Cyber Security in the Rail Sector – An Integrated Approach

Richard J. THOMAS[2], Mihai ORDEAN[1], Tom CHOTHIA[1]

[1] School of Computer Science, University of Birmingham, Birmingham, United Kingdom
[2] Birmingham Centre for Railway Research and Education, University of Birmingham, Birmingham, United Kingdom
Corresponding Author: Richard Thomas (R.J.Thomas@cs.bham.ac.uk)

Abstract

This paper presents a modelling framework for complex ICS and railway systems that enables an automated analysis of threats. The aim of this framework is to discover and explore complex attack paths through these inter-connected system architectures, making it possible to assess the risk to particular assets, test new strategies for mitigating risks, and supporting asset owners to understand their systems through integrated architecture assessments. Existing frameworks allow system owners to identify maximal strategies to protect their systems against particular attacks, however, they do not allow the asset owner to discover threat propagation and rank threats to their system, or require the system owner to determine the risk probabilities themselves. We employ probabilistic analysis using the CVSS framework, allowing system owners to concentrate on defining their architectures, rather than deriving potentially incorrect values of risk to their system. The results of the tool can be used to provide assurance and prioritise security improvements to a system. We provide an extensive example of our tool in use, modelling the security of the ERTMS Rail Signalling Standards and on-board train systems.

Keywords: cyber security; modelling; risk analysis; threat identification; NIS Directive.

1. Introduction

The rail sector has a strong focus on safety and functionality over security, presenting an issue as the system ages, where security was not a primary requirement. As an example, the train to trackside communications protocols in ERTMS employ old ciphers for confidentiality (A5/1 which dates to 1987) and integrity (EuroRadio, proposed in 1997 [1]). Today, vulnerabilities exist to these protocols [3, 9], where system owners recognise the need for security, and the requirement to appraise their exposure to threats. Modelling tools are already used by engineers to rationalise safety assumptions and validate the design of their systems, where the same technique could be used to assess the security of a given architecture. Tools available today allow an asset owner to model the security of their system but have limitations, e.g., requiring the asset owner to express the risk profile of their systems. This requirement is prone to human error, either due to a misunderstanding, or the domain and subject-relevant knowledge not being captured as part of the overall system profiling [5].

A minimalistic security assessment should address the following 5 questions: (1) what do I want to protect? (2) who do I want to protect it from? (3) how likely is it that an element needs protection? (4) how severe are the consequences in the event of a compromise? and (5) what is the cost of preventing compromise?

As an example, consider infrastructure with a vulnerable protocol between several nodes. An adversary compromising this protocol could potentially have unrestricted access to the nodes. In ERTMS, the EuroRadio layer could be overloaded, triggering the session to be terminated. Conversely, an 'air-gapped interface' between the TCMS and, say, a passenger seat reservation system would have little impact on the safe operation of the train but would have a disruptive effect on the passengers. These risks should be reliably modelled and captured to ensure they are understood and do not affect compliance, e.g., with the NIS Directive thresholds.

Railway system operators and asset owners are now faced with requirements to comply with the EU Network and Information Security (NIS) Directive, ISO/IEC 62443, and now, TS 50701 where there is a large knowledge, experience and skills gap to reach compliance. As a result, some organisations are unable to assure their infrastructure, as they do not understand the risks that exist in their architectures. Malware, for example Stuxnet, leveraged the fact that systems that were meant to be air gapped had data transferred between each other via USB, enabling propagation. Similarly, BlackEnergy [8] used spearphishing to affect a management workstation to pivot and affect other ICS components. Further, the Colonial Pipeline attack in 2021 showed that this vector remains prevalent [6]. These are but a few examples of threats as the result of increased interconnectivity that were not realised, nor through the connectivity of these systems that the risk was evaluated, or truly appreciated. Today, in the mainstream media, the work by INSINIA[1] highlights the point of safety over security when systems were developed, and increased connectivity, e.g., remote monitoring were convenient but the security foresight was lacking. Methodologies, for example, one presented in [4] considers the threats to the railways, defining a process to identify security risks to infrastructure. Using frameworks like this, automated tools for use by ICS owners can be created to support initial analysis by system owners.

