Cyber Security Case Study of an Operational Ethernet Train Communication Network

Anonymised[a]

[a]

[b]Anonymised

Abstract

This paper presents a cyber security case study of an Ethernet Train Backbone (ETB) Train Control and Monitoring System (TCMS) designed in accordance with the IEC 61375 international standard. We examined its implementation in a modern train, capturing and analysing network packets before simulating attacks at a representative test-bed built by the train manufacturer using genuine train parts. We found that, despite the quality of the implementation and considerable attention given to its security, its adherence to the standard meant we were able to inject spoofed Train Real Time Data Protocol (TRDP) datagrams directly into the TCMS, and that these datagrams were then accepted by the Train Management Computer (TMC). At best, an attack of this type could give rise to confusion and cause the driver to stop the train; at worst, it may lead an operational incident that affects the safety of its passengers. We suggest ways in which the TCMS could be improved, including the addition of a TRDP cryptographic authenticity and integrity verification mechanism, packet filtering, the re-positioning of an intrusion detection system, and an enhancement to the train's physical security.

Keywords: cyber security, packet spoofing, railway safety, train control and management system, train real time data protocol

1. Introduction

1.1. Overview

In this paper, we present a cyber security case study of an Ethernet Train Communication Network (TCN), as installed in a train that was designed and manufactured in the United Kingdom (UK) in the previous five years. This case study looks in detail at the implementation of the Train Control and Monitoring System (TCMS), a sub-network within the TCN that contains components essential for the safe operation of the train. The research established how the network had been secured, and enabled the identification of residual security vulnerabilities that may have an adverse impact on safety. Although this case study focused on a single implementation, the TCMS under examination was designed in accordance with international standards; as a result, the findings and recommendations presented in this paper are transferable to other vehicles built to the same standards. This paper builds on a previous conference paper [1], we discuss how this journal extends the conference paper in the literature review (Section 3).

The research undertaken combined the analysis of design documentation and technical specifications with practical exploration and experimentation, using newly built trains undergoing pre-delivery testing, and a test-bed or *train-in-a-rack* that was representative of those trains. The outcome of this study includes a series of recommendations that, if implemented, will mitigate all the identified vulnerabilities, together with a set of considerations that may influence future train designs.

1.2. Why is the cyber security of a train important?

A railway must operate safely, reliably and efficiently. 'The Williams-Shapps Plan for Rail' [2] describes the UK Government's vision to "maintain safe, secure railways for all". Safe and secure railways require safe and secure trains, and requirements for secure critical national infrastructure means that cyber security is now key to modern train designs. There are several reasons for this, including the desire to operate a railway with trains that offer greater levels of digitisation, the adoption of widely-used network technologies and protocols in the TCMS, and the publication of cyber security guidance, standards and legislation with which original equipment manufacturers and train operating companies are expected to comply. The following sections examine each driver in turn to set the scene for a case study that provides a real-world example of the cyber security properties of a modern train.

1.2.1. Greater digitisation

In the UK, 990 million passenger journeys were made by rail in the year ending March 2022[1], with 16.87 billion net tonne kilometres of freight usage in that period[2]. These journeys were provided by a rolling-stock fleet with an average age of 16.9 years[3]. There are a number of projects to upgrade this fleet, which includes the procurement of new passenger trains. As we shall describe below, these new trains mark a change in the technologies used to provide their TCMS functions, with a

[1] 'Department for Transport Rail Factsheet 2022', published 2nd February 2023. Referencing the Office of Rail and Road (ORR) Road Passenger Rail Usage table 1220. See: `https://www.gov.uk/government/statistics/rail-factsheet-2022/rail-factsheet-2022` (accessed 12th Jan 2024).

[2] Ibid. Referencing the ORR Freight Rail Usage and Performance table 1310.

[3] Ibid. Referencing the ORR Rail Infrastructure and Assets table 6313.

gradual and staged move away from bus, serial and train-wire to switched ethernet and Internet Protocol (IP) networks. New technologies introduce new capabilities, which bring many benefits to passengers and operators alike. However, they also introduce a new set of risks. We must therefore ensure that suitable cyber security controls continue to be developed and applied to this changing technical landscape.

1.2.2. Guidance

There have been several guidance documents that concern cyber security and its application to the railways. The 'Rail Cyber Security Guidance to Industry' [3], published by the Department for Transport (DfT) in 2016, was the first in the UK. Although now showing its age, it acknowledges rolling-stock and provides high-level guidance on secure system design, defensive strategies, and threat and incident management. This guidance was followed in 2017 by the 'Rail and Cyber Security Strategy' [4], written by the Rail Delivery Group (RDG) with input from experts drawn from a cross-section of the rail and cyber security sectors. This strategy provides a vision of how cyber security can be achieved through a change in culture, capability and governance of the industry. At a higher level, the UK Government's 'National Cyber Strategy 2022' [5] mentions the need to "protect and influence" a number of essential service sectors, including transport, to increase cyber resilience. Although lacking in technical details, these documents demonstrate the desire for improvement by both government and industry; as a set, all three show the gradual rise of cyber security and its increasing level of importance.

With respect to practical provision, a growing focus on cyber security can be seen in the development of the UK Key Train Requirements (KTR). The purpose of the KTR is to "assist rolling stock procurers, specifiers, manufacturers and system suppliers to compile procurement specifications" [6]. It contains guidance on all aspects of train design, from components and braking to passenger ergonomics and interior lighting. Annex D of version six, which was written and published the by the RDG in November 2020 [7], is concerned with "Software Security" and as its title suggests, provides software-related guidance. The same annex in version seven, which was co-authored by the UK Rail Safety and Standards Board (RSSB) and released in July 2023 [6], was renamed 'Cyber Security'. The updated annex is more detailed and wide-ranging, and it is structured around the Cyber Assessment Framework (CAF) [8] written by the UK National Cyber Security Centre (NCSC). This useful approach provides significantly greater coverage; it will also help a manufacturer to design and build a train that will allow a Train Operating Company (TOC) to demonstrate its compliance with the UK Network and Information Systems (NIS) Regulations, a requirement placed on all Operators of Essential Services (OES).

As well as producing the CAF, NCSC also publishes guidance to inform the designers of "cyber secure systems". One such publication contains a series of preventative actions that can "reduce the impact of compromise" [9], a statement that assumes a system will eventually be compromised and therefore requires *additional* mitigations in order to limit the resulting damage. This piece of guidance is particularly relevant to this case study, as vulnerabilities in the TCMS are only likely to be exposed and exploited following the initial compromise of an inter-connected system, or through direct physical action. Indeed, as a way to reduce this very impact, we recommend enhanced TCMS cyber security provision in Section 6 of this paper.

1.2.3. Specifications and standards

The most applicable technical specification that links cyber security to the railways is CLC/TS 50701:2023 'Railway applications – Cybersecurity', the current version of which was published in 2023 [10]. Work is underway to transform it into an international standard: IEC 63452 (at the time of writing it is at Draft 2023/08); however, it will be some time before this standard is published, and likely to be several years before it is fully adopted by train manufacturers.

There are other standards that apply to cyber security and rolling-stock, the most well known of which is the IEC 62443 'Industrial communication networks – Network and system security' series [11]. This series is mature and applied in many industrial settings, and forms the basis for TCMS designs. It is particularly useful in its suggestion of separating components into *zones* based on functional, safety and security requirements, connected by *conduits* through which flow carefully considered communications. It also recommends a two-part risk assessment process as the foundation on which the zones and conduits should be arranged.

There are, of course, a range of safety standards that apply to rolling-stock. In the UK, these include EN 50128:2011 'Railway applications – Communication, signalling and processing systems – Software for railway control and protection systems' [12]. Whilst standards such as these acknowledge cyber security, they are primarily concerned with safety, achieved through the verification and validation of the software installed in signalling and its allied systems, rather than the design and integration of networked components to form a TCN. The quality of this software is relevant to the security of a train, but it sits at a level of detail below that researched as part of this case study. Whilst acknowledged as fundamentally important, these standards are therefore considered out-of-scope of this paper.

1.2.4. Laws and regulations

Today, attention has turned to the implementation of cyber security in a way that guarantees compliance with relevant laws and regulations, the most significant of which is the European Union (EU) NIS Directive. The NIS Directive, Directive (EU) 2016/1148, was adopted on the 6th July 2016 [13]. It was adopted into UK law as the NIS Regulations 2018[4], places a set of cyber security requirements on a TOC, making the cyber security assessment of the trains they operate a legal necessity[5].

[4] The Network and Information Systems Regulations 2018: `https://www.legislation.gov.uk/uksi/2018/506/contents`

[5] Within the NIS Regulations, TOCs are known as OES in the railway domain. In England, Scotland and Wales, the *Competent Authority* responsible for its application is the Secretary of State for Transport; however, in practice the DfT acts on his or her behalf.

To assist OES to demonstrate their compliance with the NIS Regulations, the DfT published its 'Implementation of the NIS Directive' guidance in 2018 [14], which was followed by CAF, a method for assessing compliance, published by NCSC in 2022 [8]. CAF is the UK's NIS Regulations assessment framework that applies across every sector that falls within the scope of the Regulations; this includes every TOC and the trains they operate[6].

1.2.5. Summary

These several factors listed above have combined to raise the profile of cyber security within the railways, and with relevance to this paper, to its applicability to rolling-stock. Today, it is seen as an essential part of the train design process and a lens through which to appraise TCMS implementations. As digitisation continues apace, so will the relevance and importance of cyber security.

1.3. The purpose of this paper

The purpose of this paper is to provide details of the cyber security properties of a modern train. In doing so, it is hoped it will increase awareness of the vulnerabilities present in TCMS implementations, and the design decisions available to manufacturers who seek to mitigate them. Although referencing a single implementation, the vulnerabilities described in this paper are likely to apply to *all* modern trains that follow the same industry standards and implement standards-compliant communications protocols.

It is hoped that this paper will contribute to the cyber security improvement process, and stimulate debate amongst train owners, manufacturers and operators, together with those that develop rolling-stock components and contribute towards applicable international standards. We believe it to be beneficial for cyber security vulnerabilities to be considered collectively, rather than to leave them to train manufacturers to deal with in isolation.

It is acknowledged that, by publishing vulnerabilities, malicious parties may turn their attention to the railways and become interested in their exploitation. However, history has shown that secrecy is rarely beneficial, and that the most successful way to improve cyber security is through awareness, information sharing, and open and honest dialogue.

1.4. Paper Structure

In Section 1, we described why cyber security is important to the railways, including trains, and why we have chosen it as the subject of this paper. Section 2 provides an overview of the evolution of the TCMS and explains why ethernet now forms the basis of a modern train. The literature review in Section 3 acknowledges work already undertaken on rolling-stock, the railways more generally, and across the wider transport sector. Section 4 describes the methodology followed whilst undertaking this case study, before in Section 5 we go on to describe the technical details of the TCMS implementation. This is followed by an overview of the cyber security properties of the implementation, including details of how it has been secured, the vulnerabilities it still contains, and any mitigations that could improve its security. Section 6 contains a discussion of the security vulnerabilities identified, how those vulnerabilities could be exploited and by whom, the implications of their exploitation with respect to safety, and how such exploitation could be prevented. Finally, we present our conclusions in Section 7, noting suggestions for further research.

1.4.1. Acknowledgements

We would like to express our grateful thanks to the manufacturer that provided access to its trains, test-bed, documentation and engineering staff during the period of the research.

The manufacturer was not directly involved in the research, other than to facilitate requests for information, access to its trains, test-bed, documentation and engineering staff. It did not contribute to the writing of this paper. The manufacturer provided financial support for this research by part-sponsoring the PhD of the corresponding author.

The train images used in the figures in this paper were taken from the ConceptDraw 'DIAGRAM' application and used with the kind permission of Computer Systems Odessa.

2. Evolution of Train Communications Networks

2.1. Setting the scene

Before considering trains with an ethernet TCMS, with which this paper is concerned, we will first provide a brief overview of the evolution of TCMS technologies, and in doing so introduce some terminology that will be used in the remainder of the paper.

2.2. Brake-pipes and train control circuits

Early trains operated in a simple *pull-push* configuration, with a locomotive (or power-car) pulling or occasionally pushing the connected cars. As trains became longer and heavier, additional braking was required to supplement that provided by the locomotive. This was achieved using a pneumatic *brake-pipe* that ran the length of the train and connected the brakes distributed throughout the cars. For freight trains, this simple configuration is sufficient, although it may be supplemented by additional locomotives as is often the case in the US, where multiple locomotives are connected by a wired or wireless communication link to provide coordinated braking and power application. The doors in early passenger trains could be opened and closed by passengers on demand, even when the train was in motion. Central locking was added as a safety feature, which allowed a member of the train-crew to enable and disable the doors as circumstances required. This was achieved using a train control circuit (also known as a train-line or train-wire): a relay-based system facilitated by dedicated cables running from

[6] Directive (EU) 2022/2555, known as NIS2, came into force on the 16th January 2023 [15]. Member States must transpose the Directive into national law and apply it from the 18th October 2024, on which date Directive (EU) 2016/1148 will be repealed. The UK left the EU on the 31st January 2020 and it is currently unclear whether a similar UK law will be enacted to mirror NIS2.

the driver's cab and distributed control panels to each set of doors. A voltage applied to a train control circuit indicated the train's speed, which determined whether the door was enabled or disabled. Like the brake-pipe, the train control circuit is still in use today, and when deployed together form basis of a simple and safe train configuration.

2.3. Vehicle and train busses

As trains began to digitise and be equipped with a new generation of intelligent and distributed components, including those providing power and braking, an alternative to the train control circuit was sought[7]. The eventual replacements were: the Multifunction Vehicle Bus (MVB), described by zur Bonsen in [16] and standardised in IEC 61375-3-1:2012 [17], a "deterministic and robust" bus capable of operating in "harsh and disturbed environments"; and the Wire Train Bus (WTB), standardised in IEC 61375-2-1:2012 [18] and suitable for "frequently coupled and uncoupled" trains[8].

Before we describe these busses in more detail, we must first define the terms *consist* and *end device*. A consist is a single car, or collection of cars that are not separated the during normal operation of a train. A Consist Network (CN) is found within a consist and is used to connect components, each of which is known as an End Device (ED), that are physically located within that consist. A CN always remains within its host consist and does not extend beyond its physical boundaries. The section of a train shown in Figure 1 contains two consists: the locomotive is Consist 1; the trailing two cars are Consist 2. The architecture in this example means that the locomotive could be uncoupled and moved to the rear of the train, but as the trailing two cars from a single consist, they could not be separated from one another without the manual reconfiguration of the CN. From this description, two additional terms arise: *train-bus* and *vehicle-bus*: a train-bus runs down the full length of the train connecting together the various vehicle-busses that form the CNs.

In the example shown in Figure 1, a single WTB train-bus connects the two MVB vehicle-busses, and together they form the TCMS[9]. In an operational train, the WTB would normally be installed in a redundant configuration with respect to cabling and node deployment; nodes usually operate as on-line and standby pairs, and fail-open to ensure the continuation of communications. The WTB has a maximum length of 860m, can support up to 32 connected nodes, and provides a data rate of 1Mbps. The WTB includes an *inauguration* process that allows a train to operate following the addition, removal or re-arrangement of its constituent cars (whilst maintaining any consists and their CNs previously configured). This involves the automatic numbering of the WTB nodes and the selection of a bus master. Should the bus master fail, any other WTB node is able to take-over to ensure communications are maintained. Because of its flexibility, the WTB is usually found in *open* trains that see regular changes made to their configuration[10]. The WTB will convey both process and diagnostics data, with the data definition including commands to start and stop the engine, enable and disable the doors, apply power and control the brakes. This allows it to operate without the presence of a secondary vehicle-bus, although it is often combined with the MVB to provide connectivity within individual consists.

The MVB vehicle-bus can be combined with a train-bus, such as WTB, to connect CNs in an open train; alternatively, in a closed train it may be the only bus in operation. Configuration of the MVB is usually performed at the point of manufacture, or at the depot where the train is maintained. The MVB has a maximum length determined by its physical media: 20m (RS-485 wire-pair); 200m (shielded twisted wire-pair); or 2,000m (optical glass fibre). The MVB can have a maximum of 4,095 connected nodes, which may be sufficient in a closed train, and provides a data rate of 1.5Mbps. Should a greater number of nodes be required, multiple MVB instances would be fitted and connected by a train-bus. Like the WTB, the MVB has a bus master, which changes in the event of bus master failure, and also periodically. It will carry deterministic and real time process data, on-demand message data, and supervisory data to support the management of the bus master.

There is no cryptographic security in either the WTB or MVB: messages are not encrypted for confidentiality, nor do they contain a cryptographic authenticity and integrity verification mechanism.

2.4. Ethernet Train Backbone (ETB)

2.4.1. Introduction

As technology progressed, so did the requirement for trains to support increasing numbers of intelligent components. This included those providing safety-related functions in addition to power, braking and door control, such as the European Train Control System (ETCS) to which we will return later, and fire detection. It also incorporated a range of non-safety-related components including auxiliary and battery power, Heating,

[7] In practice, these alternatives are often used in conjunction with the train control circuit, rather than as a replacement of it. In such cases, the train control circuit may operate as a secondary safety system, for example by advertising the current train speed, which may be referenced by components including external door controllers before making a decision to enable or disable a door.

[8] A number of alternative busses are also found in rolling-stock, including those based on CANopen, LonWorks and Profibus. Unlike MVB, which was intended to serve as a replacement for the various buses, they have wider applications in industry; their use is not restricted to trains and they are consequently cheaper to procure.

[9] In the IEC 61375 standard series, the term TCN is used to encompass all the networks installed within a train. In this paper, we have used the term TCMS to make it clear that we are referring to a sub-network within the TCN that supports the control and monitoring of components deemed essential for the safe operation of the train, such as brake and external door controllers. The TCN may integrate both TCMS and non-TCMS components, such as the Passenger Information System (PIS), into a single routed network as the standard permits; however, it is typical for them to be physically separated or connected by means of a security control, such as a gateway or firewall.

[10] An *open* train may see changes made to its configuration, such as the number of cars it contains. Open trains are therefore used to provide regional and national high-speed services, as they can be quickly reconfigured as demand requires. A *closed* train would not normally see any changes made to its configuration: it would remain in a fixed formation. Closed trains are therefore suited to providing underground, metro and tram services.

