the detector based upon its ability to accurately and precisely detect anomalies in comparison to MAIDENS using the same datasets.

In order to perform detection on a set of data with a ML or DL technique it has to be in a viable format to be ingested by the detector. The set of data points can be in the form of a stream of data, such as a sequence of messages from a MIL-STD-1553B data bus. In order to collect and create a baseline dataset of MIL-STD 1553B network traffic, a bus recorder was used to record the communications of simulated RTs over a physical MIL-STD 1553B data bus shown in Figure 4. The MIL-STD 1553B data bus was setup similarly to the generic test bench and collection architecture also shown in Figure 4. The data collected was that of a “benign” flight, consisting of an aircraft in flight, where no anomalies were enacted. Once the baseline dataset was collected it was used to create a baseline model in the next phase. The collection of anomalous traffic is similar to the collection of the baseline data except for the presence of a rogue device connected to the physical MIL-STD-1553B bus, to simulate a maliciously attached device that produces anomalies on the bus in the form of different scenarios.

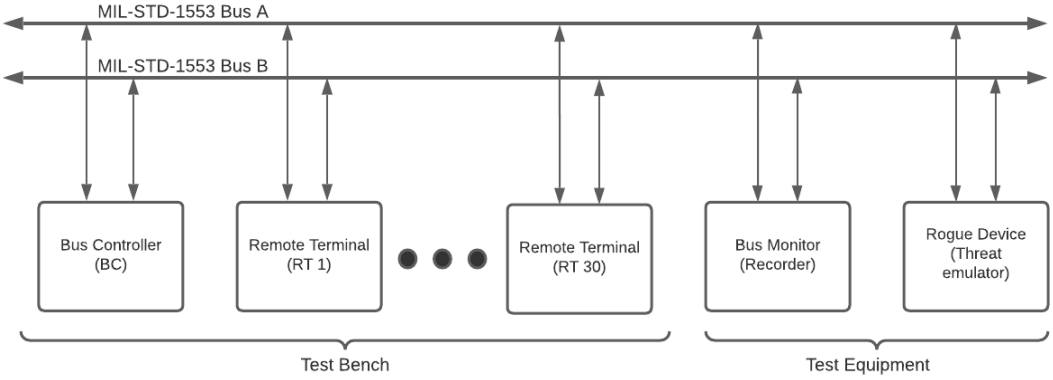

Figure 4: Generic MIL-STD-1553B Test Bench and Collection Architecture

The scenarios were pulled from the following types of attacks:

1. DoS
    a. Network Disruption (Command word DoS)
    b. Network Disruption (Status word DoS)
2. Targeted RT DoS
    a. RT deny
    b. RT subaddress deny
3. Data Integrity Violation
    a. RT hijack

Using the tool presented by Paquet (2014) each of the above five attack scenarios can be carried out on the MIL-STD-1553B bus under test. Each type of attack is described in more detailed in Table 1.

Table 1: Scenario Attack Descriptions

Scenario	Brief Description
1a)	A complete data bus DoS attack where a rogue RT in BC mode uses commands words transmitted on the data bus in order to deny all bus communication.
1b)	A complete data bus DoS attack where a rogue RT in BC mode transmits status words after each normally transmitted command word transmitted on the data bus to deny all bus communication.
2a)	A targeted DoS attack where a rogue RT in BC mode transmits status words after each normally transmitted command word intended for a specified RT to deny communication to or from that RT.
2b)	A targeted DoS attack where a rogue RT in BC mode transmits status words after each normally transmitted command word intended for a specified RT sub-address to deny communication to or from that RT sub-address.
3a)	The implementation of this attack cannot be disclosed due to it being proprietary in nature, however it essentially causes the data for an intended recipient RT to be modified and accepted as if it were the intended values.

The detector was created using the Python programming language with the TensorFlow library as the backend for the LSTM model creation. The LSTM autoencoder model used for this experiment was composed of one input layer, three LSTM layers and one output layer. Sixty-one features were extracted from the MIL-STD-1553B protocol and were used as the inputs for the input layer of the LSTM Autoencoder model. The first LSTM layer was then composed of 30 nodes. The next LSTM layer was then composed of half as many nodes, down to 15 nodes, making the first two LSTM layers, the encoder portion of the LSTM autoencoder network. The last LSTM layer was then expanded back to the size of the first layer, up to 30 nodes, to make the second and third LSTM layers the decoder portion of the LSTM autoencoder network. The final output layer then feeds the results into 61 separate outputs, one for each feature completing the LSTM Autoencoder model.