In the tool[2] presented in this paper, we model components of a given system as a graph and the data flows between components. We also consider how data changes as it passes through various systems, e.g. location data may be converted into safety-critical data after passing through a number of nodes. Moreover, tracking the datatypes (a representation of the data that is transferred between nodes) and link types (representing the properties of the point-to-point connections) allows the asset owner to generate a representative model of their system and simulate different adversary models (e.g. an inside threat, a compromised workstation or an attack to the signalling centre). Using this foundation, we can specify key assets to assess as 'targets' to the attacker, modelled as having entry points at any point in the model with various capabilities. The tool, developed with a Visio frontend, is able to take a graphical model created by the asset owner, representing the probability space and supplemented with XML-like attributes of nodes and edges in the model, subsequently find paths through the model between assets, assessing the compositional security that is offered. Using this model and path-finding capability, the tool returns attack paths and the probability of an exploit being successful on that path, highlighting some potentially interesting paths that were not previously considered or identified.

2. The SCEPTICS Modelling Tool

This section presents a detailed description of the SCEPTICS framework, starting with its inputs, its computation method and the way that security values are assigned to system components. We will then present specific details related to our implementation, focusing on important design choices.

2.1 The Modelling Tool Input

In order to assess the security of a system, our tool requires three inputs: (1) a *system graph* describing the individual components, data flows of a system, and the way these interact with each other; (2) an *adversarial model* describing the capabilities of the attacker and; (3) a *list of assets* to be targeted by the attacker. From these inputs, it is possible to identify the most likely paths that particular adversaries might use to attack a system, and, thus, identify the most vulnerable elements in the system and assess the impact of compensating controls. The level of detail is defined by the asset owner, and controlled through the defined list of assets.

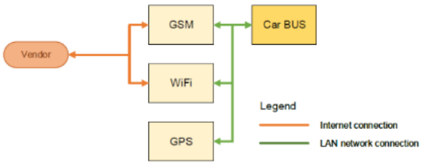

Figure 1: Toy example model based on Figure 2. The GSM, WiFi and GPS nodes refer to the appropriate assets used to provide a service, where a link between the WiFi node and Vendor represents a data link (e.g. 3G/4G).

The System Graph is structured as a directed graph, describing the architecture of the physical system to be analysed by the tool. The individual components of the physical system are nodes within the graph, and connections between them are edges. As an example from Fig. 1, the GSM and WiFi components are modelled as nodes. The Car BUS has also been modelled as a node, but is shown differently to the other nodes, demonstrating how components can be defined in a granular, or generic way. As shown in Fig. 2, the Car BUS is exploded into its constituent components, allowing the modeller to capture the shared security features of this bus. Other internal connections, like the one between the GSM and Car BUS nodes are modelled as edges,

[1] http://www.theregister.co.uk/2018/06/18/physically_hacking_scada_infosec/

[2] A copy of the full model presented in this paper and the SCEPTICS tool, with accompanying documentation, is available to download from https://github.com/sceptics-tool/sceptics-tool

enabling a finer-grained security assessment of the specific link. Each node has a specific identifier, descriptive name and a list of labels describing its supported datatypes, e.g., standardised protocols (e.g. NTP, TCP), or custom-defined types (e.g., geographical data, authenticated data). Additionally, each node has a data profile, which has a datatype, link type (similar to datatype but refers to the physical medium e.g., ethernet or radio), and a CVSS profile which describes the specific security properties of the link. Datatypes represent the properties of the point-to-point connection. Connections between components are modelled as edges. An edge is uniquely defined by the pair of nodes it links, and has a similar set of properties, and one or more data profiles.