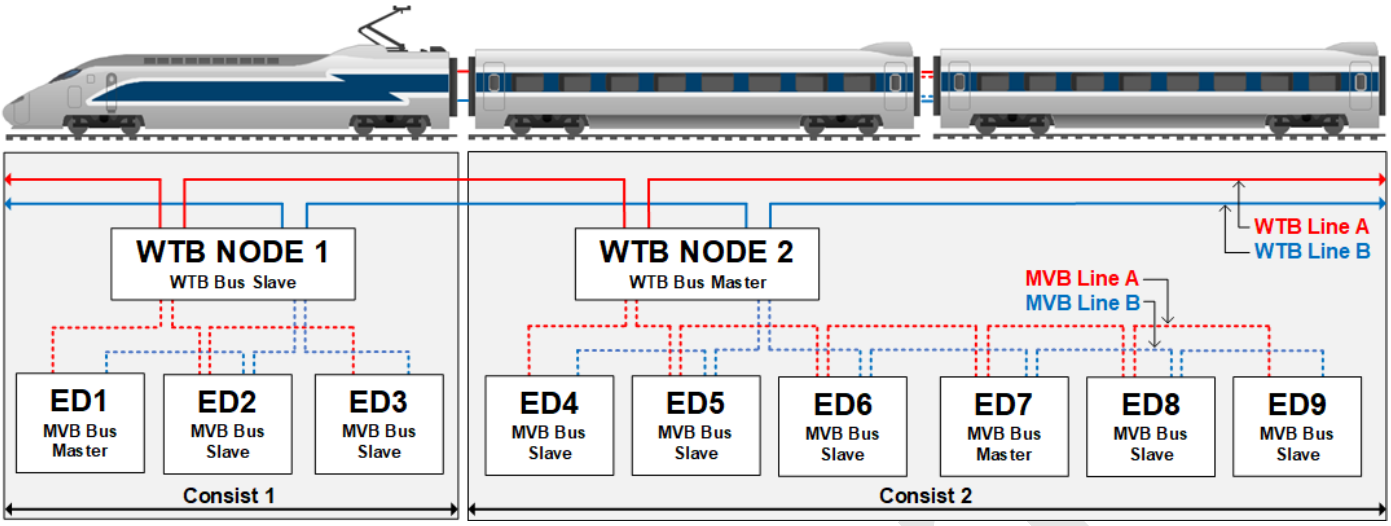


Figure 1: An example Wire Train Bus (WTB) and Multifunction Vehicle Bus (MVB) Train Communication Network (TCN)

Ventilation and Air Conditioning (HVAC), and internal lighting. There was also a need to facilitate greater levels of monitoring, diagnostics, recording and reporting, together with a whole range of PIS including information display screens, passenger counters, seat reservations displays, Closed-Circuit Television (CCTV), public address and intercoms. The extent of this list demonstrates why the WTB and MVB, with their low data rates, were no longer thought to be sufficient to form the basis of a modern TCMS.

The growth in on-board communications was mirrored by an increase of those passing between the train and the wayside[11]. These communications support the operation of the train itself and include signalling data, together with connections to fleet management systems that facilitate real-time diagnostics. They also carry PIS data of the types listed above, and also passenger wifi which is usually physically or logically separated from all other networks. Whilst many of these communications originate from non-TCMS systems, a number support TCMS safety-related functions. All these communication enter and exit the train using a variety of technologies including satellite broadband, wifi (usually used at a maintenance deport) and Global System for Mobile Communications – Railway (GSM-R)[12]. Whilst they will be acknowledged where applicable to the TCMS described in this case study, wayside communications did not form part of the research and are therefore considered out-of-scope of this paper.

The Ethernet Train Backbone (ETB) continues the practice of operating two separate busses or networks: previously the WTB train-bus and MVB vehicle-bus; and now the ETB train-network and Ethernet Consist Network (ECN) vehicle-network[13]. Ethernet represents the current technology of choice when designing a modern TCMS. This can be witnessed in new train designs, including the one that forms the basis of this case study, and also in research looking beyond the ETB as currently specified, and to support future enhancements. For example, the EU *Shift2Rail* programme, a research and development collaboration funded by its participants and the European Commission, is considering next generation ethernet TCMS implementations that are better suited to carrying safety-related data [19], as well as supporting concepts such as virtual coupling [20] and remote driving [21].

2.4.2. ETB train-network

IEC 61375-2-5:2014 [22] specifies the ETB, which like the WTB runs down the full length of the train. This part of the standard defines its physical properties, together with details of its architecture and addressing. The backbone of the ETB differs from more traditional routed networks in that it is linear: it does not comprise loops or trees. Each router, which is called an ETBN, is connected to each of its two neighbours by a single ethernet cable supporting a speed of 100Mbps at full-duplex, or an aggregated set (up to a total of four) configured in accordance with IEEE 802.1AX to provide redundancy and additional capacity[14]. In the case of the end consists, their ETBNs have only one live backbone interface; their second is connected

[11] In the railways, wayside equipment refers to that installed anywhere but on the train itself. Examples include track-side communications cabinets, together with servers managed by train owners, manufacturers, TOCs and maintenance service providers.

[12] The GSM-R a radio communications system that supports both voice and data services. It is used across Europe to provide a standard form of railway communications, including those passing between a train and the wayside, and is the bearer for ETCS.

[13] It may be useful to note that the terminology used when describing ethernet trains can be somewhat confused, as 'ETB' can be interpreted in several different ways. In this paper, 'ETB' shall refer to the combined ethernet train and vehicle network architecture, including all Ethernet Train Backbone Node (ETBN)s and EDs that have unique IP addresses and can therefore be addressed from any part of a single ETB instance; whilst 'backbone' shall refer to the linear arrangement of ETBNs that form the *core* of the network.

[14] This standard form of link aggregation first appeared in the year 2000 as IEEE 802.3ad-2000. It was revised most recently in 2020, becoming IEEE 802.1AX-2020 'Link Aggregation' [23]. IEC 613752-5:2014 refers to 802.1AX-2008, rather than the most recent 802.1AX-2020 revision.

to the coupler unit through which additional consists, and therefore ETBNs, may be added. Alternatively, and as applies to the TCMS in this case study, the ETB may be duplicated, with or without aggregated links, and with single or dual-homed EDs connected to one or both instances.

The IP subnets to be used on the ETB are clearly defined in IEC 61375-2-5:2014. The standard states that addresses in the `10.0.0.0/9` subnet are to be assigned to 'localized subnets', such as those at the wayside or in an ECN, and their use is considered to be out-of-scope of this part of standard; the `10.128.0.0/9` subnet is reserved for the ETB itself, and the manner by which this address space is further decomposed is also specified. For example, within the `10.128.0.0/9` subnet, a TCN may comprise four ETBs, two of which provide the TCMS and 'multimedia' networks (subnets `10.128.0.0/11` and `10.160.0.0/11` respectively), whilst the remaining two are 'not specified' (subnets `10.192.0.0/11` and `10.224.0.0/11`). Like the WTB, the ETB is able to cope with changes made to its length. It is therefore found in open trains, and whilst it is capable of operating without a vehicle-network, it may be combined with the ECN that fulfils this purpose. Topology changes are managed by an *inauguration* process that is initiated by the driver when the train is started or reconfigured. At the end of this process, every ETBN is numbered sequentially from the front to the back of the train, and these numbers are present in the IP addresses they are subsequently assigned (the first ETBN in the TCMS ETB would be `10.128.0.1`, the second `10.128.0.2`, and so on). The ETBN situated in the end consist with the lowest Universal Unique Identifier (UUID) becomes the first or 'top node' (`10.128.0.1` in the previous list), and is said to be at the front of the train—a fact that determines the reference train direction[15]. The inauguration process is supported by the Train Topology Discovery Protocol (TTDP), which requires individual ETBNs to advertise details of their connected neighbours until all the ETBNs in the network share an identical routing table. This routing table is then used to form a train directory, which is passed through the IEEE 802.3 Cyclic Redundancy Check (CRC) CRC-32 function to produce a 4 byte unsigned integer that represents the current train topology. This integer is placed in the Train Real Time Data Protocol (TRDP) header, which is described in Section 2.4.4, to allow a receiving or routing component to determine whether the sending component was aware of the current train configuration at the time the Protocol Data Unit (PDU) was sent[16].

Multicast addresses in the subnet `239.192.0.0/24` are also defined in IEC 61375-2-5:2014, and these are used to provide bus-like broadcast communications. Indeed, as we shall see in the case study that follows, multicast addressing is the means by which communications pass between EDs.

The major advantage of the ETB, which addresses the shortcomings of its predecessor, is its enhanced length of 100m between ETBNs, its ability to support up to 63 ETBNs distributed throughout a maximum train length of 6,300m, and its data rate of 100Mbps, which can be extended to 1Gbps and beyond (although such increased speeds are not defined in the standard).

2.4.3. ECN vehicle-network

The ECN vehicle-network is defined in IEC 61375-3-4:2014 [25]. This network is similar to the MVB in that it must remain within its host consist. In its implementation in an open train configured with an ETB, an ECN is a Local Area Network (LAN) that connects to other ECNs by means of the ETB. Within an ECN, IP addresses in the `10.0.0.0/9` subnet must be used, and these may be repeated in different ECNs distributed throughout the train. In order to prevent conflicts and to ensure that only valid IP addresses are found on the ETB (i.e. those in the `10.128.0.0/9` subnet), ETBNs act as *gateways* that perform Railway Network Address Translation (R-NAT) to traffic leaving the ECN, with R-NAT being fully defined in this part of the standard[17]. An ECN need not support dynamic configurations, and EDs in ECNs may therefore be configured with static IP addresses; alternatively, ECN IP addresses may be assigned using the Dynamic Host Configuration Protocol (DHCP), usually by a service running on the locally connected ECN-ETBN interface. In closed trains, the ECN may operate as the only network; there is no requirement for it to be paired with an ETB. The redundancy of the connection between an ECN and the ETB is achieved by deploying multiple ETBN gateways. In such a configuration, the ETBNs use a protocol such as the Virtual Router Redundancy Protocol (VRRP) to provide a single default route between the networks. Redundancy of the ECN itself is achieved through the installation of multiple ethernet switches. IEC 61375-3-4:2014 specifies a number of options, from a simple 'linear' through to a more complex 'ladder' switch topology with ED dual homing providing a continuous service following the failure of one or more of the switches.

2.4.4. Communication and application profiles

The previous two parts of the standard covered the architectural aspects of the ETB, whilst its communications profile and the protocols it carries are defined in IEC 61375-2-3:2015 [26]. The sections of this document that are relevant to this case study specify TRDP and its Process Data (Pd) and Message Data (Md) variants. TRDP is the protocol by which EDs communicate, and is the only data protocol permitted to traverse the ETB. It does incorporate any cryptographic mechanisms to guarantee confidentiality, authenticity or integrity.

[15] According to IEC 61375-2-5:2014, a consist UUID, known as a CstUUID, "is used to uniquely identify a consist in the world without the need of a central registration process." The CstUUID is a 16 byte identifier defined in accordance with RFC 4122 [24].

[16] As defined in IEC 61375-2-3-2015, this UINT32 header is called `etbTopoCnt` and found in both Pd and Md TRDP PDUs.

[17] R-NAT follows the IP address specifications defined in IEC 61375-2-5:2014 and translates IP addresses as packets pass from an ECN to the ETB, and visa-versa. It is the same as *traditional* one-to-one Network Address Translation (NAT) and allows duplicate IP addresses to be used within different ECNs, with the writing of a valid ETB IP address over the original ECN source IP addresses at the point of out-bound translation (the process is reversed for in-bound translations).

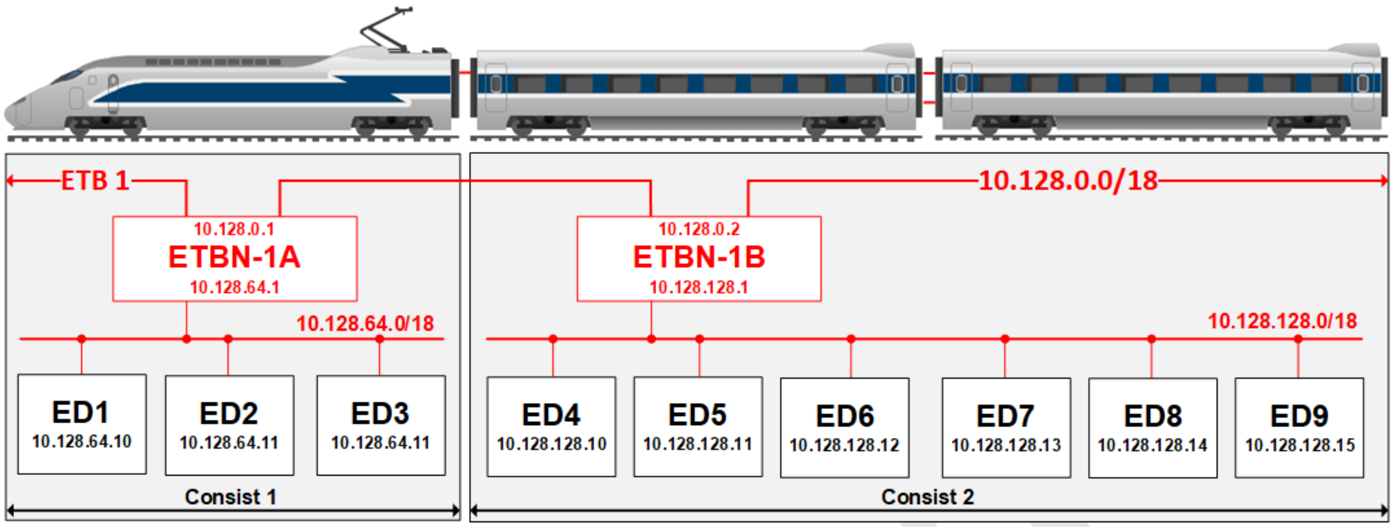


Figure 2: An example Train Control and Monitoring System (TCMS) Ethernet Train Backbone (ETB) Train Communication Network (TCN)

The TRDP Pd PDU is encapsulated into a single User Datagram Protocol (UDP) datagram. It carries periodic process data that is "cyclically transmitted between a publisher and one or many subscribers". It is connection-less in nature, although two PDUs are often combined as a *request-and-reply* pair sent between corresponding EDs. Its destination addresses can be unicast when known and multicast when not, and its assigned destination port is 17224 (i.e. the port to which a request is sent, and the source port of its corresponding reply). It has a 40 byte header that contains a 4 byte IEEE 802.3 CRC-32 `HeaderFCS` computed over the header only (the `Dataset` or payload is not included) and a maximum length is 1,472 bytes, which when further encapsulated into an IP datagram becomes 1,500 bytes: the maximum payload of a single ethernet frame. An example Pd PDU is given in Figure 3, to which we shall return later (the current IEC 61375-2-3:2015 version is shown on the right).

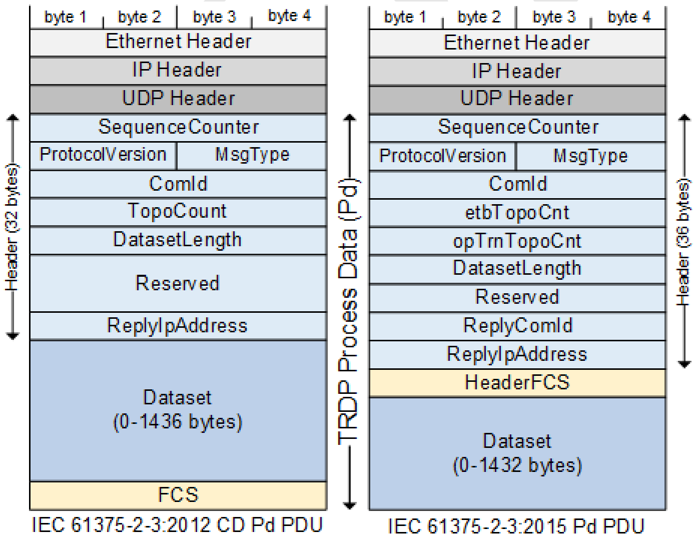


Figure 3: IEC 61375-2-3:2012 CD and 61375-2-3:2015 [26] Pd PDUs

The TRDP Md PDU can be encapsulated into a Transmission Control Protocol (TCP) segment *or* UDP datagram. It carries sporadic message data between "a caller and one or may repliers over a confirmed TRDP service." It provides an event-driven and guaranteed service, and the caller is able to define whether a reply is required or not. As applied to the Pd PDU, unicast and multicast addressing is acceptable, and the destination port assigned to both TCP and UDP variants is 17225. The Md PDU has a larger header than its Pd counterpart, being 116 bytes in length, which also includes a `HeaderFCS`. This PDU has a maximum size of 65,504 bytes and is therefore suitable for conveying large quantities of data.

Following the communications profile, IEC/TS 61375-2-4:2017 [27] details the application profile for communications belonging to the TCMS. It provides a data interface based on the communication profile defined in IEC 61375-2-3:2015, allowing for the interoperable coupling and uncoupling of compliant vehicles.

2.4.5. *ETB evolution*

In the sections above we described the evolution of the TCMS, charting its journey from train control circuits through busses to switched ethernet. ETB designs have and will continue to evolve, and an attempt to capture this is given in Table 1 at the end of this paper. In this table we describe three ETB generations: first, where the ETB is used to *monitor* EDs; second or *transition*, where it is also used control selected EDs; and third, where the ETB is responsible for all aspects of monitoring and *control*. We believe that categorising the networks in this way is useful as it allows us to consider the advantages and disadvantages brought about by greater digitisation, and how this will affect the safety and cyber security status of a modern train. This generational categorisation applies to new train designs; and also to older fleets as their TCMS implementations are fully or partly upgraded during a programme of modernisation. It is intended that Table 1 will aid the discussion in Section 6, when we consider the risks of potential future TCMS implementations.

2.4.6. *Flexibility*

IEC 61375-2-5:2014 and its associated parts described above, set out a general template for how an ethernet TCN is to be constructed. In order to support the interoperability of trains that were designed by different manufacturers, the standard series is very specific in certain respects, for example in how it prescribes particular IP subnets and use of the TRDP protocol. However, it is also flexible and allows an ETB to take various different forms. This means that whilst there are a large number of possible designs, they will all share many common characteristics. This will ensure that whilst this case study has focused on one particular implementation, its findings will apply to many other TCMS designs.

3. Literature Review

This paper builds on a previous conference paper [1], which presented the train network attack discussed in Section 5, along with a detailed attack taxonomy for train network attacks, aimed at cyber security academics and professionals. The contributions of this journal paper, beyond the initial conference paper, include the full details that an engineer would need to understand a train network, a summary of relevant rail regulations and a detailed description of stop gab mitigations that rail companies can deploy while waiting for a MAC to be added to the train network standard.

An early description of a combined WTB and MVB architecture was given by Kirrmann and Zuber in [28]. In their work, the authors noted the requirement for the TCN to transfer "process variables and messages" in a deterministic manner and in accordance with mandated periods, and its ability to facilitate the "plug-in interchangeability of equipment and vehicles". Gemmeke provided an earlier and similar overview, noting the growth of electronic sensors in the late 1990s and the need for them to be incorporated into a standardised TCN [29]. In more recent times, Ludicke and Lehner focused on the move towards a packet-switched TCN and their paper includes a useful overview of the ETB [30]. In it, they also look towards future wireless train communications and their suitability to support advanced concepts such as virtual coupling. Jakovljevic et al., following research undertaken as part of the EU's 'Shift2Rail' programme, propose a next generation TCMS [31]. They describe the limitations of a current ETB implementation with respect to its ability to convey real-time data, and the mixture of safety and non-safety-related messaging in a single and converged network. Whilst only touching on security, they propose a "drive-by-data" alternative to the ETB as specified in IEC 61375-2-5:2014. In a similar way, Astarloa suggests a future ETB based on Time-Sensitive Networking (TSN) as a way to "overcome the identified limitations" of current implementations [32]. Time-Triggered Ethernet (TTE), an alternative to TSN, "unites real-time and non-real-time traffic into a single coherent communication architecture" and is discussed by Kopetz et al. in [33]. Potential use cases in aviation suggest its future applicability to the railways; however, Loveless et al. were able to cause a "failure of critical systems" connected by this technology after launching their novel 'PCSPOOF' attack.