4. Results

The LSTM autoencoder detector was created using the methodology and design described in the previous section. The detector performs two core functions. The first created a baseline model of the MIL-STD-1553B data bus undergoing analysis. The second function compared subsequent data from the data bus under analysis for anomaly detection. The detector accepts a converted BMDX file (the file format used by Abaco BusTools software) that has undergone feature extraction for baseline creation and anomaly detection in CSV format. The program outputs either a trained LSTM model visualized in Figure 5 or a labelled CSV file that contains all of the messages, the MAE of each message and whether each message was determined to be anomalous based on a set threshold value visualized in Figure 6.

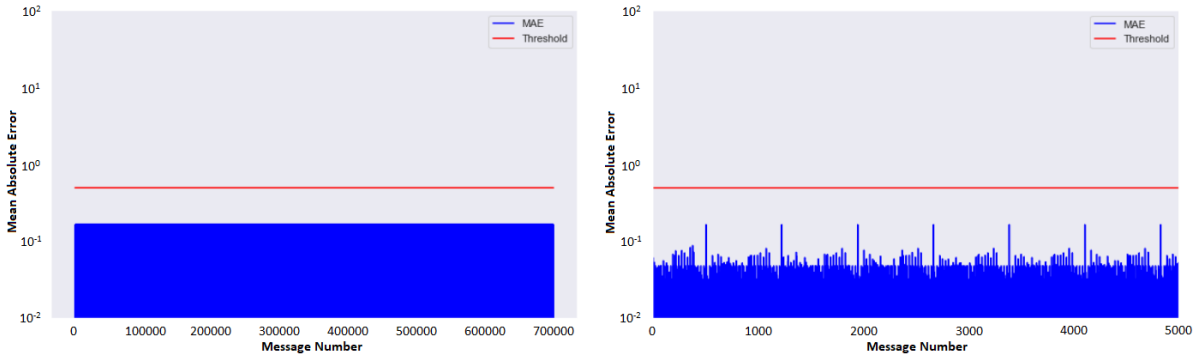

Figure 5: Visualization of Baseline recording (left) same recording scaled down to show fine detail (Right)

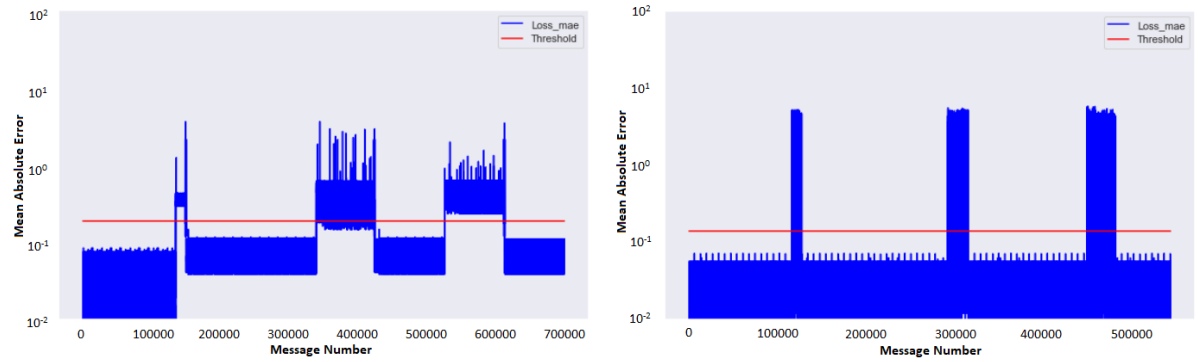

Figure 6: Visualizations of Scenario 1A (left) Scenario 1B (Right)

Figure 5 showcases a baseline LSTM autoencoder model that was created from the dataset. The five scenario recordings that underwent feature extraction were ingested by the model and any of the produced MAE values for each message that fall above the set threshold value were considered to be anomalous. In order to group anomalous messages into events, any messages with a MAE above the threshold were considered a single event once no other messages go above the threshold in a set time interval, in this case a time interval of 30 seconds was used. This allowed for the determination of the start and stop times of the events. Figure 6 showcases the output of the LSTM autoencoder anomaly detector for scenarios 1A and 1B.