The Adversarial Model describes the capabilities of the adversary. This input is important as it restricts the security evaluation to meaningful adversaries. Each adversary contains a list of attack starting nodes from the graph (i.e., the entry points) and/or exploitable datatypes and link types. The Asset List is similar to the adversary model, with the important difference in that it focuses the analysis on the system components which are to be considered as targets of an attack, i.e., the list of assets deemed critical or where an asset owner wants to understand the reachability of an adversary. As an example, a train operator may want to evaluate the ways in which the vehicle bus can become compromised. The asset can be restricted further to specific datatypes handled by that node through the definition of a list of datatypes.

A CVSS Security Profile provides an accessible, repeatable, and reliable way for the security of an asset to be described, leveraging the Common Vulnerability Scoring System (CVSS) framework[3]. The CVSS framework is extensively used in vulnerability management, including information about the severity of a vulnerability, its environmental impact and the difficulty to resolve. CVSS scores are composed of three parts: the base score - how the component may be attacked (e.g., network or physical access), the attack complexity, privileges required, and its impact on confidentiality, integrity, and availability, among other factors; the temporal score – describing whether exploits are in the wild or if there are theoretical attacks, and how these threats may be remediated and; the environmental score - how the compromise of that specific asset will affect neighbouring, linked, components and what the requirements are and the impact, if compromised, on these components.

2.3 The Computational Model

From the SCEPTICS framework, attack vectors are modelled and their security values computed. The framework uses a directed graph, providing an expressive, mathematical model for computation. The asset owner's models can be directly mapped to this mathematical structure, where links are converted to nodes. Where a node handles more than one datatype, we spit it into individual nodes for each datatype, with the CVSS profile replicated across these nodes. Bridges within datatypes result in a link being created, representing the conversion of datatypes within the node. As an example, a node bridging sensor and control data would be split into two nodes, one for sensor data and the other for control data, with a link between them. If it did not bridge between these two datatypes, no link between the two nodes would be created during the transformation step.

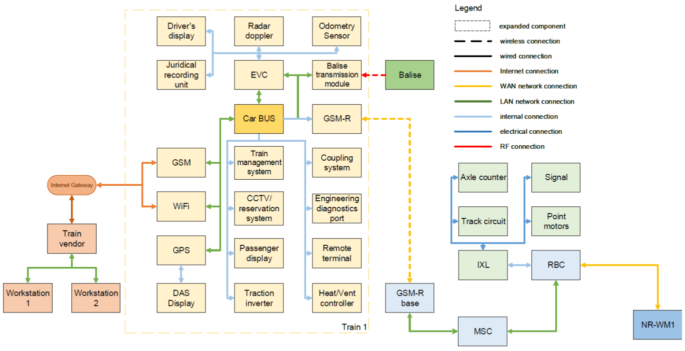

Figure 2: Model of ERTMS, with a train expanded into its constituent components.

The SCEPTICS framework uses probability theory to define the likelihood of a particular edge 'existing' between two nodes, that is, whether an edge is vulnerable to an attack. We can therefore define an attack to be a collection of paths in the system's graph starting at some node through to the attacker's target node. It is important to note multiple paths may exist between the starting node and the target node, where, for each

[3] https://www.first.org/cvss/specification-document

path, the probability of it existing is defined by the application of the inclusion-exclusion principle, enumerating each possible combination of edges and evaluating the likelihood of successful exploitation at each stage.