With regard to cyber security, Kour et al. performed a survey of its application to the railways by reviewing 90 papers published between January 2013 and August 2021 [34]. They noted research in the area of rail infrastructure, in particular signalling systems, whilst identifying a lack of papers focusing on "other important assets" such as rolling-stock and Supervisory Control and Data Acquisition (SCADA) systems. Valdivia et al. took this further by stating that cyber security is a "forgotten issue in the railways" [35]. They observed that train manufacturers do not always follow standards such as IEC 62443 when designing a TCN "since the railway industry does not require achieving a specific security level to implement a system", before advocating the linkage of safety and security "to reduce and, if possible, eliminate risk". As a conclusion, the authors recommend the implementation of "message encryption" to improve security, providing it does not compromise existing safety certifications. A similar review was performed by Okstad et al., who commented on the railway's "preoccupation with safety" whilst noting it had "less of a focus on cyber security" [36]. Here, the authors suggest the use of independent assessors as a way by which to understand the application of cyber security in this domain, as occurred during this case study; also, how cyber threats change faster than those that affect only safety, and how this fact will have an impact on the future reliability of safety related systems.

At an applied level and working with a TCMS component manufacturer, Holmberg evaluated a number of modelling methodologies to better understand the threats faced by a TCMS [37]. In doing so, they were able to identify several theoretical ETB vulnerabilities, including the "lack of verified connection between two communicating entities". The implications of this were not investigated further or verified by the development of an attack in a real-world setting. Despite this, the thesis explains how a threat modelling process can assist with the identification of TCMS vulnerabilities and their associated attack vectors. Whilst primarily focused on signalling systems security, Gordeychik describes how an attacker can modify "bus-wire data" to inject "specially crafted commands" to take control of track-side components, before postulating various theoretical vulnerabilities in MVB and ETB TCMS implementations [38]. It was suggested that an attacker could become a MVB bus master, and that TRDP is vulnerable to "forgery and session hijacking". In the latter case, it was stated how spoofed datagrams could be injected into the TCMS by "guessing" sequence counter values; we were able to demonstrate a similar attack during this case study, although our method did not require any form of "guessing" for it to prove successful. As a way by which to detect attacks launched against the TCMS, Yue et al. proposed an Intrusion Detection System (IDS) that uses an 'ensemble' of techniques to increase the likelihood of success [39]. Whilst the attacks described in the paper were *noisy*—they included port-scanning and denial-of-service—their detection rate was 97.5%, which demonstrates the potential benefit of this type of security control.

Generation	Description	End Devices	Advantages	Disadvantages
First *(Monitor)*	The ETB is used to monitor EDs. Displays showing safety-related information to the driver, such as the train speed, are updated over non-ETB connections (the data providing this information may be duplicated over the ETB and shown on additional ETB-connected displays).	EDs are controlled over non-ETB connections, such as the train control circuit. To support their monitoring, EDs have a second connection to the ETB, either directly or by means of an interface unit that translates between TRDP and legacy communications technologies. Monitoring status requests, such as those sent by a TMC, are submitted over the ETB; replies are returned in the same way.	Clear separation of control and monitoring functions. Compromise of the ETB will not allow an attacker to control an ED. Attacker requires knowledge of, and access to, a range of communications technologies. Failure of an ETB network component, such as a switch or router, will not affect the operation of the train (although the monitoring of the EDs may be degraded).	Increased TCN complexity, as multiple and different communications technologies are present on the train. This may increase the: equipment and integration costs; space required to house the interface unit and cables; weight of the train resulting from the different technologies and their interfaces; and maintenance and running costs. Use of the insecure TRDP protocol will allow an attacker to spoof all monitoring messages.
Second *(Transition)*	The ETB is used to monitor EDs, and to control a subset of those EDs. Displays showing safety-related information to the driver are updated over ETB or non-ETB connections.	EDs are ethernet-enabled and are monitored over the ETB (without the need for an interface unit). A subset of EDs are also controlled over the ETB (possibly with the use of an interface unit); the remaining EDs are controlled over non-ETB connections.	Provides a pathway for the staged transition from a first to third generation ETB. Selective integration of control and monitoring functions reduces TCN complexity. Attacker still requires knowledge of, and access to, a range of communications technologies. May see a small reduction in: equipment and integration costs; space requirements; train weight; and maintenance and running costs.	Enduring (although reduced) TCN complexity caused by residual *legacy* communications technologies. Compromise of the ETB will allow an attacker to attempt to take control of a limited set of EDs. Failure of an ETB network component may affect the operation of the train. Use of the insecure TRDP protocol will allow an attacker to spoof a some control and all monitoring messages.
Third *(Control)*	The ETB is used to monitor and control EDs. Displays showing safety-related information to the driver are updated over the ETB.	EDs are controlled and monitored over the ETB. There is no requirement for an interface unit.	Convergence of control and monitoring functions further reduces TCN complexity, and enables the integration of all ethernet-enabled EDs into a single ecosystem. Will see a reduction in: equipment and integration costs; space requirements; train weight; maintenance and running costs.	Attacker requires knowledge of, and access to, a single communications technology. Compromise of the ETB will allow an attacker to attempt to take control of all EDs. Failure of an ETB network component may affect the operation of the train. Use of the insecure TRDP protocol will allow an attacker to spoof all control and monitoring messages.

Table 1: Three generations of Ethernet Train Backbone (ETB), which describes how implementations are evolving in train designs

Looking towards train control, Bendele et al. performed a risk assessment of a UK European Rail Traffic Management System (ERTMS)-based signalling system [40]. Their methodology incorporated an analysis of the trust relationships that exist between on-board and wayside systems. As detailed results are not provided in the paper (they were deemed to be of a "propitiatory" or "sensitive" nature), it is unclear whether the analysis included the interface between the ERTMS and Train Management Computer (TMC), or the data shared between these important components. This is unfortunate, as these communications are examined in detail in this case study. It was, however, interesting to note that the authors described how local attacks are "more difficult to mitigate" than those launched remotely, and that a risk assessment should therefore include internal interfaces in order to provide an understanding of "the damage that could be done to the system" following its compromise. These two statements are in agreement with the results of this case study, and are important considerations when undertaking a risk assessment or broader cyber security assessment. With a focus on ERTMS communications, Chothia et al. proposed an attack against the data passing between a train and the wayside [41]. ERTMS, which incorporates ETCS as its signalling and train control component, performs speed monitoring and calculates the maximum speed at which a train can safely travel. The authors stated that their attack, which targeted the radio component of the system, would give an attacker a 1% change of taking control of a train, given a sufficient quantity of network traffic. They went on to consider several mitigations, and subsequently proposed an ERTMS cryptographic key management and distribution scheme [42]. In a more recent paper, Köcher described how security had been applied to ERTMS, noting that it had not been "specifically addressed" in the interoperability specifications [43]. The author explained how the ERTMS Security Core Group (ESCG) had subsequently developed controls for current and future implementations, which included a Public Key Infrastructure (PKI) to support secure communications. The complexity introduced by cryptographic key management was noted, and how "complete PKI centralisation would appear unlikely", given the the desire of train operators to remain autonomous.

The disruption that may follow the compromise of a TCMS, in what ever form that compromise might take, was highlighted recently in the case of the Impuls 45WE trains manufactured by Polish company Newag and operated by the Lower Silesian Railways[18]. In this example, the trains began to suffer mysterious malfunctions when they had not been submitted to the manufacturer for routine maintenance. An investigation by Dragon Sector, a Polish 'hacker group', required significant specialist work to reverse-engineer elements of the TCMS, and resulted in their allegation that the malfunctions had been deliberately introduced by the train manufacturer to hamper third-party maintenance attempts. Whilst the case is on-going, it demonstrates the outcome of a potential compromise with respect to its impact on train owners, operators, maintainers and passengers: contractual penalties, lost revenue, service disruption and reputational damage, not to mention any safety-related incidents that could result from the inclusion of unauthorised code.

The review of previous and related work revealed that there *is* interest and on-going research in the application of cyber security to the railways. However, this tends to be focused on infrastructure and communications, and with respect to rolling-stock, on standards and evaluation criteria, the components they contain, and future technologies and architectures. The reason for this is likely to be the difficulty of obtaining and examining a modern train: trains, especially from new fleets and fitted with an ETB, are prohibitively expensive and built to meet specific customer requirements; they have high levels of utilisation, with little *down-time* during which to conduct a detailed study; and the understandable reluctance of manufacturers and operators to subject their vehicles to independent scrutiny. Consequently, we believe the research we have undertaken is the first of its kind, and we hope it will stimulate similar collaborations and publications in the future.

4. Methodology

4.1. Introduction

The following section describes the methodology followed during the research. It included several distinct stages, from initially reviewing high-level cyber security documentation, to practical experimentation in a train and test-bed environment.

4.2. Initial documentation

Initially, the train manufacturer made available a design report that provided a high-level view of the TCN architecture, and described how a cyber security control had been retrospectively added to the ETB of a particular class of train: the insertion of a security gateway between the TCMS and all other networks. This was followed by a meeting with the report's authors, during which the network architecture and operation of the components it contained were described.

After the meeting, a number of additional high-level documents were requested and provided. These included reports that provided: a high-level overview of the overarching cyber security context, requirements and plan; the original train requirements analysis undertaken prior to design and manufacture; the summarised results of a two-part threat analysis study that led to the addition of the security gateway; the penetration testing plan (the actual penetration test had not taken place when the research was undertaken); the lock-and-key matrix referring to the storage compartments on the train in which the components were housed; and the register containing all previously identified risks with mitigations where appropriate. These documents were very much *work-in-progress*, with several at the draft or early release version stage. The information provided was commensurate with that typically given to a penetration tester performing a grey-box penetration test, the purpose of which is to reduce the time required to perform the task and therefore increase the tester's efficiency.

[18] See 'Dieselgate, but for trains—some heavyweight hardware hacking': https://badcyber.com/dieselgate-but-for-trains-some-heavyweight-hardware-hacking/ (accessed 18th Apr 2024).

4.3. Trains

Following an examination of the documentation, access was given to two trains that were visited on three separate occasions (the trains were of the same class and had an identical number of cars). The trains visited were complete but had not yet been delivered to their eventual owner; they were undergoing final testing and upgrading during the visits. We were able to connect laptop computers to the networks they contained, although this was not always possible due to concealed cabling, inaccessible components or safety restrictions. For safety reasons, the trains remained stationary at all times.

As the trains had been fitted with structured cabling, often with connectors and cables obscured or hidden, connectivity to the TCMS was undertaken either at ETBN maintenance ports (a third active interface on selected ETBNs that gave access to all network traffic passing through the components), or by temporarily breaking an existing connection and inserting a switch between a component and the ETBN. The insertion of the switch provided additional ports in which to connect, and the ability to collect traffic originating from, or destined to, a specific component. With this level of access, it was possible to map the network and capture large numbers of TRDP packets for later analysis.

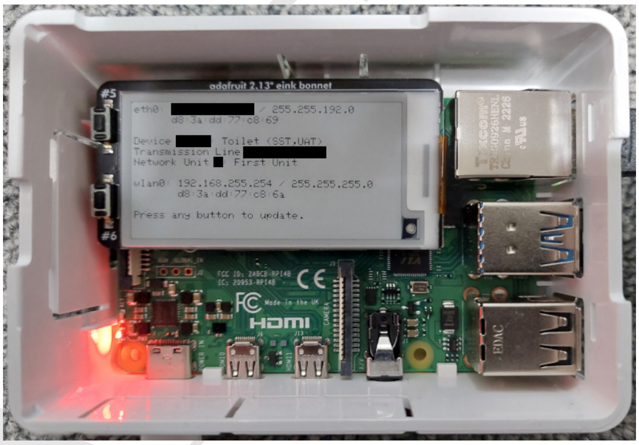
Whilst on the trains, a physical assessment of the installation was undertaken, with all available compartments opened and inspected. This showed how the components had been distributed, connected and physically secured, as well as their general appearance, and the types and numbers of network interfaces they offered. It also allowed potential connection points to be identified, into which an attacker could connect a malicious device. The manufacturer's maintenance engineers kindly facilitated this part of the information gathering process.

Care was taken to ensure that the integrity of the TCMS was not compromised during the research. All testing was passive or non-destructive, and did not involve attempted exploitation. To prevent the inadvertent introduction of malware, fresh installations of open-source operating systems were used, with signed software installed from trusted sources. The attack described in Section 5 required the injection of network packets from our own devices; no modifications were made to the configuration of any networked component installed on the trains or at the test-bed.

4.4. Tools and development

With a basic theoretical and practical understanding of the ETB gained from the documentation review and visits to the trains, we began to map the networks and analyse the traffic they carried. This gave detailed knowledge of the operation of the TCMS, specifically the protocols used, the datasets they contained, and how they were addressed with respect to their source and destination IP addresses and ports. The investigations were undertaken using two laptops, each running the Ubuntu Linux 20.04.6 LTS (Focal Fossa) operating system. A MikroTik RB960PGS hEX router was used to provide connectivity between EDs and their corresponding ETBNs, and to facilitate packet capture. Network mapping and traffic analysis was performed using tools such as `nmap`, `tcpdump`, `tcpreplay`, `tshark` and `wireshark`.

A number of python scripts were written to assist with the analysis of the TRDP, and to generate bespoke PDUs both in software and directly from the command-line. These were inspired by, and tested using, a private fork of the 'TCNopen' project that includes `trdp_spy`: a `wireshark` plugin for parsing TRDP packets[19]. It became clear at this stage of the research that the TRDP PDU format identified on the trains and at the test-bed differed from that defined in IEC 61375-2-3:2015, a point to which we shall return in Section 5.



```
eth0: [redacted] / 255.255.192.0
      d8:3a:dd:77:c8:69

Device [redacted] : Toilet (SST,UAT)
Transmission Line [redacted]
Network Unit [redacted] : First Unit

wlan0: 192.168.255.254 / 255.255.255.0
       d8:3a:dd:77:c8:6a

Press any button to update.
```

Figure 4: Raspberry Pi 4 Model B (4GB) used during this case study

A Raspberry Pi 4 Model B (4GB) was purchased and configured to assist with the research, and to undertake the attack derived from it. This device ran the Raspberry Pi Linux OS Lite (64-bit) release 2023-12-11 operating system, a port of Debian 12 Bookworm, and was fitted with an Adafruit 2.13-inch Monochrome E-Ink Bonnet to assist with debugging. It could be connected directly into an ETB CN, before identifying itself as a legitimate ED and being issued with an associated and valid IP address (this is shown in Figure 4, when the Raspberry Pi successfully identified itself as a universal access toilet). A wireless interface allowed the Raspberry Pi to be controlled remotely. This was useful as space on the trains was often limited; it would also be helpful to an attacker, who could fit it into a train compartment before retiring to control to it from a nearby seating area, or from the wayside if using a Virtual Private Network (VPN) with mobile data or passenger wifi acting as a bearer.

4.5. Test-bed environment

The final stage of the research involved the use of the manufacturer's test-bed, which provided a more flexible, comfortable and safer environment for experimentation than that found on the trains. The test-bed was fitted into three 42U racks and whilst it did not contain the rich variety of components found on

[19] Thorsten Schulz's 'TCNopen' GitHub repository: `https://github.com/T12z/TCNopen` (accessed 5th Mar 2024).

the trains (for example, there were no external doors, brake controllers or traction systems) it was representative. Like the train, it contained two redundant ETBs, had an operational TMC and monitoring server, a driver's desk with a display unit and fully-operational traction control that allowed the train's speed to be set; and an interface unit to which were connected a number of non-ethernet components, including train control circuits, and a real and simulated ETCS European Vital Computer (EVC).

Usefully and unlike on the trains, the test-bed ETBNs contained a number of operational and empty ports, which allowed laptops and the Raspberry Pi to be connected throughout the network. Because of its flexibility, the opportunity to manipulate the train's controls in a safe manner, coupled with the assistance of on-site electrical and electronics engineers, ensured that the majority of the research was undertaken at this location.

Whilst at the test-bed, a small number of specification documents were provided. These related to TCMS IP address allocations, TRDP protocol usage, and the operation of the ETBNs and a selection of EDs connected to the ETB. A TCMS engineer was also able to supply information about the TRDP PDU formats, and answer questions as they arose. However, the majority of the research and the knowledge resulting from it came from experimentation and reverse-engineering of the various protocols detected, with the results being confirmed through integration testing with genuine components on the real trains, or through discussions with engineers who had access to technical specification documents.

4.6. Limitations

It is important to state that this case study forms part of a cyber security assessment of the design and implementation of the TCMS, with a focus on the protocols it carried and the way in which EDs were controlled and monitored. It was not a penetration test concerned with the vulnerabilities of the individual components themselves.

During the research, we were not given access to all available design documentation, specifications and engineers. In addition, access to the train and test-bed were limited in terms of time: preparatory work was undertaken to maximise the usage of these resources, and whilst this case study reached several conclusions in the time available, extended access to the facilities would likely have resulted in additional or more detailed findings. As a result, suggestions for further work are given in Section 7.

As a result of these limitations, this case study may not be considered *exhaustive*; it is, however, representative of what a determined and capable attacker, such as a nation state, would be able to achieve given adequate time and technical knowledge of the target, as they would likely be able to obtain. With access to additional resources and greater time spent on the trains and/or at the test-bed, we believe it would be possible to develop more sophisticated and reliable attacks.

5. Results

5.1. Introduction

In this section, we will describe the ETB installed on the trains examined during this case study, with an almost exclusive focus on the TCMS. This will include the network architecture and methods by which it was used to control and monitor the train. We will make accompanying cyber security related comments as we progress, before describing an attack that was successfully launched against a key TCMS component. The purpose of this attack was to cause am operational incident and highlight a vulnerability faced by all modern ethernet trains that follow the applicable parts of the IEC 61375 standard series, some mitigations for which are offered in Section 6. The method by which the following information was obtained is described in Section 4, with a reminder that the majority of it was discovered through practical investigations and experimentation.

5.2. Network architecture

The trains examined during this case study comprised five cars, each of which was configured as a single consist. The design report revealed that the order of the three central cars could be changed; however, to operate with an optimal network architecture, all five must be present. This is because of the redundant duplication of EDs. For example, the security gateways connecting the TCMS to non-TCMS networks were installed in different cars. As these components had been supplied as a fail-over pair, the removal of either of these two cars would degrade the redundancy of the connections between the networks. Practically, therefore, the five-car configuration would not be modified; instead, a second train-set, built to an identical design, would be coupled to either end of the train should additional passenger capacity be required.

An inspection revealed that the two cabs were identical and contained duplicated TCMS components running as hot fail-over pairs. For example, each cab contained a TMC, one of which was active at any given time. They were also fitted with an interface unit that provided connectivity to non-ethernet-enabled components, such as external door controllers, and a display unit for conveying information to the driver. The failure of any one of these components in either cab would see the secondary automatically and immediately take-over to ensure a continued service. In order to provide this level of resilience, the TCMS used multicast messaging to ensure all components remained up-to-date with the current configuration and status of the train, with these messages flowing freely across the ETB.