All attacks were detected for all five scenarios. When determining only whether an attack has occurred, the event classification accuracy, precision and recall were all calculated to be 100% indicating the LSTM Autoencoder detector had no issues detecting every attack occurrence. Table 2 shows the average detection time accuracy and detection time precision results for each scenario as well as a combined average from all scenarios. The LSTM autoencoder detector detected the start and stop time of the first scenario within 0.0060 ±0.0058 and 0.0060 ±0.0031 seconds, respectively. This means that the detector was within 6 ±6 and 6 ±3 messages of determining the start and stop time of the threat scenarios, respectively. The overall average start and stop time for all combined scenarios is 0.0512 ±0.0354 and 0.2476 ±0.3194 seconds for determining an anomalous event or 50 ±35 and 250 ±320 messages.

Table 2: LSTM Autoencoder Detection Time Accuracy and Precision Results

Scenario	Average Start Time Difference (seconds)	Average End Time Difference (seconds)
1a)	0.0060 ±0.0058	0.0060 ±0.0031
1b)	0.0033 ±0.0011	0.0060 ±0.0075
2a)	0.0045 ±0.0004	0.0039 ±0.0007
2b)	0.04791 ±0.0378	1.1964 ±1.5465
3a)	0.1943 ±0.1322	0.0587 ±0.0391
Average	**0.0512 ±0.0354**	**0.2476 ±0.3194**

The MAIDENS detector used the baseline BMDX file that was converted into CSV format to create a baseline profile. The baseline profile was then used determine the threshold used to detect anomalies in subsequent bus traffic with attack data that was analysed by the MAIDENS detector. For this research all of the MADIENS data was re-run through the detector and the same results were obtained as in the works of Généreux et al (2020). When determining only whether an attack has occurred, the event classification accuracy, precision and recall were all calculated to be 100% indicating the MAIDENS detector also had no issues detecting every attack occurrence. Table 3, shows the overall detection time accuracy and detection time precision results for each scenario. The MAIDENS detector, detected the start and stop time of the first scenario within 0.38 ±0.15 and 47.11 ±37.53 seconds, respectively. This means that the detector was within hundreds to thousands of messages of determining the start and stop time of the threat scenarios. The average start and stop time for all combined scenarios is an average of 0.64 ±0.40 seconds for the start time and 10.27 ±7.98 seconds for the stop time.

Table 3: MAIDENS Detection Time Accuracy and Precision Results

Scenario	Average Identified Start Time Difference (seconds)	Average Identified End Time Difference (seconds)
1a)	0.38 ±0.15	47.11 ±37.53
1b)	0.45 ±0.26	0.89 ±0.13
2a)	0.41 ±0.46	0.78 ±0.32
2b)	0.81 ±0.57	2.26 ±1.77
3a)	1.14 ±0.54	0.3 ±0.16
Average	**0.64 ±0.40**	**10.27 ±7.98**

The LSTM autoencoder anomaly detector outperformed the MAIDENS detector in every scenario. Derived from the data in Table 4, there was an 1054.69% increase in the average detection start time and a 4147.82% increase in the average detection end time when comparing detection time in seconds. In terms of messages, the MAIDENS detector can detect the start time within 640 ±400 messages and the stop time within 10270 ±7980 messages, whereas the LSTM autoencoder detector can detect the start time within 51 ±35 messages and the stop time within 250 ± 320 messages. Based on the median overall average, the LSTM autoencoder anomaly detector is 1058.82% more effective at detecting the start time and 4108% more effective at detecting the end time of an anomalous event in terms of messages than the MAIDENS detector.

Table 4: Average Start and End Detection Times in Seconds

	Average Start and End Detection Time (Seconds)					
	Start Low	Middle	Start High	End Low	Middle	End High
LSTM-Autoencoder	0.0158	0.0512	0.0866	-0.0718	0.2476	0.567
MAIDENS	0.14	0.54	0.94	2.29	10.27	18.25
% Increase	886.076	1054.69	1085.45	3189.42	4147.82	3218.70

There were a couple of drawbacks that were noted during experimentation. The first is the manual setting of a threshold value. For both datasets an acceptable threshold value was able to be set for all anomaly detection. However, depending on how the MIL-STD-1553B protocol was implemented or what state of flight an aircraft is in during recording, this value will likely need to be adjusted accordingly. The setting of the threshold value may require expert knowledge on the system and is a significant factor to ensure the model is detecting only anomalous events. The second drawback noted related to data collection and how the threat emulation tool from Paquet (2014) did not have the ability to tag individual messages as anomalous. The LSTM autoencoder by design is able to tag each message as anomalous or not. Overall, the increase in performance when compared to MADIENS is worth the few drawbacks, with each drawback that could be studied and expanded on further in the future to potentially minimize their effect.