2.3 SCEPTICS Tool Design

Here, we discuss the tool implementation, highlighting some design choices and optimisations. As previously stated, the tool uses directed graphs, with start and end nodes to determine the possible probabilistic attack vectors, and paths from the start to end nodes. We use breadth-first search to discover these paths, which is efficient and supports 'routing'-like functionality. Given that system models are usually complex and contain many layers of information (i.e., a connection can contain details about the protocols used, the physical medium, purpose of data, etc.), performing accurate and meaningful security computations on these models is non-trivial. For example, to obtain an accurate analysis of the system presented in Fig. 1, one would need to decompose the system into 12 independent graphs (i.e. two medium types: wired, wireless and 6 connection types), and apply the computation method outlined earlier for each graph. This method, however, while somewhat accurate would still not capture any interaction between these connections. What this means is that nodes are appended to a path only if they both support a specific datatype and are connected by an edge that also supports that datatype. To capture the interaction between layers, we use the bridge elements of the graph nodes. These describe the capability of a node to convert data from one layer (e.g. Wired connection) to another. A path can, thus, can be comprised of several shorter paths, each belonging to a different layer, if they share bridge nodes that connect these layers. For example, in Fig. 2, the path [GSM-R]->[GSM-R base]->[MSC] is a valid path across wired (continuous line) and wireless (dotted line) connection types if the [GSM-R base] node is defined as a bridge between the two. The same path is also a valid across the WAN network connection (yellow line) and LAN network connection (green line) and if the [GSM-R base] node is configured as a bridge between them.

3. Analysis Profiles

The modelling tool presented in this paper enables system owners to carry out a number of tasks and analyses. The first and foremost type of analysis carried out by the tool is a discovery of all paths to key assets defined by the asset owner. In addition to path discovery, the tool can carry out alternative analyses including:

'All Roads Lead to' analysis: Assets in the graph can be considered `bastions' which should be protected from attack. In our tool, paths terminating at the specified assets are assessed, and a list returned of the paths that may be taken by an attacker which exceed some predefined threshold. As a concrete example, an RBC uses a number of direct sources of information before making a safety critical decision. However, indirectly-connected systems may have a more pronounced impact on the system than that of the directly connected systems.

Patient-Zero analysis: When a node which becomes compromised, it is important to understand the effects that an attack has on a system. Given a specific starting node, the tool will follow paths from that node outwards, identifying paths that have a likelihood of being successfully compromised over a given threshold. This allows the asset owner to appreciate how an attack can propagate in ways that they had not identified.

Testing New Strategies: ICS owners, given the architecture of their system, may wish to model and assess new strategies to better inform their security review process. By amending the CVSS vectors in the graph, or override the probability values, the tool identifies where changes are best made, and develop future strategies.

Core to the tool is its path discovery capability, carry out probabilistic analyses using the Inclusion-exclusion principle to assess how an attack may propagate throughout the model. This path discovery enables the asset owner to understand all the `via' steps that an attacker may use as part of their cyber kill chain to compromise of one of the asset owner's identified critical assets.

4. Applying the Integrated Framework to ERTMS

The architecture of ERTMS as an inter-connected system with diverse protocols and standards provides a strong basis to test the SCEPTICS tool and framework, where the model in Figure 2 is assessed. This model represents a typical ERTMS and train architecture where each part of the model can be as granular as required. To demonstrate this, we have decomposed the train into its constituent components. As shown in Fig.2, we focus on individual flows, where one node represents a single unit, e.g. a single passenger display. For each node and link in the model, a corresponding CVSS Profile was assigned, based on published work and interviews with rail experts. As input to the model, we specified a set of assets that the attacker would start their attack from, that is, Workstation1, GPS and the Car BUS. For the set of assets that are to be assessed for exploitability, we specified the targets to be the EVC, RBC and Car BUS.