Figure 5 shows a simplified example of the network architecture found in the first two cars. It was produced following the network mapping exercise on the trains and confirmed at the test-bed. As can be seen, the TCMS was formed of two ETBs: ETB 1 shown in red; and ETB 2 shown in blue. These two ETBs ran the full length of the train and were provided by two ETBNs in each car: one ETBN for each ETB. The IP subnet of ETB 1 was `10.128.0.0/11`, as is allocated to the TCMS in IEC 61375-2-5:2014; whilst that of ETB 2 was `10.160.0.0/11`, which according to the same standard is to

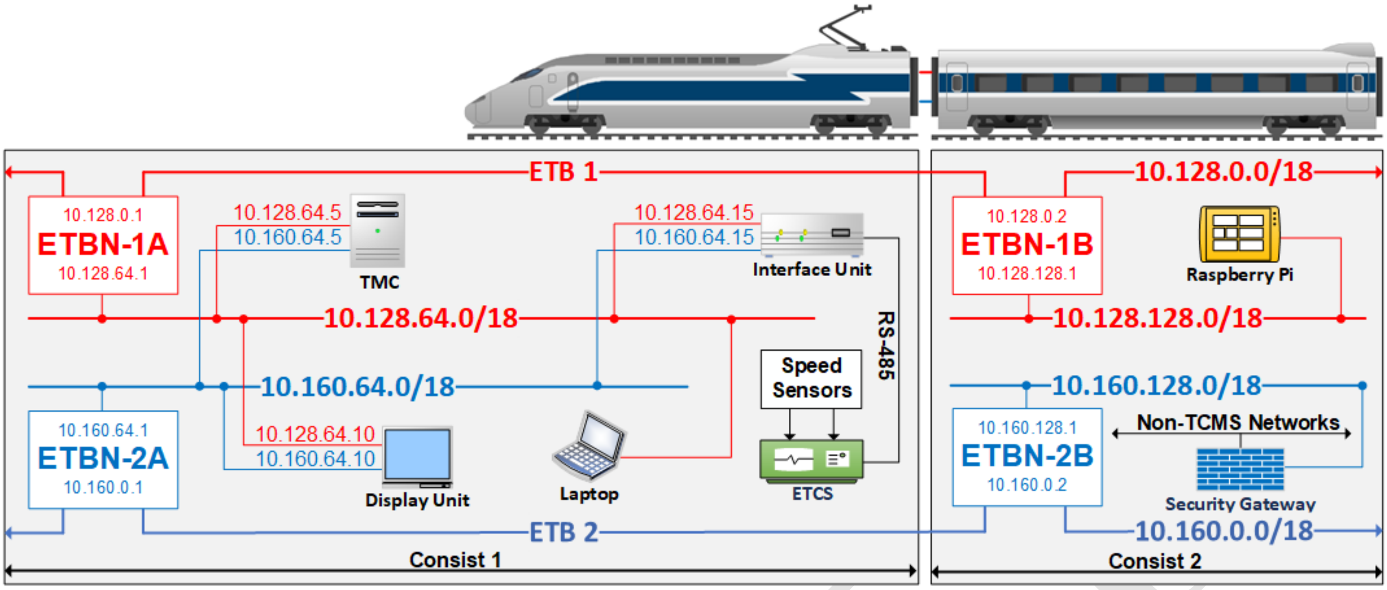


Figure 5: A simplified section of the Ethernet Train Backbone (ETB) installed on the train that was examined during this case study

be used for ‘multimedia’. With regard to the two /18 ETB backbone subnets (i.e. the subnets that contained the ETBN backbone interfaces), the ETBNs were numbered sequentially from .1 within their subnets, as suggested by the standard. For example, for ETB 2: ETBN-2A was in the first car and had been assigned the IP address 10.160.0.1, ETBN-2B was in the second and been assigned 10.160.0.2, and so on. The /18 CN provided by each ETBN sat within the wider /11 ETB subnet; they were incremented in contiguous /18 blocks following the first /18 subnet allocated to the backbone.

Each ETB was found to be fully routable: any ED could be reached from any point within its ETB instance. Experiments revealed that packet filtering did not appear to be performed by the ETBNs. Furthermore, the network did not contain any ECNs as defined in IEC 61375-3-4:2014, so there was no requirement to support R-NAT.

The architecture appeared to have been taken from one of the examples provided in IEC 61375-2-5:2014, where redundancy of the TCMS is achieved through the duplication of ETBs, rather than by placing redundant pairs of ETBNs in a single ETB and implementing redundant ECN switch topologies. The advantages of this architecture are: high availability; the need for a smaller number of network components to provide the desirable levels of redundancy when compared to other configuration options detailed in the standard; and its relative simplicity to manage and maintain. However, its open nature means that, once a foot-hold in the ETB has been established, a compromised or malicious component has connectivity to all other components. Hopping from one ETB to the other would require the prior compromise of a dual-homed ED, the most likely targets being those running embedded versions of Microsoft Windows (the remaining components did not offer any open TCP or UDP ports; nor was it possible to route through any of the dual-homed components).

5.3. IEC 62443 approach

In terms of the IEC 62443 international standard series, which the train manufacturer used as the basis for the design, the TCN was subjected to a two-part EBIOS[20] risk assessment process, the first of which was performed by an independent third-party, and the second by the internal cyber security team working in partnership with the eventual train owner and operator. The latter took place during a number of workshop sessions and included a wide range of contributions from experts skilled in the areas of: cyber security; railway operations; rolling-stock design, engineering and maintenance; and fleet management. This resulted in a carefully-considered architecture partitioned according to the safety requirements of the components it contained.

ETB 1 is the first and lowest-level zone containing safety-related EDs that have the highest requirements with respect to their security. ETB 2, the second zone, sits in an identical security level as it contains the same safety-related EDs, as well as those defined as being non-safety-related. These two zones are connected by dual-homed EDs, although this conduit is not intended to be traversed during the legitimate operation of the train.

The security gateways represent a control on the conduit from these first two safety-related zones to a number of non-TCMS networks that sit within several higher-level non-safety-related zones. These gateways were configured as firewalls, with rules allowing traffic from specific source and destination IP addresses, ports and protocols to pass, whilst denying all other access attempts. Within these parameters, the gateways

[20] Widely used in both the public and private sectors, EBIOS, or the ‘Expression des Besoins et Identification des Objectifs de Sécurité’, is a risk analysis methodology comprising five phases that provides risk owners with a thorough and high-level approach to risk management. It was originally developed by the French government and is now maintained by a group of experts.

only forward TRDP PDUs in the format used in this TCMS implementation, namely those defined in IEC 61375-2-3:2012 CD. They also run an IDS on their ETB 2 and external non-TCMS interfaces to detect malicious activity, which when detected is submitted as an alert to the train operator's Security Event and Incident Management (SEIM) system.

5.4. ETBNs

The ETBNs were designed and built by the train manufacturer. They each contained two ETB backbone and ten CN interfaces within the same physical unit[21]. This meant that additional CN switches were not required; however, in the case of the ETBNs fitted in the two cabs, which provided connectivity to a larger number of EDs than those in the intermediate cars, additional connection points were provided by secondary switching units. Both ETBs ran at a speed of 100Mbps full-duplex as is required by IEC 61375-2-5:2014[22].

5.5. EDs

EDs were connected to either one or both of the ETBs[23]. The design aspiration for this network, or rather of subsequent implementations based on it, was that ETB 1 would be facilitate *control*, with the duplication of control messaging on ETB 2, which acted as its hot fail-over; ETB 2 was to support *monitoring*, until such a time that ETB 1 failed when it was then also used for control. In practical terms, as this was a First Generation ETB as defined in Table 1, the ETB did not facilitate any aspect of train control: both ETBs were used to monitor EDs, which were controlled over secondary interfaces such as the train control circuit. Therefore, and to sit within this architecture, dual-homed and monitored safety-related EDs were connected to both ETBs; whilst all remaining single-homed and non-safety-related monitored EDs were connected to only ETB 2. An example of this can be seen in Figure 5, which shows the safety-related and dual-homed TMC connected to both ETBs; whilst the non-safety-related security gateway was only connected to ETB 2.

5.6. Protocols

During this case study, a total of 6,525,948 packets were captured from the ETBs on the trains and at the test-bed. An analysis of these packets revealed that, after excluding TTDP and layer-2 protocols, all were TRDP UDP Pd PDUs; no TCP or UDP Md PDUs were detected. The reason for this is that the Pd PDUs carried periodic data between EDs sent at either 20ms *short-period* or 200ms *long-period* intervals, behaviour that formed the basis of this TCMS implementation. A technical specification document revealed that the Md PDUs were intended to carry larger quantities of sporadic non-safety-related data, such as that pertaining to error reporting and the exchange information with the wayside. As no errors were detected during the packet capture, and as the external communications links were inactive, these PDUs were therefore unlikely to be seen.

The initial dissection of the captured TRDP PDUs gave unexpected results: the value of the `DatasetLength` header, which defined the length of the `Dataset` in bytes (excluding up to three bytes of padding), did not match to the actual length the extracted `Dataset`; nor could the `HeaderFCS` be verified. This was eventually resolved after an integration engineer stated that a non-standard PDU format was used in this particular TCMS implementation. It was found to be based on a 2012 draft version of the eventual IEC 61375-2-3:2015, in which the PDUs differ from those in the final release. The reason for this was that the installed components were designed and built before the final publication of the standard. The subtle differences between these versions are shown in Figure 3, and whilst not deliberately security-by-obscurity, the unexpected format did hamper the development of the attack described in Section 5.9. It demonstrates that, even with standardisation, additional work may be required to customise an attack targeted at a particular train or TCMS implementation.

With knowledge of the TRDP format, it was possible to parse the captured packets to gain a greater understanding of the TCMS. Analysis revealed that, in most cases, Pd PDU status requests were sent from the TMC to individual EDs or groups of EDs[24]. Such a Pd *request*, known as a 'Pr' PDU, was identified by the TRDP `MsgType` header shown in Figure 3, which contained the two byte hexadecimal value `5072` as defined in IEC 61375-2-3:2015. EDs within these groups replied individually, formatting their responses into a Pd 'Pd' *data* reply PDU, which was identified by the value `5064` in the same header. The *streams* of these Pr-Pd pairs were not as are often found in more traditional IP networks: they did not pass directly from a specific source to destination host, and it was not obvious how the requests and replies were associated. This is described further in Section 5.7.

Together with giving instructions, the 'Pr' PDU conveying the status requests from the TMC also provided information to the groups of EDs. This information, known as *common data*, detailed the current status of the train and was therefore a potential target of attack. Figure 6 shows the format of the `Dataset` within these PDUs[25].

The leading common data within this `Dataset` comprised a two byte `DeviceId` (`FF` to define it as common data), a two byte and self-explanatory `DataLength`, and the trailing `Data`. One or more ED-specific instruction sets followed, with

[21] On each ETBN, the ETB backbone interfaces were provided by two female 4-pin M12 sockets, with an additional socket used for maintenance purposes and to which could be connected a maintenance laptop (it is by this method that the TCMS components were updated). The two CN interfaces were each provided by a male 5-port Souriau-Sunbank MSG 5 'Quadrax' receptacle (4 pins per port) which when combined supported connectivity for up to 10 EDs per ETBN.

[22] For reference, the non-TCMS networks ran at 1Gbps full-duplex, and a physical inspection revealed that all the components in those networks were connected by 8-pin M12 sockets that supported this increased speed.

[23] The connection from an ETBN to an ED was terminated at a male four-pin Souriau-Sunbank SMS receptacle on each ED.

[24] The EDs appeared to be grouped with respect to the functions they performed, and presumably to increase the efficiency of TRDP by reducing the number of packets on the ETB.

[25] When addressed to individual EDs, the common data was simply omitted.

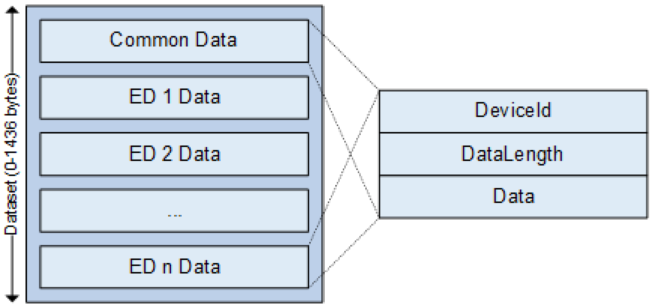


Figure 6: Contents of the `Dataset` present in TRDP Pd ‘Pr‘ *request* PDUs

each being denoted by its particular `DeviceId`.

Research at the test-bed revealed some of the contents of the `Data` field of the common data. For example, the manual adjustment of the train's speed caused bytes 41 and 42 of the data to change, and it became obvious that this was the method by which the train's speed was advertised to EDs[26]. Other bytes could also be identified, such as those denoting the current date and time.

The `DeviceId` for a particular ED was pre-defined, and each ED knew its own `DeviceId`. To determine how these had been assigned, the TMC was disconnected from the ETB and re-connected to the MikroTik router, to which was also connected a laptop and the up-link to the local ETBN. `dhcpdump`, a DHCP packet dumping tool running on the laptop, revealed that the TMC, on re-connection, submitted a DHCP option containing a sequence of bytes in its `DHCPDISCOVER` and `DHCPREQUEST` messages[27]. The corresponding `DHCPOFFER` contained the assigned IP address, and further tests revealed that the fourth quad of this IP address was the decimal representation of the hexadecimal `DeviceId` (in the cases when an ED was ethernet-enabled and had an IP address). For example, the `DeviceId` of the TMC was 5, and it was therefore always assigned an IP address ending in `.5` (`10.128.64.5` in the first CN of ETB 1, as shown in Figure 5). Using this method, it was possible to ascertain the IP address and `DeviceId` of each ethernet-enabled ED, and for the laptop to identify as any such component by providing the required DHCP option and byte sequence[28]; to save time, those of non-ethernet-enabled EDs connected by means of an interface unit were obtained from a technical specification document. The `DeviceId` values were useful as they allowed attacks to be targeted at specific EDs, even when they were not directly connected to the ETB; similarly, the common data allowed incorrect train status information to be injected into the ETB and sent to specific groups of EDs.

Multicast Subnet	**ETB**	**Pd/Md**	**PDU Purpose**
`239.10.0.0/16`	1	Pd	Safety-related
`239.20.0.0/16`	2	Pd	Safety-related
`239.30.0.0/16`	2	Pd/Md[30]	Non-safety-related

Table 2: Identified multicast addresses, their locations, PDU types and purposes

5.7. Addressing

Multicast addressing was the way by which EDs communicated, and this was the only form of addressing detected during this case study[29]. The multicast addresses identified on the trains and at the test-bed are shown in Table 2, together with the ETBs in which they were detected, and the type and purpose of their associated PDUs. As an example, when submitting a TRDP Pd 'Pr' *request* PDU to a group of safety-related components situated in ETB 1, the TMC sent the packet to a multicast address in the `239.10.0.0/16` subnet (the exact address used within this subnet, together with its destination port, was determined by the group in which the destination EDs resided, the period of the communication and format of the `Dataset`). For resilience, the same packet was also sent to a corresponding multicast address in `239.20.0.0/16` subnet in ETB 2.

EDs listened for packets addressed to the multicast addresses with which they or their groups were associated. Should a TRDP PDU arrive at an address assigned to a group, each ED in the group would extract the common data, followed by the `DeviceId` of each ED referenced in the `Dataset`. If an extracted `DeviceId` matched that of the ED, the relevant `Data` was processed; if not, the PDU was discarded[31]. When a PDU was addressed to a specific component and not a group, it did not contain any common data, and the receiving ED would therefore proceed to processing the `Data`.

To explain this behaviour in more detail, and as a means by which to introduce the attack described in Section 5.9, the communications between the TMC and ETCS unit, which was connected to both ETBs 1 and 2 by an interface unit as shown in Figure 5, was as follows.

1. The TMC with the IP address `10.128.64.5` crafted a TRDP Pd 'Pr' *request* PDU requesting the status of the ETCS unit. This was addressed to the destination IP address `239.10.3.10` (assigned to the ETCS unit) and sent on ETB 1[32]. The TRDP `SequenceCounter` in this

[26] The train speed was also advertised by means of the train control circuit, which allowed EDs to determine whether the train was stationary or travelling within particular speed boundaries, such as under 5 Miles Per Hour (MPH), under 25 MPH, and so on.

[27] To preserve the confidentiality of the train manufacturer, the submitted DHCP option and its associated byte sequence have been deliberately omitted from this report.

[28] Interestingly, this activity did not cause an error to be displayed on the TCMS console; it was possible for the laptop to identify as any ED, such as a traction unit, lavatory or catering system, even if these were duplicated in any given CN.

[29] The technical specification documents showed that unicast addressing would be used in the case of non-safety-related TRDP Md PDUs, as it was by this method that large quantities of data were exchanged. These exchanges always take place between specific EDs; therefore, there is no requirement for them to be shared across the ETB.

[30] As mentioned previously, no TRDP Md PDUs were detected during this case study. A technical specification document indicated that the destination addresses of Md PDUs would be found within this subnet.

[31] In the case of an interface unit, the interface unit itself performed this operation, before communicating with the EDs to which it was also connected, if the `DeviceId` was relevant.

[32] The request was also sent to the destination IP address `239.20.3.10` on ETB 2, but this redundant behaviour has been omitted for brevity.

PDU was set by the TMC; for the sake of this example it was 10000.

2. The interface unit with the IP address 10.128.64.15 was configured to process PDUs on behalf of the ETCS unit (the DeviceId of the ETCS unit was 6). In doing so, it received and dissected PDUs destined for the 239.10.3.10 multicast address. On receipt of such a PDU and as no common data was present (the ETCS unit was not part of a group and therefore the PDU did not contain the FF common data identifier), the Data was extracted revealing the intended destination: DeviceId 6, the ETCS unit.
3. The interface unit crafted a TRDP Pd 'Pd' *data* reply PDU to report the status of the ETCS unit, which included the train's speed. This was addressed to the destination IP address 239.10.13.1 (assigned to the reply) and sent on ETB 1. The TRDP SequenceCounter in this PDU was set by the interface unit; again, for the sake of this example, it was 20000.
4. The TMC listened for ETCS unit 'Pd' replies destined for the 239.10.13.1 multicast address. On receipt of the PDU, the TRDP SequenceCounter was compared to that in the previous reply (19999), and as the value was greater, the PDU was accepted. The data, which included the train's speed, was then extracted[33].
5. The TMC updated its variable holding the ETCS train speed. This was subsequently written into the common data of other TRDP Pd 'Pr' PDUs, and sent to the display unit with the IP address 10.128.64.10 so the speed could be seen by the driver.

This process was repeated every 200ms, with the TRDP SequenceCounter incremented by one in each PDU sent. As can be seen, to ascertain the timeliness and therefore the validity of the PDU, the receiving entity simply checked that the SequenceCounter was greater than that in the previous reply. The 'Pr' and 'Pd' PDUs were therefore independent of one another, and as the contents of the 'Pr' were not included in the 'Pd', there was no need for the ETCS unit to wait for a request before submitting its reply. Add to this a lack of cryptographic authenticity and integrity verification, and an attack could be developed to cause the TMC to accept a fake train speed.

Before moving on to describe this attack, there are two additional points that are relevant to the operation of the TCMS. First, and as a useful security feature, all the multicast packets were visible to a dual-homed monitoring server that was connected to both ETBs. This server was able to generate detailed logs describing the status of the train at any given time. Second, the open architecture allowed spoofed PDUs to be injected into an ETB from any CN connected to it, or from the backbone itself. For example and with reference to Figure 5, the Raspberry Pi connected to the 10.128.128.0/18 CN in the second car and configured without an IP address injected a TRDP Pd PDU containing the '68 65 6C 6C 6F 57 6F 72 6C 64' byte sequence in its Dataset. The source of this PDU was set to 10.128.64.5, the IP address of the TMC in the cab, and the destination to 239.10.1.10, a valid multicast address associated with the TMC and previously observed on ETB 1. The laptop, connected to the 10.128.64.0/18 CN in the cab and being assigned the IP address of a brake controller unit after submitting a valid DeviceId in its DHCP request, successfully received this PDU and was able to decode the byte sequence contained in the Dataset: 'helloWorld'. It was evident that there was no packet filtering, nor did the ETBNs prevent the obviously spoofed packet from passing between the components. No errors or warnings were displayed on the TCMS console, and as this process was performed on ETB 1, it was not visible to the IDS running on the security gateway.