The above results showcase the viability of DL methods as a means of detecting of anomaly on a MIL-STD-1553B Bus. An LSTM autoencoder is just one example of many available DL methods that can be explored for anomaly detection on network architectures like the MIL-STD-1553B protocol. Advances in means of detection

of anomalies can not only inform users of intrusions on their networks, but can also be a means of fault detection for maintenance actions.

5. Conclusion

While the detection of an actual cyber-attack on a physical aircraft has yet to be publicly realized, the consequences of such an event cannot be ignored. There are methods that have been developed for the MIL-STD-1553B protocol and include signature-based, and anomaly-based detection techniques. The aim of this research was to improve the time-related performance metrics when detecting attacks on the MIL-STD-1553B data bus traffic using a LSTM autoencoder deep learning technique.

Not only was it successful in detecting attacks in five different scenarios, but it also significantly outperformed the MAIDENS detector presented by Généreux et al (2020) in both detection time accuracy and detection time precision. The LSTM autoencoder detector indicated the start and stop timings of each anomalous scenario to a hundredth or thousandth of a second with no false positive or negative results. The degree of detection accuracy would be more beneficial, compared to MAIDENS, in a forensics investigation as the detected time are closer to the actual attack time, resulting in far fewer frames to needlessly analyse. This work provides substantial evidence of the applicability of the LSTM autoencoder detector for anomaly detection on MIL-STD-1553B networks.

References

Généreux, S. J. J., Lai A. K. H., Fowles C. O., Roberge, V. R., Vigeant, G. P. M. and Paquet, J. R. (2020) "MAIDENS: MIL-STD-1553 Anomaly-Based Intrusion Detection System Using Time-Based Histogram Comparison", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 56, No. 1, February, pp. 276-284.

ISO 5725-1 (1994), "Accuracy (trueness and precision) of measurement methods and results — Part 1: General principles and definitions", [online], https://www.iso.org/obp/ui/#iso:std:iso:5725:-1:ed-1:v1:en

Lounis K., Mansour Z., Wrana M., Elsayed M. A., Ding S. H. H., and Zulkernine M. (2022) "A Review and Analysis of Attack Vectors on MIL-STD-1553 Communication Bus", *IEEE Trans. Aerosp. Electron. Syst.*, pp. 1–1.

Mahmoud M., Kasem M., Abdallah A. and Kang H. S. (2022) "AE-LSTM: Autoencoder with LSTM-Based Intrusion Detection in IoT", *2022 International Telecommunications Conference (ITC-Egypt)*, Alexandria, Egypt, pp. 1-6.

Maleki S., Maleki S., and Jennings N. R. (2021) "Unsupervised anomaly detection with LSTM autoencoders using statistical data-filtering", *Appl. Soft Comput.*, Vol. 108, p. 107443, September.

"MIL-STD-1553 Designer's Guide" (1998), ILC Data Device Corporation.

Paquet J. (2014) "Uncovering MIL-STD-1553 vulnerabilities: exploitability of military aircraft networks", MASc, Royal Military College of Canada.

Provotar O. I., Linder Y. M., and Veres M. M. (2019) "Unsupervised Anomaly Detection in Time Series Using LSTM-Based Autoencoders", *IEEE International Conference on Advanced Trends in Information Theory (ATIT)*, December, pp. 513–517.

Said Elsayed M., Le-Khac N. A., Dev S., and Jurcut A. D. (2020) "Network Anomaly Detection Using LSTM Based Autoencoder", *Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, Alicante Spain, November, pp. 37–45.

Stan O., Elovici Y., Shabtai A., Shugol G., Tikochinski R., and Kur S. (2017) "Protecting Military Avionics Platforms from Attacks on MIL-STD-1553 Communication Bus", *ArXiv170705032 Cs*, July, [online], http://arxiv.org/abs/1707.05032

Stan O., Cohen A., Elovici Y., and Shabtai A. (2019) "Intrusion Detection System for the MIL-STD-1553 Communication Bus", *IEEE Trans. Aerosp. Electron. Syst.*, pp. 1–1, 2.

Taylor A., Leblanc S., and Japkowicz N. (2016) "Anomaly Detection in Automobile Control Network Data with Long Short-Term Memory Networks", *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, October, pp. 130–139.

Venskus J., Treigys P., and Markevičiūtė J. (2021) "Unsupervised marine vessel trajectory prediction using LSTM network and wild bootstrapping techniques", *Nonlinear Anal. Model. Control*, Vol. 26, No. 4, pp. 718–737.