4.1 Selection of Attacker Entry Points

For the set of attacker entry points (the adversarial model defined by the asset owner) above, we chose these specifically as they have differing interdependencies in the rail network. Balises in ETCS Level 2 and 3 are trusted by the train to provide accurate location and track profile information, which are relayed to the RBC such that it can make safety-critical decisions related to train position (e.g., Movement Authorities). In the event a train reported an incorrect location, with no external verification, the train following would be given an overlapping movement authority. The train also relies on balise data for track profile information, e.g. tilting parameters, speed restrictions and gradients. If compromised, the train may be placed in an unsafe situation, as there is limited validation of this data. Remote Condition Monitoring enables train operators and maintainers carry out real-time, predictive, monitoring of rolling stock fleets. The train data sent via this mechanism allows the vendor to plan preventative maintenance when the train returns to the depots, in addition to enabling remote triage of issues whilst the train is in service. As such, tampering with the data relayed to the vendor could lead to the train being withdrawn. Furthermore, if the train is remotely managed, the safe operation of the train could be affected, as an attacker may carry out reconnaissance on the proprietary systems of the train. We explore the engineer workstations at the vendor as a possible entry point for the attacker in the model. In our model, we have two workstations, Workstation1, which has been compromised, and Workstation2, which is unaffected.

4.2 Defining Target Assets

As target assets for the model[4], we chose systems which have either high safety requirements, or ones that, if compromised, would cause significant disruption. The *European Vital Computer (EVC)* is the train's on-board system for ERTMS, controlling train supervision. If a *balise* provided inaccurate information, the EVC would pass this to the RBC, potentially returning an unsafe decision. The EVC also is responsible for on-board operations, where, if the balise reported an incorrect linespeed, there would be a risk to life. The *Car BUS* is a bus that runs down the length of the train. In modern rolling stock, it carries a variety of data, e.g. power and braking commands. If an attacker could insert data here, they would be able to control a number of train functions.

4.3 Results

Using the model shown in Fig.2, and the assets and entry points in this section, we obtain the following results:

```
Adversary a1:

[Balise]->[Balise transmission module]->[EVC]:0.32736

[Balise]->[Balise    transmission    module]->[EVC]->[GSM-R]->
    -> [GSM-R base]->[MSC]->[RBC]:0.00016

[Balise]->[Balise    transmission    module]->[EVC]->[GSM-R]->
    ->   [GSM-R   base]->[MSC]->[RBC]->[MSC]->[GSM-R   base]->
    -> [GSM-R]->[EVC]:0.32747
```

```
Adversary a2:

[Workstation1]->[Train      vendor]->[Internet      Gateway]->
    -> [WiFi]->[Car BUS]:0.13456

[Workstation1]->[Train      vendor]->[Internet      Gateway]->
    -> [GSM]->[Car BUS]:0.03275

[Workstation2]->[Train      vendor]->[Internet      Gateway]->
    -> [WiFi]->[Car BUS]:0.0313

[Workstation2]->[Train      vendor]->[Internet      Gateway]->
    -> [GSM]->[Car BUS]:0.00762
```

The results are given as probabilities that the chain could be successfully exploited, i.e., the likelihood an attacker would be successful in breaching each component from the starting node to the `critical asset'. A high probability would indicate that an attacker would be more likely than not to succeed, and should be used as a starting point for improvements by the asset owners. The attacks that are returned by the tool are likely and real attack vectors to the railway, confirmed by rail experts by testing our tool, validating our results. We observe that for Adversary *a1*, an attacker who has a Balise as an entry point, they have a probability of *0.32736* of successfully reaching and affecting the EVC on the train, e.g. issuing invalid line speeds, or convincing the train that it is in a different location. We note, however, that when we consider the attack affecting an RBC, the probability reduces to *0.00016*, but the return path, where the RBC has made a decision, has a much higher probability. This is because the balise location data flows to the EVC, and reported to the RBC, where the RBC will convert, i.e. bridge the data into command data, which flows back to the EVC. In existing tools, they would terminate analysis at the

[4] A longer version of this paper, assessing other adversaries and target nodes is available at http://research.rjthomas.io/wcrr2022-sceptics

RBC, whereas our tool identifies this data conversion at the RBC, as the tool supports many datatypes for nodes and links, and is hence able to find this path. For Adversary *a2*, however, the results are interesting – the compromised engineer workstation, *Workstation1* has a much higher probability of successfully affecting the Car BUS than a secure Workstation, *Workstation2*. In reality, if an engineer could remotely manage a train, this could be a potential threat, and highlights where good security practices are key to minimise exposure.