5.8. Physical security

Having explained the operation of the TCMS, we will now describe the physical security controls applied to its constituent components. This is to give an indication of the difficulty of connecting a malicious device, such as a Raspberry Pi, to the ETB. The attack described in Section 5.9 assumed this level of physical access to ETB 1; however, it would also be successful following the prior compromise of an ED, or network component such as an ETBN.

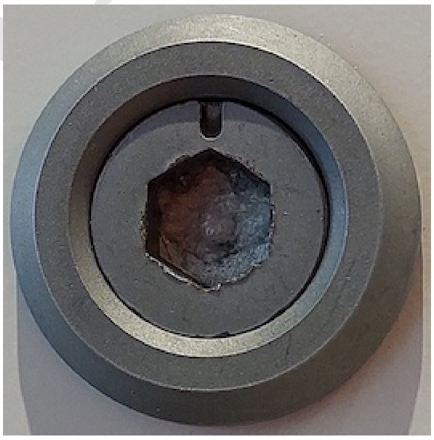


Figure 7: Hexagonal recess lock securing TCMS network components

Components forming the ETB were distributed throughout the train as dictated by the availability of storage space and the need for redundancy. TCMS components were housed in cabinets on one side of the train; those pertaining to non-TCMS networks, such as the PIS, were housed in separate cabinets, usually on the opposite side. The TCMS cabinets were secured by two 8mm hexagonal recess locks as shown in Figure 7, whilst those containing non-TCMS components were secured by two square recess locks of the same or a similar size—all of which could be opened using widely available tools. There were no visible UK power sockets in any of the cabinets.

With regard to the TCMS, the ETB 1 and 2 ETBNs, with their M12 sockets and MSG 5 receptacles, were readily accessible from the passenger sections of each car, once the relevant

[33] Other checks were made, such as the validity of the FCS, before the PDU was accepted.

compartment had been identified and opened. Other EDs, such as external door controllers, were similarly accessible, but their SMS connectors were often concealed behind steel plates. Key TCMS components, such as the TMCs, monitoring servers and ETCS unit, had been installed in the two cabs and were therefore inaccessible to passengers.

Access to the ETB, and therefore the TCMS, could be gained in a number of ways, such as through an ETBN's M12 maintenance port, or by inserting a switch between an ETBN and ED to provide additional connection points.[34]. A number of suitable locations in which an attacker could insert a malicious device were identified. One of these, situated towards one end of the train, was unlikely to be visited by passengers, and would be unused and unoccupied at certain times of the day. The locations were covered by CCTV and suspicious behaviour may therefore be detected by the train crew; however, with familiarity of the components, the installation of a malicious device, such as a battery-powered Raspberry Pi, would take less than one minute.

The conclusion of the physical security review was that, notwithstanding the CCTV and train crew, an attacker is only ever two hexagonal recess locks, one M12 socket and a static IP address away from the TCMS[35].

5.9. Developing an attack

5.9.1. Introduction

In this section we shall describe an attack launched against a TCMS safety-related component. It is important to repeat that as the trains examined during this case study were fitted with a First Generation ETB, the purpose of which was to *monitor* rather than *control* the TCMS, it was not possible to command any of its constituent components.

At first glance, this inability to control EDs may produce a false sense of security—and safety. However, the ability to monitor could, at best, cause confusion that might result in the driver stopping the train; at worst, it may even lead a operational incident. To highlight this and as an example, it would not possible for an attacker to open the external doors following the compromise of a First Generation ETB; however, it would be possible for the same attacker to inform the driver that the doors had been opened when the train was travelling at speed. This may see the train being returned to the depot for diagnostics and maintenance, especially if performed intermittently. Following the same example, the attacker could also report that doors were closed when in fact they had been wedged open. This would be a more serious breach of safety that may result in a passenger falling from the train.

5.9.2. Target identification

As it was not safe to interfere with the operation of a train in live traffic, the test-bed was used to develop the attack. The most accessible and suitable component selected as the target was the ETCS unit, to which we recently referred in Section 5.7.

As previously mentioned, ETCS is the signalling and train control component of ERTMS, providing train supervision and Automatic Train Protection (ATP). It allows a train to determine its location and monitor its speed, and for automatic interventions to take place to prevent incidents. The system was developed in Europe, and adopted by the EU to encourage competition and improve railway interoperability between member states. Although referred to as a 'unit' in this paper, it is modular in nature and can be customised according to the environment in which it is operates. Typical components include an EVC, Driver Machine Interface (DMI) and GSM-R data radio. A useful description of ETCS is given by the International Union of Railways (UIC) in [44], with the RDG providing a UK-specific focus in [45].

With respect to the trains examined during this case study, the ETCS units were installed in each cab and used to determine their speed. Inputs providing this data typically include axle tachometers and a Doppler radar focused on one of the tracks. As the test-bed was stationary and did not contain either of these measurement devices, the inputs were simulated by a Raspberry Pi Pico, supplied and programmed by the train manufacturer. The result was a fully-functioning and simulated ETCS implementation: as the driver moved the throttle to accelerate the train, the Pico generated the required raw speed inputs and submitted its status to an interface unit over an RS-485 connection. The interface unit was then able to advertise the train's speed on the ETB. Figure 5 shows how the ETCS unit was connected; the example in Section 5.7 describes how the train's speed was requested and reported.

5.9.3. PDU identification

As mentioned in Section 5.6, it was possible to ascertain the IP address and `DeviceId` of an ethernet-enabled ED by observing its interactions with its local DHCP server[36]. Using this technique, we were able to identify all the TRDP Pd 'Pd' *data* reply PDUs leaving the interface unit. These *candidate* PDUs were then inspected to ascertain whether they contained the current train speed at bytes 104 and 105: if they did, they were associated with the ETCS unit and therefore the ones we intended to spoof. Testing revealed that the PDUs containing the train's speed originated from the IP address `10.128.64.15` (that of the interface unit) and were sent to the `239.10.13.1` multicast address. Further analysis over a longer time period allowed us to conclude that *all* the PDUs sent to this particular multicast address contained the output from the ETCS unit, a fact that enabled the train speed to be quickly identified and spoofed from any location on the ETB.

[34] It is important to state that the manufacturer intended to disabled all unused network interfaces before the final delivery of the trains, with the exception of the ETBN maintenance ports; however, the trains were undergoing maintenance and testing during this case study, and may therefore have been in a less secure configuration than would normally be the case.

[35] A static IP address is required when connecting to an ETBN's management port. DHCP would provide an IP address when connecting to a CN, as would occur by the insertion of a switch between an ETBN and ED, although a valid `DeviceId` would be required.

[36] It was also possible to determine the IP address and `DeviceId` by disconnecting the ED from the ETB, connecting it to a laptop running the `tcpdump` or `wireshark` tool and observing the captured packets.

5.9.4. *Packet spoofing*

The following attack was undertaken using the findings described above. The purpose of the attack was to cause a fake train speed to be shown on the display unit situated in each cab, in this example by reducing the actual speed by ten percent. Such malicious behaviour may cause the driver to over-speed, a fact that could be significant if travelling in a speed-restricted track section, over points or around bends. The attack took advantage of the fact that the TRDP Pd 'Pr' *request* and 'Pd' *data* reply PDUs were not closely associated: they were sent as independent *steams* such that the reply was not dependent on the request. This greatly simplified the process as there was no need to inspect a request before injecting a spoofed reply; one could simply inject a new reply using the previous and legitimate one as an up-to-date template[37]. The method of the attack was as follows, with the EDs connected as per Figure 5 and the sequence of spoofed PDUs as shown in Figure 8.

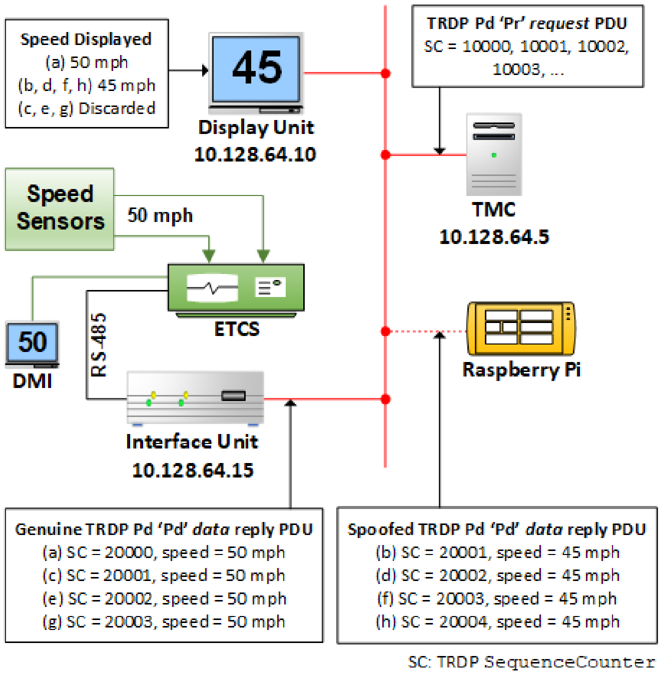

Speed Displayed
(a) 50 mph
(b, d, f, h) 45 mph
(c, e, g) Discarded
45
Display Unit
10.128.64.10
TRDP Pd 'Pr' request PDU
SC = 10000, 10001, 10002, 10003, ...
TMC
10.128.64.5
Speed Sensors
50 mph
ETCS
50
DMI
RS-485
Raspberry Pi
Interface Unit
10.128.64.15
Genuine TRDP Pd 'Pd' data reply PDU
(a) SC = 20000, speed = 50 mph
(c) SC = 20001, speed = 50 mph
(e) SC = 20002, speed = 50 mph
(g) SC = 20003, speed = 50 mph
Spoofed TRDP Pd 'Pd' data reply PDU
(b) SC = 20001, speed = 45 mph
(d) SC = 20002, speed = 45 mph
(f) SC = 20003, speed = 45 mph
(h) SC = 20004, speed = 45 mph
SC: TRDP SequenceCounter

Figure 8: Attack to spoof the train speed as determined by the ETCS unit

1. Using the throttle, the train speed was set to 50 MPH, which was correctly shown on the display unit.
2. The Raspberry Pi, situated in the second car, was configured to capture TRDP Pd 'Pd' *data* reply PDUs destined for the `239.10.13.1` multicast address. As a reminder, PDUs sent to this address contained the current train speed calculated by the ETCS unit and transmitted by the interface unit every 200ms.
3. On receipt of the first of these 'Pd' PDUs (a), both the TMC and Raspberry Pi extracted the TRDP headers and `Dataset`. In this example, the `SequenceCounter` had been set to `20000`; the train speed stored in bytes 104 and 105 of the `Dataset` had been set to `500` in decimal, which represented 50.0 MPH.
4. As the `SqeuenceCounter` was one more than that in the previously received PDU, the TMC accepted the PDU, maintained the train speed at 50 MPH, and continued to show that speed on the display unit.
5. The Raspberry Pi constructed a new TRDP Pd 'Pd' *data* reply PDU (b), incrementing the `SequenceCounter` by one (`20001`) and reducing the train speed by ten percent (setting it to `450` in decimal). A new `FCS` was calculated and the fake PDU injected into the ETB. All other PDU data, including its source and destination IP addresses and ports, remained unchanged.
6. The fake PDU was received by the TMC. As the `SequenceCounter` was one greater than that of the previously received PDU, it was accepted and processed. The train speed was set to 45 MPH and displayed accordingly.
7. The genuine PDU was then received by the TMC. As the `SequenceCounter` was equal to that of the previously received fake PDU, it was discarded as a duplicate. The train speed remained at 45 MPH.
8. On receipt of the genuine PDU described in the step above, the Raspberry Pi repeated its spoofing process: incrementing the `SequenceCounter` by one; overwriting bytes 104 and 105 with the fake train speed; and injecting the PDU.

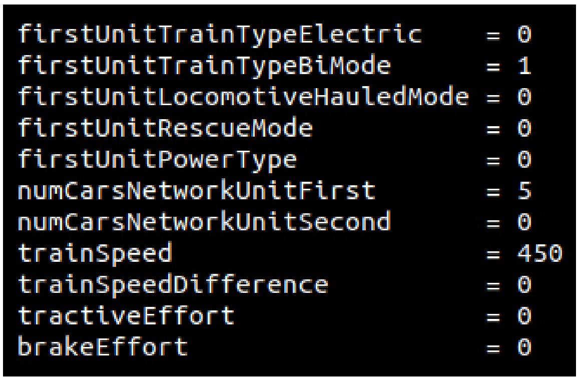
This process of 'racing the SequenceCounter', when repeated in a loop (c-d, e-f, g-h), ensured that the fake train speed, being the expected TRDP Pd 'Pd' *data* reply PDU containing the next valid `SequenceCounter`, was always accepted by the TMC. Conversely, each genuine PDU, containing a `SequenceCounter` that had already been seen, was regarded as a duplicate and dropped by the TMC. The result was that the fake train speed was shown on the display unit for the duration of the attack; when the spoofing was paused, the genuine PDUs were again accepted and the speed returned to its correct value.

Interestingly, the the fake train speed was carried beyond the display unit. The TMC, accepting it as genuine, included it in the common data sent to groups of EDs, as described in Section 5.6. This meant that, even if the driver questioned the legitimacy of the speed, it was potentially and at the same time being used by the TMC and EDs in any speed-related calculations. In a second example, an easily identifiable fake train speed was injected by the Raspberry Pi using the 'racing the SequenceCounter' technique. A laptop situated in the ETB 1 CN in the fourth car extracted the common data from a captured TRDP Pd 'Pr' *request* PDU, sent by the TMC to the `239.10.1.20` multicast address following its acceptance of the spoofed PDUs[38].

[37] Consequently, the fake PDU is only ever one PDU *behind* the genuine one. This ensures that the genuine data carried in the fake PDU is timely, being only one cycle (20ms or 200ms, depending on the period of the communication) *late*.

[38] The `239.10.1.20` multicast address referred to a group containing the auxiliary power supply and traction unit.

The output of this common data, extracted and parsed according to the manufacturer's specification, is shown in Figure 9, with the fake train speed being clearly visible. This example caused an "OVER-SPEED!" warning to be flashed on the display unit.



```
firstUnitTrainTypeElectric     = 0
firstUnitTrainTypeBiMode       = 1
firstUnitLocomotiveHauledMode  = 0
firstUnitRescueMode            = 0
firstUnitPowerType             = 0
numCarsNetworkUnitFirst        = 5
numCarsNetworkUnitSecond       = 0
trainSpeed                     = 450
trainSpeedDifference           = 0
tractiveEffort                 = 0
brakeEffort                    = 0
```

Figure 9: Fake train speed carried in TRDP Pd 'Pr' *request* PDU common data

5.9.5. Duel speed displays

Figure 8 shows the situation of the ETCS DMI and how it was connected within the TCMS. The DMI, which was present on the trains but not at the test-bed, showed the train speed as reported by the ETCS unit through a direct wired connection, rather than over the ETB. Consequently, if a train was subjected to this attack, the DMI would always show the genuine speed, whilst the display unit, which was updated over the ETB, would show the fake speed. In this situation and with prior knowledge or training, the driver may opt to trust the ETCS speed over the TCMS speed, although they would not know which was correct, or why there was a difference. The likely outcome is that the train would proceed slowly to the next station before diagnostics were performed.

5.9.6. Counter and checksum

The attack proceeded as described in Section 5.9.4. However, the displaying of the fake train speed was unstable: it was occasionally replaced with the genuine speed for a short period of time, usually less than one second. This led to a further analysis of the genuine TRDP Pd 'Pd' *data* reply PDUs in order to identify any *changing* bytes in addition to those holding the speed (bytes 104 and 105). The result was as follows.

- Byte 66 was found to be a simple counter, incrementing by one in a continuous loop within the 0-255 range.
- Bytes 152 and 153 also changed, either individually or as pair. Testing revealed that these were checksum bytes taken over a portion of the `Dataset`, with each holding the cumulative XOR output value of the preceding even and odd bytes respectively.

The Python code on the Raspberry Pi was updated to ensure these three bytes were correctly set in every spoofed PDU. However, this did not improve the stability of display. A later discussion with a TCMS engineer revealed that the bytes were likely to be intended for the interface unit, which would perform the validation on receipt of an ETCS status update received on its RS-485 interface. On receiving a valid update, the interface unit therefore appeared to simply copy the ETCS data verbatim into the PDU; the validity of the `Dataset` was not confirmed by the TMC, as the check was assumed to have already taken place. The consequence of this was that a modification made to a PDU after its transmission by the interface unit would not be detected by the TMC, even if the `Dataset` was invalid[39].

5.9.7. Primary and secondary ETBs

In Section 5.5 we mentioned that, as a design aspiration, ETB 2 was to act as a hot fail-over for ETB 1. We were therefore unsure how the dual-homed TMC would behave when receiving conflicting data—specifically the train speed—on each of its two ETB interfaces.

When performing the attack, the spoofed TRDP Pd 'Pd' *data* reply PDUs were injected by the Raspberry Pi on ETB 1, sent to the `239.10.13.1` multicast address and received by the TMC on its ETB 1 interface. However, the interface unit whose replies we were spoofing was also submitting corresponding PDUs on its ETB 2 interface: they were sent to the `239.20.13.1` multicast address and also received by the TMC. This meant that, for these PDUs, the `Dataset` received by the TMC was different on each of its two interfaces: those received on ETB 1 contained the fake train speed; whilst those on ETB 2 contained the genuine speed.

Perhaps unsurprisingly, the TMC's ETB 1 interface was found to take priority: if the two `Datasets` differed, those received on ETB were accepted; whilst those on ETB 2 were discarded. Therefore, the attack could only be performed on ETB 1. This, however, did not constitute a problem, as both ETBs were equally accessible on the train and in test-bed; and advantageously, the security gateway, on which was running an IDS, was connected to ETB 2, meaning it was unable to collect the spoofed packets and detect the attack.

5.9.8. Display unit updates

In the time available, it was not possible to determine exactly how the display unit obtained the train speed. It may have read the bytes directly from the TRDP Pd 'Pd' *data* reply PDUs normally submitted by the interface unit, and by the Raspberry Pi during the attack. Alternatively, it may have extracted the speed from the common data published in the 'Pr' *request* PDUs addressed to the groups of which it was a member, or from some other PDUs. It may even have received a personal update from by the TMC. It would be useful to determine how this was achieved, as it may then be possible to use the attack to simply update the display unit in isolation, rather than to target the multicast replies destined for the TMC and in doing so *poisoning* all EDs with a fake train speed in the common data they consume. This will be an area of further research.

[39] The unstable display was caused by buffering on the Raspberry Pi. This periodically delayed the transmission of fake PDUs, which allowed the genuine PDUs to be accepted by the TMC. The unwanted behaviour was subsequently corrected by setting an asynchronous packet processing option in the Python code.