4.4 Testing New Strategies

From the results above, the balise has a high probability of influencing the RBC, where a probability of 10^{-6} is considered the safety limit [10]. Using our tool, system owners can also simulate improvements to component security rather than just evaluating them. As an example, replacing the Balise in Fig. 2 with a new Balise, *SecureBalise,* the simple addition of a MAC to the Balise payload prevents an attacker from setting their own payload without knowing the balise-derived key, reducing the attack likelihood from *0.32736* to *0.06591* for the path from the Balise to the EVC. This demonstrates how our tool can inform security decision-making. Another solution would be to send partial balise data to the RBC for validation. This would give a similar CVSS profile to the *SecureBalise* node as the attack complexity has increased, requiring additional capabilities, yielding similar probabilities to *SecureBalise*. Alternative strategies, e.g., moving the balises to a operating mode similar to ETCS Level 1 (where the balise is connected to the RBC) can also be tested, although, would have significant costs.

5. Conclusion

We have presented a modelling tool and methodology which enables ICS owners to assess the security of their infrastructure, and allow them to carry out `first-steps' remediation. This is done through probabilistic analysis, path discovery and leveraging the CVSS framework to provide insights that are not possible in complex ICS architectures. The tool has further applications outside of the ICS sector, e.g., modelling corporate systems security and allows asset owners to reason about the security of their system with confidence, as well as providing assurances of the security in their systems.

Acknowledgment: Funding for this paper was provided by the UK's Centre for Protection of National Infrastructure (CPNI) and the Engineering and Physical Sciences Research Council (EPSRC) via the SCEPTICS: A SystematiC Evaluation Process for Threats to Industrial Control Systems project and the National Cyber Security Centre UK (NCSC UK), Research Institute in Trustworthy Inter-Connected Cyber-Physical Systems (RITICS) and the UK Rail Research and Innovation Network (UKRRIN).

References

[1] Braband, J. 1997. Safety and Security Requirements for an Advanced Train Control System. Safe Comp 97. Springer. 111–122.

[2] Checkoway, S. et al. 2011. Comprehensive Experimental Analyses of Automotive Attack Surfaces. Proceedings of the 20th USENIX Conference on Security (USA, 2011), 6.

[3] Chothia, T. et al. 2017. An Attack Against Message Authentication in the ERTMS Train to Trackside Communication Protocols. Proceedings of the 2017 ACM AsiaCCS (New York, NY, USA, 2017), 743–756.

[4] Evans, R. et al. 2016. SCEPTICS: A Systematic Evaluation Process for Threats to Industrial Control Systems. (2016).

[5] Hawthorn, A. 2017. A Proven Approach to Requirements Engineering – The Why, What and How of REVEAL.

[6] Hobbs, A. 2021. The Colonial Pipeline Hack: Exposing Vulnerabilities in U.S. Cybersecurity. SAGE Publications: SAGE Business Cases Originals.

[7] Koscher, K. et al. 2010. Experimental Security Analysis of a Modern Automobile. 2010 IEEE Symposium on Security and Privacy (2010), 447–462.

[8] Lee, R.M. et al. 2016. Analysis of the cyber attack on the Ukrainian power grid. Electricity Information Sharing and Analysis Center (E-ISAC). (2016).

[9] Lu, J. et al. 2015. Time-Memory Trade-off Attack on the GSM A5/1 Stream Cipher Using Commodity GPGPU. 13th International Conference on Applied Cryptography and Network Security (ACNS 2015) (2015).

[10] Wolff, J. 2007. What is the Value of Preventing a Fatality? Risk: Philosophical Perspectives. T. Lewens, ed. Routledge.