5.10. Summary

In Section 5, we described the architecture of the ETB installed on the trains examined during this case study, before moving on to document an attack targeted against one particular safety-related TCMS component: the ETCS unit. This attack is but one example of many: the technique of 'racing the SequenceCounter' can be used to spoof TRDP Pd 'Pr' *and* 'Pd' PDU with the objective of causing confusion, disruption or an operational incident. This is true of a First Generation ETB configured for monitoring purposes only and against which the attack was launched; the consequences of targeting a Second or Third Generation ETB, which support both monitoring *and* control, would be much more serious.

The attack assumed access to ETB 1. This is readily achievable physically but perhaps more difficult logically, given the zoned network architecture, deployment of a security gateway between the zones, and the attention given to cyber security by the train manufacturer. Despite this, we must attempt to mitigate the vulnerabilities identified and in doing so "reduce the impact of compromise", as recommended by NCSC and mentioned Section 1.2.2. If the ETB is compromised at some future date, and we must assume that it will be, steps should be taken now to limit any secondary actions that might affect passenger safety.

In the next section of this paper, we will make recommendations to mitigate the identified vulnerabilities, the main one being the inclusion of a cryptographic authenticity and integrity verification mechanism in the TRDP PDUs.

6. Discussion

6.1. Introduction

In this section, we will consider a number of measures that will mitigate the vulnerabilities identified in Section 5, in particular those that led to the development of the 'racing the SqeuenceCounter' attack. In doing so, we will describe how a small number of modifications would significantly improve the security of the TCMS, noting that the single major vulnerability was a consequence of the train manufacturer's adherence to an international standard, rather than a result of its own design decisions. We will also note the quality of the TCMS implementation, acknowledging the considered steps the manufacturer took to understand the threats it faced, as it perceived them at the time.

6.2. HMAC authenticity and integrity verification

The vulnerability exploited by the 'racing the SequenceCounter' attack described in Section 5.9 exists because an ED was unable to ascertain whether a TRDP PDU it received was sent by a legitimate ED or malicious third party. This calls into question the security of the entire TCMS implementation, although it is important to state that it was the result of specifications laid down in IEC 61375-2-3:2012 CD, rather than a design decision taken by the train manufacturer. Furthermore, this same issue potentially affects *all* trains built in accordance with this standard, including the IEC 61375-2-3:2015 version, unless a custom authenticity and integrity verification mechanism had been added to the PDU by the train or TCMS components manufacturer. Thankfully, a relatively straightforward solution exists: the addition of an Hash-Based Message Authentication Code (HMAC) to each PDU; however, the intricacies of such a modification will require careful consideration, as it necessitates short-term software and procedural changes that will have a long-term impact on the management of the fleet.

An HMAC is a keyed Message Authentication Code (MAC) that allows a component to establish whether a message it received originated from a legitimate sender, and that it had not been modified in transit (either maliciously or in error). Practically, this can be achieved using a cryptographic hash function and shared symmetric secret key, the latter being known only to the legitimate sender and receiver. In simple terms and in such a system, a sender would concatenate a key to a message, which would then be hashed; following this, the key would be removed and the hash concatenated to the message before it was sent. On receipt, the receiver would remove the concatenated hash and perform the same operation, comparing the hash results to establish the message's authenticity and integrity: identical hashes indicate that the message originated from a legitimate sender and was received as sent; different hashes suggest that one or both of these statements is false. The hash function selected to perform this check must provide a sufficient level of 'collision resistance': it must be unfeasible to create any two messages that yield an identical hash, thus denying an attacker the opportunity to generate a carefully-crafted spoofed message that would pass the verification. In order to provide a level of protection against future quantum computers, it is also important that the hash function is regarded as 'quantum resistant'. The Secure Hash Algorithm (SHA) '2' family of hash functions as defined by the National Institute of Standards and Technology (NIST) in Federal Information Processing Standards (FIPS) 180-4 [46] meets both these requirements, with SHA-256, which generates a hash 32 bytes in length, being a suitable candidate for application to TRDP PDUs[40]. The proposed mitigation, therefore, to address this fundamental vulnerability is to append a SHA-256 HMAC, generated using a symmetric 32 or 64 byte key, to each TRDP PDU. This modification is shown in Figure 11 for the Pd PDU, and described below[41].

- In this TCMS implementation, which is based upon IEC 61375-2-3:2012 CD: add the HMAC immediately after the `FCS`, computing it over the entire PDU including the `FCS`.

[40] FIPS 180-4 also defines SHA-512, which produces a longer hash value than that of SHA-256: 64 bytes instead of 32. This gives greater collision resistance should it be required in the future, although it would need an additional 32 bytes of space in the TRDP PDU to accommodate the longer hash value. Other hash functions, such as those in the SHA-3 family, could also be considered, although these would have a similar affect on the PDU.

[41] The TRDP Md PDU has not been included in this discussion as none were detected during this case study. However, the SHA-256 HMAC could be added to it in exactly the same way, noting that the Md PDU does not have the same length constraint as that of the Pd PDU; it is therefore able to accommodate such a modification.

The HMAC will not be included in the FCS calculation, meaning that the final 32 bytes of the PDU can be ignored by an ED that is unable to validate it.

- In the IEC 61375-2-3:2015 version, which has been included for completeness: add the HMAC immediately after the Dataset, computing it over the entire PDU including the HeaderFCS header. The HMAC will not be included in the HeaderFCS calculation, as this value is computed over the header only.

The unfortunate consequence of the addition of this HMAC is that 32 bytes of the TRDP Pd PDU's Dataset are lost, remembering that this particular PDU has a maximum length of 1,472 bytes to ensure it can be encapsulated into a single 1,500 byte ethernet frame. This may not be a significant problem and is discussed further in the following section, although it would require a change in the composition of one particular PDU identified during this case study.

There are a number of considerations relating to the use of an HMAC, the most significant being the ability of an ED to perform the required calculation in a timely manner: in the case of this TCMS implementation, in a way that is commensurate with the 20ms communications period. Whilst it would be unlikely to introduce a delay to more computationally powerful EDs such as the TMC or an ETBN, others, such as those produced by third-parties (for example, the external door controllers), may be unable to compute the HMACs within the period. However, many EDs, including the ETCS unit, were connected to the ETB by means of an interface unit. This significantly reduces the work required to facilitate an HMAC: the interface units could be upgraded, if required, rather than the individual EDs they support. In the case of an ED that has a direct connection to the ETB but is unable to participate in HMAC validation, either because of a lack of computational power or its ability to be upgraded, the HMAC could simply be ignored until the offending unit is replaced with a more suitable version[42]. Whilst not ideal, HMAC validation is essentially optional and could be enabled in phases, allowing it to be included as part of a medium-term train upgrade programme.

A second consideration refers to the detection and reporting of invalid HMACs. Attempted TRDP PDU spoofing attacks, such as that described in Section 5.9, would be detected by the TMC after receiving a sequence of PDUs containing invalid HMACs. When this occurs, the TMC could send an alert to the train operator's SEIM system. However, when targeting other EDs, for example by submitting a fake train speed in 'Pr' *request* common data, it may be that the receiving ED cannot be configured to generate an alert. It may therefore be beneficial for intermediary components, such as ETBNs, to participate in HMAC validation as they do for the FCS, reporting violations as they are detected[43]. Alternatively, an IDS, such as that active on the security gateway, could be configured to do the same, generating an alert when the the number of failed HMACs reaches a particular threshold, or when the behaviour represents a departure from that considered 'normal'.

6.3. TRDP Pd PDU lengths

During this case study, a total of 6,525,948 packets were captured from the ETBs on the trains and at the test-bed. Of these, 6,147,280 were TRDP, and all took the form of Pd PDUs. The distribution of the lengths of these PDUs is shown in Figure 10, and we observed that there were 49 possible lengths, with 304 bytes being the most common (821,801 PDUs).

Given that the maximum permissible length of a TRDP Pd PDU is 1,472 bytes, so it can be encapsulated into a single ethernet frame, it must be no longer than 1,440 bytes if it is to accommodate an additional 32 byte SHA-256 HMAC. An examination of the lengths of the captured PDUs revealed how many were capable of accommodating such an HMAC.

- 6,051,187 (98.44%) PDUs had a length less than 1,440 bytes and were therefore able to accommodate a SHA-256 HMAC.
- 96,093 (1.56%) PDUs had a length that was exactly 1,472 bytes and were therefore unable to accommodate a SHA-256 HMAC.

The 1.56% of TRDP Pd PDUs that were unable to accommodate a SHA-256 HMAC originated from either the TMC (68,698 PDUs) or monitoring server (27,395 PDUs), and all were destined to the safety-related multicast addresses 239.10.67.1 (ETB 1) or 239.20.67.1 (ETB 2), which was believed to be a group comprising the TMC, monitoring server and display unit[44]. There are several options with regard to the PDUs that were unable to accommodate the HMAC. These are as follows.

1. The specification governing the Dataset of the offending PDU could be re-written, with it being reduced in length by 32 bytes. The following PDUs would then contain this excluded data.
2. In particular cases, and as a retro-fit to this TCMS implementation, the Dataset could remain unchanged, with the HMAC being conveyed in a second follow-up PDU. PDUs not containing an HMAC could be identified by a specific attribute, such as their destination multicast address (for example 239.10.67.1) or ComId, combined with the fact that they are exactly 1,472 bytes in length

[42] A sending ED could dynamically indicate its inability to generate an HMAC, for example by toggling a bit in each TRDP PDU Dataset it sends. However, as this inability would be known in advance, it would be preferable for the receiving ED to be configured to ignore the appended HMACs in PDUs submitted by the affected ED. This would prevent an attacker from *down-grading* a PDU in order to exclude it from the receiving ED's verification process.

[43] As applied to the IDS and has been described more fully in Section 6.8, an ETBN performing HMAC validation would require access to *all* the keys with which the evaluated HMACs were generated. This could be achieved by placing a copy of every relevant key on the ETBN, or preferably by implementing a key derivation function that allows it to derive individual ED keys.

[44] Unfortunately, it was not possible to ascertain the purpose of these PDUs in the time available. It was noted, however, that not every PDU sent to these two multicast addresses had a length of exactly 1,472 bytes (and that those with a shorter length would be capable of accommodating a SHA-256 HMAC).

Period	Hours	Days	Years
Short (20ms)	23,860.92	994.20	2.72
Long (200ms)	238,609.29	9,942.05	27.23

Table 3: Time required for the four byte TRDP `SequenceCounter` to repeat

(or, more generally, have a length greater than 1,440 bytes meaning they are unable to accommodate an HMAC). On receipt of such a PDU, the ED would simply buffer its contents and wait for the next PDU which, arriving with an incremented `SequenceCounter`, would contain the HMAC of the previous PDU in its `Dataset`.

3. As a longer-term solution, jumbo ethernet frames could be investigated as a way to increase the maximum size of a Pd PDU, even if only by 32 bytes[45]. Unlike frames defined in the IEEE 802.3 standard, which are limited to 1,500 bytes in length, their jumbo variants are often 9,000 bytes and would therefore be able to accommodate the additional 32 bytes required for the SHA-256 HMAC.

The accommodation of any of these options would almost certainly require a software update of the affected EDs, and in the case of jumbo frames, possible hardware upgrades throughout the TCMS, for example to the ETBNs. Given their recent delivery, hardware upgrades may be deemed unsuitable for the trains examined during this case study; however, a software update could be performed during periodic maintenance and may therefore offer the most pragmatic solution.

6.4. Replay prevention

The addition of an HMAC to a TRDP PDU would significantly increase the difficulty of a packet-spoofing attack, such as that described in Section 5.9. In this case, an attacker would be unable to generate a valid HMAC for the injected PDU, thus causing it to be rejected by the TMC. Despite this, it would be possible for the same attacker to collect a large number of PDUs containing a variety of different train speeds, and to replay them into ETB 1 at a point in time when the `SqeuenceCounter` was valid. In order to render this unfeasible, the PDU must contain a sufficient quantity of timely data by which it could be validated, and for this data to also be passed through the HMAC function.

The `Dataset` in the spoofed TRDP Pd 'Pd' *data* reply PDUs injected when 'racing the SqeuenceCounter' did not contain a sufficient quantity of timely data: there was a counter at byte 66, but an invalid value in this location did not prevent the attack from succeeding. As there were no other similar variables in the `Dataset`, and as it may be unwise to rely on this ever being the case, the TRDP headers offer the most suitable location in which the timely data could be stored.

The IEC 61375-2-3:2012 CD TRDP Pd 'Pd' *data* reply PDU used in this TCMS implementation contained the following headers, the suitability of which to provide timely data will now be described.[46].

- `SqeuenceCounter`: Table 3 shows how long it would take for the four byte `SequenceCounter` to repeat with respect to both long and short period communications, with a repeating value providing an opportunity for an attacker to replay a PDU. The time periods in this table assume that each component maintains the state of its `SequenceCounter`, together with those of the components with which it communicates, and that it is always incremented from its previous value, even after inauguration[47]. Unfortunately, this was not the case on the trains and at the test-bed, where all component `SequenceCounter` values were reset to '0' following inauguration[48]. The regular resetting of this header severely degrades its ability to prevent the replay of a PDU.

- `TopoCount`: This value denotes the current train configuration following inauguration. According to IEC 61375-2-3:2015, the equivalent header, which appears to be `etbTopoCnt`, is obtained by passing the current train directory through the IEEE 802.3 CRC-32 function to produce a four byte unsigned integer that represents the current train topology. As this directory does not contain any timely data, the output will be the same for any two identical train configurations (i.e. it will be the same value following a simple power-cycle). Consequently, it does not change frequently enough and should therefore not be relied upon to prevent replay attacks[49].

- All the remaining headers were either fixed or unsuitable (where given, values are in hexadecimal): `ProtocolVersion` was always set to `0100` denoting TRDP version '1.0'; `MsgType` was set to `5064`

[45] An increase of 64 bytes would ensure that a SHA-512 HMAC could be considered in future TCMS upgrades.

[46] Although this case study is concerned with the TRDP Pd PDUs specified in IEC 61375-2-3:2012 CD, as were found on the trains and at the test-bed, the same need for timely data and an HMAC also applies to those specified in in IEC 61375-2-3:2015. The PDUs specified in IEC 61375-2-3:2015 contain additional headers as shown in Figure 3, namely `etbTopoCnt`, `OpTrnTopoCnt` and `ReplyComId`; however, these also do not contain timely data. The same applies to the Md PDUs.

[47] Inauguration may occur at least once per day, for example when coupling or uncoupling a car, or power-cycling the train.

[48] IEC 61375-2-3:2015 specifies that the `SequenceCounter` should have a "start" value of '0' for both TRDP Pd and Md PDUs. This behaviour is believed to be necessary in order to support the addition of components, for example when the train configuration is changed or a faulty component replaced. In these cases, each newly-added component would be unaware of the next `SequenceCounter` expected by the receiver.

[49] The data passed through the IEEE 802.3 CRC-32 function when generating the TRDP `etbTopoCnt` header as defined in IEC 61375-2-3:2015 includes one or more Consist Universally Unique Identifier (CstUUID) values derived from the train inauguration process. As the CstUUID assigned to each consist (and set accordingly on each ETBN) is likely to be unique, different trains will generate different `etbTopoCnt` headers. Therefore, the inclusion of `etbTopoCnt` in the HMAC will prevent a TRDP PDU captured on one train from being successfully replayed on another. It will not, however, prevent the same attack against a single train, on which the PDU was both captured and replayed.

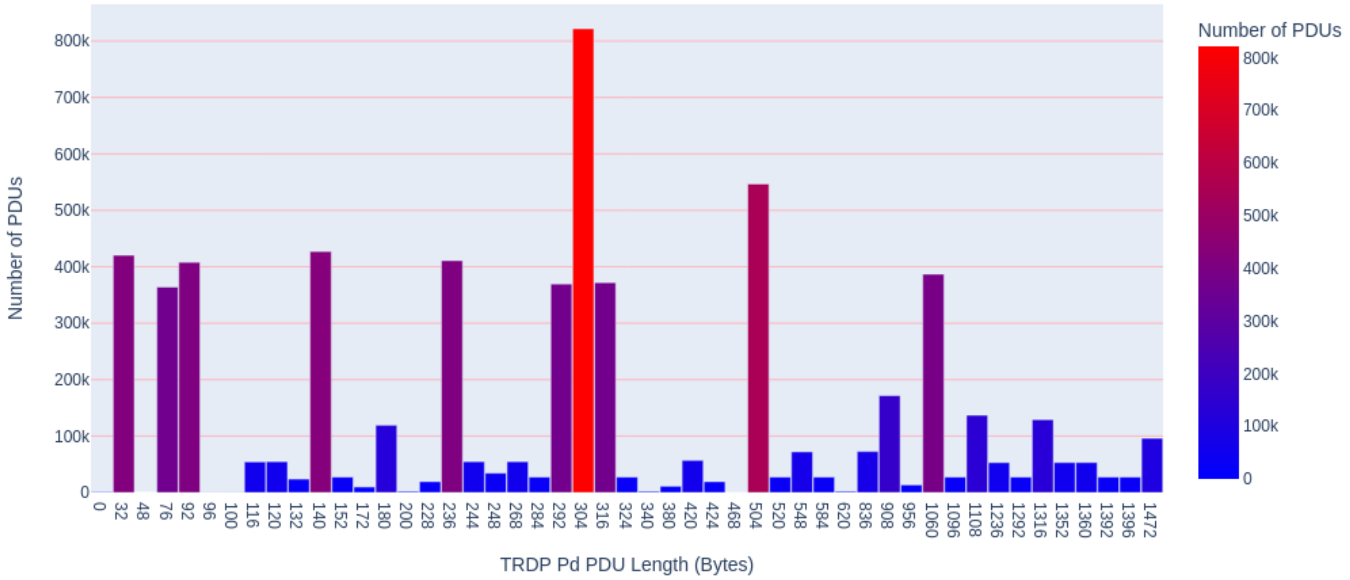


Figure 10: Lengths of the 6,147,280 captured TRDP Pd PDUs (those with a length greater than 1,440 bytes cannot accommodate a SHA HMAC)

denoting a 'Pd' *data* reply; ComId identified the type of Dataset and was identical in every PDU sequence; DatasetLength denoted the length of the Dataset which remained constant and valid when bytes were modified, as occurred during the attack; Reserved was set to 0000 as specified in the standard; ReplyIpAddress appeared to be unused and also set to 0000.

As can be seen, all these headers are either unsuitable or unreliable, and cannot provide the required timely data to prevent replay attacks. It would therefore be prudent to include a specific anti-replay mechanism in all TRDP PDUs: a four byte timestamp in the header[50]. In order not to increase the length of this PDU or further reduce its Dataset, the timestamp could be placed in the first four bytes of the Reserved header found in this TCMS implementation[51]. The resulting TRDP Pd PDU would be as shown in Figure 11, with an identical change made to the Md PDUs.

The inclusion of a timestamp would require all participating EDs to obtain and maintain the correct date and time, within a tolerable period of deviation. Thankfully, this information is already provided in the common data situated in the Dataset of TRDP Pd 'Pr' *request* PDUs sent by the TMC. These PDUs are either sent directly to EDs, or are accessible by them. Therefore, assuming that these PDUs possess a valid HMAC, they are capable of providing time synchronisation; they also negate the need for an alternative, such as the Network Time Protocol (NTP), which would require additional modifications to the TCMS.

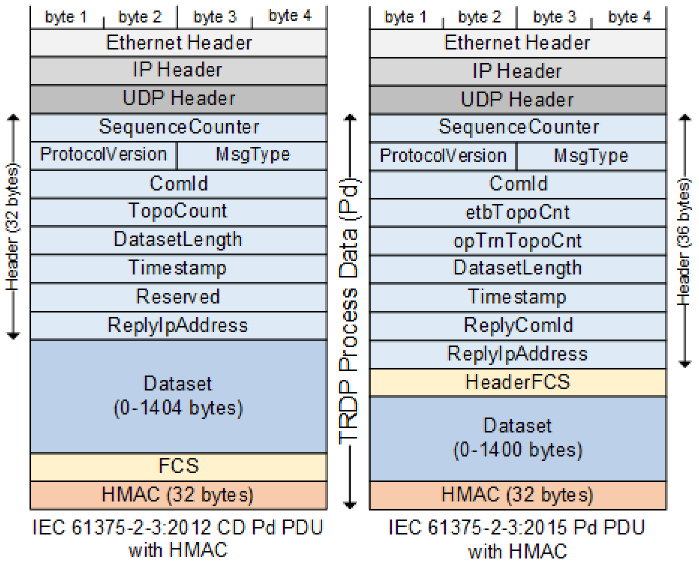


Figure 11: IEC 61375-2-3:2012 CD and 61375-2-3:2015 Pd PDUs with inserted 4 byte Timestamp and 32 byte HMAC

[50] It would also be possible to associate a specific TRDP Pd 'Pr' *request* with a corresponding 'Pd' *data* reply and thus implement a challenge-response based HMAC solution (for example, using the ComId and ReplyComId fields), or by doing the same in the Dataset. However, the ReplyComId header was not present in the PDUs used in this TCMS implementation, and such behaviour may interfere with functions that rely on receiving a specific value in one or both of these headers. Similarly and as mentioned in Section 5.6, the *streams* of 'Pr' *request* and 'Pd' *data* replies were not closely associated in that the latter was not sent in response to the former; therefore, mandating such an association may require a time-consuming and costly re-design of the TCMS.

[51] The timestamp could also replace the entire four byte Reserved header in the TRDP Pd and Md PDUs defined in IEC 61375-2-3:2015.

6.5. HMAC key management

The modification of the TRDP Pd and Md PDU to accommodate an HMAC is relatively straight forward, but it comes with an overhead that is associated with every cryptographic solution: key management. It is important that the proposed key management scheme is not a burden, and that it integrates unobtrusively into the fleet maintenance programme. There are various options that could be considered, and we present several of those below, ranking them in order of the security they provide and complexity they would introduce.

However, there are several points to consider before doing so. First, that even the most simple key management scheme supporting the proposed HMAC represents a significant improvement in security: it would increase the difficulty of successfully launching a TRDP PDU spoofing attack. Second, that it acknowledges that, as keys would be stored on EDs, the compromise of an ED would likely result in the attacker gaining access to its key. This is an unfortunate fact, but its likelihood has been mitigated by the TCN architecture and approach taken by the train manufacturer to integrate cyber security both early in the design process and throughout the train's lifetime, for example by subjecting it to an annual penetration test. The residual risks can be further reduced by the security controls described in this section of the paper, in particular those pertaining to packet filtering, IDS placement and physical security, and by the consideration of procedures typically associated with cryptographic key management, such as the secure generation, distribution and revocation of keys, and the secure destruction of the components on which they have been stored. Third, that any key management scheme must take into account the open nature of the trains, in that they may be coupled together in order to form a longer vehicle. The consequence of this is that it is unlikely to be possible to provide keys that are unique to a single train, and that, instead, they must be common to the entire fleet[52]. With respect to the numbers of keys, the following options are available to support the addition of the HMAC to the TRDP PDUs.

1. Every ED in every train would be configured with the same key, and this key would be used to perform all HMAC validation. This is clearly the most simple option and the one with the lowest management overhead; however, the compromise of a single ED would render the entire fleet vulnerable until the keys could be replaced.
2. Every ED would be configured with a unique key, shared only with those EDs with which it communicates. This is the most secure option, as a compromised key would only render one particular ED's communications vulnerable (across the entire fleet). It also, however, has the greatest management overhead, as unique keys would need to be generated and distributed to every ED, the total number of which would be in the hundreds.
3. As a middle-ground, the key management scheme could be based around an ED's `DeviceId`. In this scheme, each component type, denoted by its `DeviceId`, would have its own key. For example, every ETCS unit, which had a `DeviceId` of 6, would be configured with an identical key, and these would differ from those of other ED types, such as the traction unit and HVAC. With 26 possible `DeviceId` values in existence, as shown in a technical specification document, this would mean 26 different keys per fleet[53].
4. Similar to the above, keys could be allocated to the destination multicast addresses with which EDs are associated. An analysis of a technical specification document revealed a total of 35 possible multicast addresses, but these could be reduced by the merger of associated addresses allocated to each ETB (i.e. `239.10.13.1` on ETB 1 and `239.20.13.1` on ETB 2 could be associated with a single key), and those that denote short and long period types. This would make it possible to reduce the number of keys to approximately 15.

The advantage of the third and forth of these schemes is that it would be relatively easy to associate a key with a particular component, collection of components or multicast address. This would simplify the software updates required to facilitate the addition of the HMAC. It also results in a lower number of keys than if every ED was configured with its own.

To simplify the management of components that validate TRDP PDUs originating from a large number of EDs, such as the TMC or secure gateway when operating as an IDS, a key derivation function could be employed. In this way, each ED key would be derived from a single *master* key, for example by concatenating the master key to an ED identifier, such as a `DeviceId` or destination multicast address, before passing it through the HMAC function to generate the ED key. By following this approach, components such as the TMC would not need to be configured with copies of individual ED keys; instead, as they would be in possession of the master key, they could generate them as required.

The discussion in this section has assumed that HMAC authenticity and integrity verification would be performed using symmetric cryptography. If sufficient computational power was available to all EDs, as may be the case in the future, it would be possible to implement an alternative asymmetric public key system that would use digital signatures to validate TRDP PDUs. Such a system, which may take advantage of a rolling-stock PKI, would require further research and significant investment; however, it may offer a more elegant and sustainable long-term solution.

[52] It may be that *all* trains within a particular fleet must be configured with identical keys to ensure they can be coupled to *any* other train from the same fleet. However, it may be that only *certain* trains are actually ever coupled. This would allow sub-sets of trains within a fleet to be configured with identical keys, thus reducing duplication and increasing the security of the system.

[53] The exact number of keys needed would require further research, as it is dependent on the exact numbers of communicating entities rather than the total number of EDs. For example, when communicating with an interface unit, to which was connected a ETCS unit and other EDs, the TMC would use the interface unit's key when generating and validating HMACs, rather than the key of every ED connected to it. Similarly, communications between the TMC and another ED, such as an external door controller, would only require a single key: that of the external door controller. There may be no requirement for the TMC to have its own individual key.

There are other important aspects of key management that will require further research. These include the method by which the keys are distributed, the regularity of the subsequent re-keying, and the way by which they can be revoked. It may be that experience drawn from other systems that have a similar cryptographic requirement, for example ERTMS communications, could be applied in this area.

6.6. ERTMS data

Looking towards the future, ERTMS data passing from the wayside to the train's ETCS unit is cryptography protected, with the ETCS unit being able to ascertain the authenticity and integrity of the data it receives. It is vital that this data, when forwarded by the ETCS unit to the TMC, is similarly protected. This could be achieved by giving the TMC access to the same cryptographic mechanisms and keys as those used by the ETCS unit, with the ETCS unit then simply forwarding to the TMC the ERTMS data it received. Alternatively, the same level of protection could be obtained using the HMAC-protected TRDP PDUs recommended in this paper. A failure to do so would allow an attacker to inject fake ERTMS messages into the TCMS, which may affect the safety of the train[54].

6.7. Packet filtering

Packet filtering is perhaps the simplest technical method by which to prevent the attack described in Section 5.9, together with other spoofing attacks launched from across the ETB. The ETBNs offer a suitable location for the application of this particular security control: a small number of filtering rules applied to each ETBN would prevent the passage of spoofed IP packets passing between CNs and their constituent EDs.

It is important that the implemented filtering rules are few in number and operate on the IP packet headers only, as opposed to those requiring the *real-time* dissection of TCP and UDP packets to reveal the encapsulated TRDP PDUs. The reason for this is to ensure that the filtering process does not introduce unacceptable quantities of latency and jitter, as was described by Wusteney et al. following their analysis of packet filters in time-sensitive networks [47], and Zvabva et al. who performed a similar study in an experimental industrial setting [48]. As the TCMS is a safety-related system supporting periodic data sent at 20ms intervals, it is vital that filtering does not degrade network performance to such an extent that compromises its safe and efficient operation, or require a significant system redesign.

Two suggested filtering rules are described below, with each being expressed in the Linux *iptables* command format, and including examples taken from the attack described in Section 5.9 and ETB shown in Figure 5.

The purpose of the first rule is to prevent spoofed packets from *entering* a CN. The ETBN, acting as the gateway between the backbone and is local CN (and being aware of the subnets to which it is connected), is able to deny the entry of packets to that CN if they claim to originate from within it. For example, ETBN-1A could be configured to drop all packets arriving at either of its backbone interfaces that originate from an IP that falls within the `10.128.64.0/18` subnet[55]. The *iptables* rule to be applied on ETBN-1A would be as follows, where `ETB_IFACE` refers to both of the ETBN's backbone interfaces, and `CN_IFACE` to its CN interface:

```
iptables -A FORWARD -i ETB_IFACE \
  -o CN_IFACE -s 10.128.64.0/18 -j DROP
```

This could be taken a step further by removing the output requirement (`-o CN_IFACE`), which would cause the ETBN to also deny spoofed packets as they pass through it on their journey along the backbone.

A second and similar rule could also be added to prevent spoofed packets from *leaving* a CN, for example on ETBN-1B:

```
iptables -A FORWARD -i CN_IFACE \
  -o ETB_IFACE ! -s 10.128.128.0/18 -j DROP
```

This would not deny spoofed packets that have been injected directly into the backbone, but it does add a second layer of protection from malicious or compromised EDs situated in a CN.

The `ETB_IFACE` interface list in both these rules may be limited to just the pair of interfaces connected to the backbone; however, it may also contain the maintenance port in order to prevent the direct connection of malicious devices to the ETBN[56]. In addition, the *iptables* DROP rule should be preceded by an identical LOG rule in order to generate an alert from the dropped packet. This would allow the event to be reported to the train operator's SEIM system.

Either of these two filtering rules would have prevented the attack described in Section 5.9, as it was launched from a CN different from that containing the intended targets. This, however, assumes that the TMC checked that the source IP address of the received packet was that of the interface unit, before accepting the TRDP Pd 'Pr' *data* reply PDU as valid. If it did not, there would have been no need for the Raspberry Pi to spoof the interface unit's IP address, and the attack would have succeeded irrespective of the above filtering rules[57]. Regardless of this, it would have still been possible to perform the attack by physically moving the Raspberry Pi into the CN containing

[54] It would be advantageous to ensure the authenticity and integrity of *all* data passing between the ETCS unit and TMC, regardless of its direction of travel.

[55] The fact that this did not happen meant that the Raspberry Pi was able to send spoofed packets to the TMC by purporting to be the interface unit, even though it was situated in a different CN from that containing both these EDs.

[56] The application of packet filtering rules to the maintenance port would have an impact upon the use of the legitimate maintenance laptop and further consideration would need to be given as to how maintenance would be undertaken. For example, access could be granted to packets originating from the laptop's static IP address at one or more of the ETBNs, or by a direct connection into a CN.

[57] In this case, the Raspberry Pi would have simply required a valid IP address in the CN to which it was connected, and for this legitimate IP address to have been used as the source of the attack. Unfortunately, it was not possible to validate this behaviour of the TMC in the time available and it therefore remains an area of further research.

these two EDs; however, this would have restricted the attacker to two possible entry points—one in each cab—or require the prior compromise of a component in one of these two locations. As can be seen, whilst not perfect, security controls such as these do increase the difficulty of undertaking a successful attack when applied as part of a layered approach.

As a final recommendation, a detailed analysis of the TRDP communication profile, noting valid unicast source and multicast destination IP address pairs, would allow for the creation of rules that allow packets to pass if they confirm to its specification, whilst denying all others by default. For example, ETBN-1A would be unlikely to be configured with a rule requiring it to forward a TRDP Pd 'Pd' *data* reply PDU sent to the `239.10.13.1` multicast address, as this destination is reserved for the PDUs submitted by the ETCS unit, which resides in its own CN. Such analysis may result in a large number of packet filtering rules and introduce unwanted complexity to the ETBN configurations; however, that may not be the case, and assuming the rules do not prevent the TCMS successfully failing-over in the event of an ED failure, the implementation communications profile filtering rules would further enhance the security of the TCMS. It is therefore worthy of further research in order to ascertain its suitability.

6.8. IDS placement

In Section 5.7 we mentioned that the security gateway, on which was running an IDS, had a connection to ETB 2, together with other non-TCMS networks[58]. This can be seen in Figure 5, which shows how it was integrated into the wider TCN. The connectivity of this component meant that the IDS was able to monitor some unicast and all multicast traffic on ETB 2, but that it had *no* visibility of the traffic on on ETB 1. The irony of the decision to connect the security gateway to ETB 2 is that the attack described in Section 5.9 could only succeed on ETB 1—the network to which the security gateway was *not* connected and was therefore unable to monitor. There are two solutions to address this unfortunate shortcoming: move the security gateway to ETB 1; or, preferably, ensure that it is able to monitor the traffic on *both* ETB instances.

Moving the security gateway from ETB 2 to ETB 1 may appear to be the simple option; however, the TCMS requires it to be situated in the network containing none safety-related EDs that are only connected to ETB 2. Such a move, therefore, would likely require a significant TCMS re-design; it would also leave ETB 2 exposed to potential undetected attacks. The preferable solution would be to connect the security gateway to ETB 1 *and* ETB 2, and for the newly-connected ETB 1 interface to operate in a passive manner: monitoring traffic but not participating on the network (i.e. the interface would not be configured with an IP address). This would provide IDS coverage to the entire TCMS, and is thought to be possible because of an apparently unused interface on the security gateway, in addition to that reserved for management purposes. In some industrial networks, such connectivity may be regarded as bridging or bypassing the clearly defined zones; however, in this case both ETBs effectively operate as a single safety and security zone, and already contain a number of dual-homed EDs that are connected to both networks, such as the TMC. Once deployed in this way, the IDS could then be configured to detect the attack described in Section 5.9, and indeed any other attack launched from or against ETB 1. The former could be achieved by identifying two TRDP Pd 'Pr' *request* PDUs sent within one period of each another that have an identical `SequenceCounter` and other TRDP header values, but a different `Dataset`[59]. In addition and following the recommendation made in Section 6.2, the IDS could be also configured to detect invalid HMACs, which may be indicative of malicious behaviour[60].

6.9. Physical security

Section 5.8 explained how the TCMS components were fitted in cabinets secured with hexagonal recess locks, as shown in Figure 7. Indeed, excluding possible human intervention, these locks were all that stood between an attacker and the ETB. Whilst not wanting to introduce a complicated system that hampers maintenance, it seems sensible for these hexagonal locks to be replaced with (or accompanied by) a keyed alternative, given the sensitivity of the components they contain. Even if the replacement locks and their associated keys were identical across the fleet, it would represent an improvement on the current situation. It may even be practical for each train to have its own set of locks and keys, and for the same approach to be applied to cabinets containing non-TCMS components, such as the PIS.

It is important to state that this improvement would not prevent a physical attack, as an attacker would be able to pick, drill or cut the lock, or possibly obtain a copied or stolen key. It would, however, slow the attacker down and increase the likelihood of detection. This improvement is not perfect in itself and would form part of a layered approach to security, with proportionate controls added wherever possible in order to deter or hamper an attack. Taking this a step further, tamper seals and/or magnetic contact switches could be added to the cabinets in which TCMS components reside. This would offer yet another layer of physical security, and allow for the detection of opened cabinets, especially outside scheduled maintenance

[58] It was not possible to assess the behaviour of the IDS during this case study, or its ability to detect the attack described in Section 5.9. However, it is hoped that this will be an area of future research.

[59] The modified `Dataset` bytes would likely reveal the intended purpose of the attack.

[60] To validate an HMAC, the IDS would require access to the key with which it was generated. In the case of the secure gateway inspecting traffic on the ETB, it would ideally need to be configured with *all* the HMAC keys distributed throughout the train. As mentioned previously, this could be achieved by placing a copy of every relevant key on the secure gateway, or preferably by implementing a key derivation function that allows it to derive individual ED keys. This may be deemed appropriate as the component already provides an essential security function; however, the collection of keys in a component connected to higher-risk non-TCMS networks and housed in a publicly accessible compartment must be included in any future risk assessment, especially one that considers the provision of physical security.

periods. This recommendation is likely to require additional research in order to assess its feasibility and cost, and the impact it would have on the long-term maintenance of the fleet.

6.10. Summary

In this section, we proposed a series of mitigations that would address the vulnerabilities identified in Section 5, and in doing so further improve the cyber security of the trains inspected during this case study. These recommendations vary in complexity: from the seemingly straightforward connection of an additional IDS interface and the replacement of locks; to a modification TRDP that is used to provide the TCMS. It is clear that the complexity of managing a fleet of trains is significant, and the implementation of any of these recommendations will require further research to assess their impact. This applies to both the trains examined during this case study, and also to new fleets currently being designed, noting that it is more efficient to make improvements early, rather than to retro-fit them post-delivery.

In the next section, we will draw conclusions and consider future TCMS implementations. We will describe how the use of an accompanying control and monitoring system, such as a train control circuit, will continue to enhance the security of the TCMS, mitigate future vulnerabilities, and ensure the continued production of safe and secure trains.

7. Conclusions

7.1. Design and approach

The train manufacturer designed and implemented a fully-operational TCN, including a TCMS in which were situated safety and non-safety-related components. Connections from the TCMS to external networks, such as the PIS and ultimately the wayside, are made by means of a security gateway, that monitored and controlled traffic passing between the zones. Passenger wifi had been physically separated from all other networks.

The manufacturer's approach was considered and informed: it incorporated aspects of IEC 62443 into the TCN design; it followed the parts of IEC 61375 pertaining to the ETB, with some minor omissions caused by overlapping system design and standard publication dates; and included the owner and operator in the entire design process, together with external third-parties to further enhance the security of the train. Workshops, attended by a experts in different fields, were regularly held to identify, understand and mitigate risks.

It was evident that cyber security was continuous and key aspect of the design and manufacturing process, and that this looks set to continue through the delivery and maintenance phases, and indeed throughout the lifetime of the train.

7.2. First generation ETB

The train was fitted with a first generation ETB, meaning that although the architecture was capable of supporting control, its purpose was to facilitate monitoring, with control being provided by additional technologies, such as the train control circuit. This separation meant it would not be possible for an attacker, who had gained access to the ETB, to take control the train's safety systems, such as traction or braking. However, they would be able to cause significant disruption, for example by providing the driver with incorrect status information; or even a operational incident, for example by causing the displaying of an incorrect train speed, which may cause the driver to travel too quickly over a set of points or around a bend.

A key conclusion drawn from this relates to future TCMS designs: dual control and monitoring systems, such as an ETB coupled with a train control circuit and ETCS DMI, provide resilience against the compromise of any one of these systems. In such a case, the impact of an ETB compromise would problematic but not catastrophic, especially if potential vulnerabilities are continuously considered and mitigated where appropriate, including those detailed in this paper.

7.3. Attacking the TCMS

We demonstrated that it was possible to use the 'racing the SequenceCounter' attack to target the TMC and cause a fake train speed to be displayed to the driver. This was undertaken using commodity hardware, access to the IEC 61375 international standard series, and following experiments performed on a train and at a test-bed, albeit for a limited period of time. The skills and access required to undertake such an attack may exceed those of many potential attackers; however, they are certainly within the capabilities of a nation state, organised crime group or competent gang of cyber criminals.

The fundamental vulnerability exploited during the case study exists because the manufacturer implemented TRDP as specified in IEC 61375-2-3:2015, rather than as the result of a design decision it took, or a particular defect with the design or implementation. Indeed, this vulnerability applies to *all* trains built according to this standard, regardless of their manufacturer. Our primary recommendation, therefore, is to incorporate an HMAC cryptographic authenticity and integrity verification mechanism into TRDP, either by making a local custom modifications to the PDUs, or preferably through an update to the standard. When combined with our secondary recommendations that refer to packet filtering, IDS placement and packet filtering, such an action would significantly increase the security of the TCMS.

7.4. Second and third generation ETBs

It is likely that future train designs will be based on the ETB as specified in IEC 61375-2-5:2014. It is understandable that the desire to reduce weight, manufacturing, running and maintenance costs, and the complexity of future designs, will result in a greater use of the ETB as both a control *and* monitoring system, with the integration of separate displays into a single screen. Here, we advise a cautionary approach: the removal of an alternative control system or display, such as the train control circuit and ETCS DMI, should only be undertaken after the potential consequence have been fully understood. Specifically, this is not recommended whilst TRDP fails to incorporate appropriate cryptographic authenticity and integrity verification.

7.5. Layered approach

The security of the TCN, in particular the TCMS, is best achieved following a layered approach: no single security control is likely to deter or prevent an attack. This has largely been achieved by the train manufacturer, although it appears that the TCMS, representing the *core* of the TCN, has been excluded from this process. We believe that it is essential that the security layers should be extended into the TCMS so that the outcome of any future compromise will be limited. It is therefore critical that the protocol used to monitor and control the train, TRDP, is not vulnerable to attacks that undermine its integrity, or that simple physical actions can place an attacker into the core of the TCMS.

7.6. Further research

In the paper, we made various references to potential further research that would increase our understanding of the TCMS examined during the case study. These included: determining how the display unit obtained the speed it showed to the driver in order to simplify and better target the attack; the purpose of the TRDP Pd PDUs that were unable to accommodate a SHA-256 HMAC due to their length; a key management system capable of supporting the integration of this HMAC into a fleet of trains; the optimum configuration of packet filtering rules on the ETBNs; whether the TMC validated the source IP address of PDU in order to determine its authenticity; the configuration of the IDS and its ability to detect the 'racing the Sequence-Counter attack'; and proposed modifications to the locks fitted to cabinets containing TCN components.

Given the relatively limited period of time during which the research took place, greater access to the test-bed would likely result in additional findings and recommendations, including those that would further enhance the security of the trains. It would also be useful to extend the research beyond one particular train class and to include those manufactured elsewhere. This would provide for a more thorough understanding of different TCN designs, vulnerabilities and mitigations, and lead to the publication of further case-studies and best practice guidance that would benefit the railway industry as a whole.

Given the number of components connected to the the ETB, including those that submit alerts and data to the wayside, it would be interesting to understand how this information is processed by operator, and how it could be best interpreted to ensure an attack is detected at the earliest opportunity.

During the case study, the train inauguration process was not examined. This process is likely to reveal additional vulnerabilities, such as those pertaining to the reported train length and its use in calculations performed by the TMC, together with potential denial of service attacks, for example by causing a train to repeatedly fail to inaugurate and therefore remain out of service.

Looking forwards, it would be beneficial to devise an assessment framework based on the research undertaken during this case study. This would allow similar work to be performed to a recognised standard and by different individuals. Also, one that would allow the cyber security of any train to be assessed and determined in a way that gives consistency across the railway industry, and allows a manufacturer to demonstrate the *quality* of their implementation to a train's owner and operator.

References

[1] Anonymous, A cyber security analysis of an ethernet train network, in: Under submission, 2025.

[2] Secretary of State for Transport, Great British Railways: The Williams-Shapps Plan for Rail, His Majesty's Stationery Office, London, UK, 2021, ISBN: 978-1-5286-2465-7, accessed 9th Jan 2024.
URL `https://assets.publishing.service.gov.uk/media/60cb29dde90e0743ae8c29c1/gbr-williams-shapps-plan-for-rail.pdf`

[3] Department for Transport, Rail Cyber Security Guidance to Industry, accessed 9th Jan 2024 (Feb. 2016).
URL `https://assets.publishing.service.gov.uk/media/5efcace9e90e075c52d05a94/rail-cyber-security-guidance-to-industry-document.pdf`

[4] Rail Delivery Group, Rail Cyber Security Strategy, release 1.1, accessed 9th Jan 2024 (Jan. 2017).
URL `https://www.raildeliverygroup.com/about-us/publications/96-2017-01-rail-cyber-strategy/file.html`

[5] HM Government, National Cyber Strategy 2022, accessed 9th Jan 2024 (2022).
URL `https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/1053023/national-cyber-strategy-amend.pdf`

[6] Rail Delivery Group and Rail Safety and Standards Board, KTR v7 Key Train Requirements, version 7, accessed 9th Jan 2024 (Jul. 2023).
URL `https://www.rssb.co.uk/-/media/Project/RSSB/RssbWebsite/Documents/Registered/Registered-content/Using-Standards/KTR-v7.pdf`

[7] Rail Delivery Group, KTR v6 Key Train Requirements, version 6, accessed 9th Jan 2024 (Nov. 2020).
URL `https://www.raildeliverygroup.com/about-us/publications/12715-ktr-v6/file.html`

[8] National Cyber Security Centre, Cyber Assessment Framework, version 3.2, accessed 17th Jul 2024 (Apr. 2024).
URL `https://www.ncsc.gov.uk/static-assets/documents/cyber-assessment-framework-v3.2.pdf`

[9] National Cyber Security Centre, Secure design principles, Guides for the design of cyber secure systems, version 1.0, accessed 28th Feb 2024 (May 2019).
URL `https://www.ncsc.gov.uk/collection/cyber-security-design-principles`

[10] CLC/TS 50701:2023, Railway applications – Cybersecurity, Standard, European Committee for Electrotechnical Standardization (CENELEC), Brussles, Belgium, accessed 16th Aug 2024 (Aug. 2023).
URL `https://standards.cencenelec.eu/dyn/www/f?p=CENELEC:110::::FSP_PROJECT,FSP_ORG_ID:74651,1257173&cs=1D69865FDB0D3AE0252C0C15A453FDB3A`

[11] IEC TS 62443-1-1:2009, Industrial communication networks – Network and system security – Part 1-1: Terminology, concepts and models, Standard, International Electrotechnical Commission, Geneva, CH, accessed 12th Jan 2024 (Jul. 2009).
URL `https://webstore.iec.ch/publication/7029`

[12] EN 50128:2011, Electrical and electronic applications for railways, Standard, Brussles, Belgium, accessed 12th Jan 2024 (Jun. 2011).
URL `https://standards.cencenelec.eu/dyn/www/f?p=CENELEC:110::::FSP_PROJECT,FSP_ORG_ID:43626,1257173&cs=1D08CFCB1437DD03F4B6F8799117C2A1B`

[13] European Union, DIRECTIVE (EU) 2016/1148 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 6 July 2016 concerning measures for a high common level of security of network and information systems across the Union, volume L194, pages 1–30 (19th July 2016).

[14] Department for Transport, Implementation of the NIS Directive, DfT Guidance version 1.1, accessed 9th Jan 2024 (Dec. 2018).
URL `https://assets.publishing.service.gov.uk/med`

ia/5ee3a21ae90e0704315ecf32/implementation-of-the-nis-directive-dft-guidance-document.pdf

[15] European Union, DIRECTIVE (EU) 2022/2555 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 14 December 2022 on measures for a high common level of cybersecurity across the Union, amending Regulation (EU) No 910/2014 and Directive (EU) 2018/1972, and repealing Directive (EU) 2016/1148 (NIS 2 Directive), volume L333, pages 80–152 (27th December 2022).

[16] G. zur Bonsen, The Multifunction Vehicle Bus (MVB), in: Proceedings 1995 IEEE International Workshop on Factory Communication Systems. WFCS'95, 1995, pp. 27–34. doi:10.1109/WFCS.1995.482647.

[17] IEC 61375-3-1:2012, Electronic railway equipment – Train communication network (TCN) – Part 3-1: Multifunction Vehicle Bus (MVB), Standard, International Electrotechnical Commission, Geneva, CH, accessed 9th Jan 2024 (Jun. 2012).
URL https://webstore.iec.ch/publication/5402

[18] IEC 61375-2-1:2012, Electronic railway equipment – Train communication network (TCN) – Part 2-1: Wire Train Bus (WTB), Standard, International Electrotechnical Commission, Geneva, CH, accessed 9th Jan 2024 (Jun. 2012).
URL https://webstore.iec.ch/publication/5398

[19] J. Goikoetxea, Shift2Rail CONNECTA: The Next Generation of the Train Control and Monitoring System, in: Proceedings of 7th Transport Research Arena TRA 2018, April 16-19, 2018, Vienna, Austria, 2018. doi:10.5281/zenodo.1421620.

[20] MOVINGRAIL, Deliverable D3.3: Proposals for Virtual Coupling Communication Structures, version 1.1, accessed 29th Feb 2024 (Feb. 2021).
URL https://projects.shift2rail.org/download.aspx?id=af007bef-6472-46bc-9536-a5fe989e284e

[21] J. Goikoetxea, I. de Arriba, I. Lopez, G. Hemzal, A. Mazzone, Remote driving and command of trains: The Shift2Rail approach., Transportation Research Procedia 72 (2023) 3723–3729. doi:10.1016/J.TRPRO.2023.11.546.

[22] IEC 61375-2-5:2014, Electronic railway equipment – Train communication network (TCN) – Part 2-5: Ethernet train backbone, Standard, International Electrotechnical Commission, Geneva, CH, accessed 9th Jan 2024 (Aug. 2014).
URL https://webstore.iec.ch/publication/5400

[23] IEEE 802.1AX-2020, IEEE Standard for Local and Metropolitan Area Networks – Link Aggregation, Standard, Institute of Electrical and Electronics Engineers, Piscataway, NJ, USA, accessed 14th Feb 2024 (May 2020).
URL https://standards.ieee.org/ieee/802.1AX/6768/

[24] P. Leach, M. Mealling, R. Salz, Request for Comments: 4122 – A Universally Unique IDentifier (UUID) URN Namespace, Standard, Internet Engineering Task Force, Wilmington, DE, USA, accessed 14th Feb 2024 (Jul. 2005).
URL https://datatracker.ietf.org/doc/html/rfc4122

[25] IEC 61375-3-4:2014, Electronic railway equipment – Train communication network (TCN) – Part 3-4: Ethernet Consist Network (ECN), Standard, International Electrotechnical Commission, Geneva, CH, accessed 9th Jan 2024 (Mar. 2014).
URL https://webstore.iec.ch/publication/5405

[26] IEC 61375-2-3:2015, Electronic railway equipment – Train communication network (TCN) – Part 2-3: TCN communication profile, Standard, International Electrotechnical Commission, Geneva, CH, accessed 9th Jan 2024 (Jul. 2015).
URL https://webstore.iec.ch/publication/22903

[27] IEC TS 61375-2-4:2017, Electronic railway equipment – Train communication network (TCN) – Part 2-4: TCN application profile, Standard, International Electrotechnical Commission, Geneva, CH, accessed 9th Jan 2024 (Feb. 2017).
URL https://webstore.iec.ch/publication/31865

[28] H. Kirrmann, P. Zuber, The IEC/EEE train communication network, Micro, IEEE 21 (2001) 81–92. doi:10.1109/40.918005.

[29] K. Gemmeke, Experiences with the implementation of the train communication network, in: Transactions on the Built Environment, Vol. 6, 1994, ISSN 1743-3509, accessed 17th Apr 2024.
URL https://www.witpress.com/Secure/elibrary/papers/CR94/CR94032FU.pdf

[30] D. Ludicke, A. Lehner, Train Communication Networks and Prospects, IEEE Communications Magazine 57 (9) (2019). doi:10.1109/MCOM.001.1800957.

[31] M. Jakovljevic, A. Geven, N. Simanic-John, D. M. Saatci, Next-Gen Train Control / Management (TCMS) Architectures: "Drive-By-Data" System Integration Approach, in: ERTS 2018, 9th European Congress on Embedded Real Time Software and Systems (ERTS 2018), Toulouse, France, 2018.
URL https://hal.science/hal-02156252

[32] A. Astarloa, TSN in the Railway Sector: Why, What and How?, Tech. rep., University of the Basque Country; System-on-Chip engineering, Spain (June 2020).

[33] H. Kopetz, A. Ademaj, P. Grillinger, K. Steinhammer, The time-triggered Ethernet (TTE) design, in: Proceedings - Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, ISORC 2005, Vol. 2005, 2005. doi:10.1109/ISORC.2005.56.

[34] R. Kour, A. Patwardhan, A. Thaduri, R. Karim, A review on cybersecurity in railways, Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 237 (1) (2023). doi:10.1177/09544097221089389.

[35] L. J. Valdivia, I. Adin, S. Arrizabalaga, J. Añorga, J. Mendizabal, Cybersecurity-The Forgotten Issue in Railways: Security Can Be Woven into Safety Designs, IEEE Vehicular Technology Magazine 13 (1) (2018). doi:10.1109/MVT.2017.2736098.

[36] E. H. Okstad, R. Bains, T. Myklebust, M. G. Jaatun, Implications of Cyber Security to Safety Approval in Railway, in: Proceedings of the 31st European Safety and Reliability Conference, ESREL 2021, 2021. doi:10.3850/978-981-18-2016-8_486-cd.

[37] J. Holmberg, Threat modeling for train control and management systems based on the ethernet train backbone, 2016.
URL https://api.semanticscholar.org/CorpusID:59463817

[38] S. Gordeychik, Cyber Resilience of Railway Signaling Systems, 2019.

[39] C. Yue, L. Wang, D. Wang, R. Duo, X. Nie, An Ensemble Intrusion Detection Method for Train Ethernet Consist Network Based on CNN and RNN, IEEE Access 9 (2021). doi:10.1109/ACCESS.2021.3073413.

[40] R. Bloomfield, M. Bendele, P. Bishop, R. Stroud, S. Tonks, The Risk Assessment of ERTMS-Based Railway Systems from a Cyber Security Perspective: Methodology and Lessons Learned, in: T. Lecomte, R. Pinger, A. Romanovsky (Eds.), Reliability, Safety, and Security of Railway Systems. Modelling, Analysis, Verification, and Certification, Springer International Publishing, Cham, 2016, pp. 3–19.

[41] T. Chothia, M. Ordean, J. De Ruiter, R. J. Thomas, An Attack against message authentication in the ERTMS train to trackside communication protocols, in: ASIA CCS 2017 - Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security, Association for Computing Machinery, Inc, 2017, pp. 743–756. doi:10.1145/3052973.3053027.

[42] R. J. Thomas, M. Ordean, T. Chothia, J. De Ruiter, TRAKS: A universal key management scheme for ERTMS, in: ACM International Conference Proceeding Series, Vol. Part F132521, 2017. doi:10.1145/3134600.3134631.

[43] J. Köcher, Cyber security measures for ERTMS from the rail operators' perspective, in: Signalling and Data Communication, Vol. 115, 2023, accessed 27th May 2024.
URL https://incyde.com/fileadmin/user_upload/57_72_Koecher_Poschinger.pdf

[44] O. Levêque, ETCS Implementation Handbook, International Union of Railways (UIC), Editions Techniques Ferroviaires (ETF), 2008, version 2.1, accessed 15th Mar 2024.
URL https://uic.org/cdrom/2011/05_ERTMS_training2011/docs/ETCS_handbookf.pdf

[45] Rail Delivery Group, RDG Guidance Note: ETCS On-Board Equipment, issue 1, accessed 15th Mar 2024 (Oct. 2017).
URL https://www.raildeliverygroup.com/media-centre-docman/acop/281-rdg-gn-nti-005etcson-boardequipmentv2/file.html

[46] Information Technology Laboratory, National Institute of Standards and Technology, FIPS PUB 180-4: Secure Hash Standard (SHS) (Aug. 2015). doi:10.6028/NIST.FIPS.180-4.

[47] L. Wusteney, M. Menth, R. Hummen, T. Heer, Impact of Packet Filtering on Time-Sensitive Networking Traffic, in: IEEE International Workshop

on Factory Communication Systems - Proceedings, WFCS, Vol. 2021-June, 2021. doi:10.1109/WFCS46889.2021.9483611.

[48] D. Zvabva, P. Zavarsky, S. Butakov, J. Luswata, Evaluation of Industrial Firewall Performance Issues in Automation and Control Networks, in: 29th Biennial Symposium on Communications, BSC 2018, 2018. doi:10.1109/BSC.2018.8494696.

Acronyms

ATP Automatic Train Protection

CAF Cyber Assessment Framework

CCTV Closed-Circuit Television

CN Consist Network

CRC Cyclic Redundancy Check

CstUUID Consist Universally Unique Identifier

DfT Department for Transport

DHCP Dynamic Host Configuration Protocol

DMI Driver Machine Interface

ECN Ethernet Consist Network

ED End Device

ERTMS European Rail Traffic Management System

ESCG ERTMS Security Core Group

ETB Ethernet Train Backbone

ETBN Ethernet Train Backbone Node

ETCS European Train Control System

EU European Union

EVC European Vital Computer

FIPS Federal Information Processing Standards

GSM-R Global System for Mobile Communications – Railway

HMAC Hash-Based Message Authentication Code

HVAC Heating, Ventilation and Air Conditioning

IDS Intrusion Detection System

IP Internet Protocol

KTR Key Train Requirements

LAN Local Area Network

MAC Message Authentication Code

Md Message Data

MPH Miles Per Hour

MVB Multifunction Vehicle Bus

NAT Network Address Translation

NCSC National Cyber Security Centre

NIS Network and Information Systems

NIST National Institute of Standards and Technology

NTP Network Time Protocol

OES Operators of Essential Services

ORR Office of Rail and Road

Pd Process Data

PDU Protocol Data Unit

PIS Passenger Information System

PKI Public Key Infrastructure

R-NAT Railway Network Address Translation

RDG Rail Delivery Group

RSSB Rail Safety and Standards Board

SCADA Supervisory Control and Data Acquisition

SEIM Security Event and Incident Management

SHA Secure Hash Algorithm

TCMS Train Control and Monitoring System

TCN Train Communication Network

TCP Transmission Control Protocol

TMC Train Management Computer

TOC Train Operating Company

TRDP Train Real Time Data Protocol

TSN Time-Sensitive Networking

TTDP Train Topology Discovery Protocol

TTE Time-Triggered Ethernet

UDP User Datagram Protocol

UIC International Union of Railways

UK United Kingdom

UUID Universal Unique Identifier

VPN Virtual Private Network

VRRP Virtual Router Redundancy Protocol

WTB Wire Train Bus