Hindawi
Security and Communication Networks
Volume 2018, Article ID 5634181, 9 pages
https://doi.org/10.1155/2018/5634181

Research Article

A Framework for Vulnerability Detection in European Train Control Railway Communications

Irene Arsuaga,[1] **Nerea Toledo**,[1] **Igor Lopez**,[2] **and Marina Aguado**[1]

[1]*Department of Communication Engineering, University of the Basque Country (UPV/EHU), Alameda Urquijo s/n, 48013 Bilbao, Spain*
[2]*Research & Development Department, Construcciones y Auxiliar de Ferrocarriles (CAF), J. M. Iturrioz 26, 20200 Beasain, Spain*

Correspondence should be addressed to Nerea Toledo; nerea.toledo@ehu.eus

Received 17 November 2017; Accepted 22 February 2018; Published 15 May 2018

Academic Editor: Prem Mahalik

Railway systems have evolved considerably in the last years with the adoption of new communication technologies. Aiming to achieve a single European railway network, the European Rail Traffic Management System (ERTMS) emerged in Europe to substitute multiple and noninteroperable national railway communication systems. This system and its security strategies were designed in late 1990s. Recent works have identified vulnerabilities related to integrity, authenticity, availability, and confidentiality. In the context of defining effective countermeasures to mitigate potential vulnerabilities, these vulnerabilities have to be analysed. In this article we introduce a framework that attempts to challenge ERTMS security by evaluating the exploitability of these vulnerabilities.

1. Introduction

The increased needs of transportation in a common market and the lack of interoperability between different railway operational styles in Europe brought up the need for a common rail management system in Europe. In the mid-1980s, the railway community began to search for a common European operation management for railways, called European Rail Traffic Management System (ERTMS) [1]. This solution was created to substitute the heterogeneous national train control landscape scenario.

The ERTMS communications are radio based communications and, thus, wireless systems are used to transmit the movement authorities from the Radio Block Centre (RBC), the entity in charge of managing trains operation, to the trains. Up to now, the wireless communication technology in use is the GSM-R (Global System for Mobile Railways), a specific version of GSM devoted to railway communications.

In order to guarantee the security of the communications, the GSM-R network has to ensure several security properties. On the one hand, the data transmitted should be kept confidential. Moreover, the data should not be changed by an attacker before arriving to the train, to ensure that the train does not receive fake movement authorities. On the other hand, the communication network should be always available for the exchange of needed messages. That is, the network has to ensure the CIA triad: confidentiality, integrity, and availability, which is a widely known model designed to guide information security policies within an organization and represents the most crucial security properties. Since safety is one of the most critical issues to be addressed in the railway context, in this work we focus on train movement authorization message exchanges, so our primary concern is data integrity, even if availability and confidentiality will also be affected.

With the goal of guaranteeing data confidentiality and integrity, different encryption systems are used in the different communication layers. For GSM and also for GSM-R, A5/1 encryption system is used. In addition, the EuroRadio protocol is used to ensure the authenticity and the integrity of the communications. However, it has been proved that both protocols have vulnerabilities [2, 3].

Finally, it should be taken into account that radio jamming devices could jam, block, or interference wireless communications, being able to break the availability of the network.

Apart from the aforementioned lack of security of GSM-R and EuroRadio, it is necessary to point out the evolution of the railway communications in the last years. Although in recent past railway systems were close systems, recently, a new trend of connecting all the elements of the railway network to the Internet is getting relevance. This fact results in exposing railway communications to intrinsic vulnerabilities of Internet and, thus, challenging the security in these scenarios.

Due to the easiness of not fulfilling the security properties defined in the CIA triad, the railway context can be considered a hostile environment and, hence, safety and security are demanded in these networks.

Many efforts have been done to provide safety in the railway scenario [4–6], but security is an emerging demand, and even if different research efforts have also been done for providing secure railway communications [7, 8], there are still several limitations. Moreover, methodologies used in safety analysis, based on probabilistic hazards, are not valid for it. That is, it is not feasible to calculate when an attacker will detect a vulnerability and/or exploit it. Therefore, it is necessary to design a vulnerability detection system, in order to try to avoid the exploitation of those vulnerabilities by means of defining countermeasures.

Our contribution focuses on presenting a framework that will be able to exploit the vulnerabilities of the ERTMS system regarding integrity and authenticity. By means of this framework, it will be possible to know if this attack can be done in real time or not.

The article is organized as follows; in Section 2, we describe the ERTMS protocol and analyse why the railway context is a hostile environment. In Section 3, we analyse the work done relating to this topic. We describe our framework in Section 4, emphasising the benefits of having a framework that attacks different vulnerabilities, describing it, with the limitations that it has, and finally describing the process we will follow to know if the described attack could succeed in real time; then we conclude in Section 5.

2. Overview of ERTMS

The ERTMS is composed of two elements: (1) the European Train Control System (ETCS) for the signalling and (2) the GSM-R for the communication.

2.1. ETCS. The ETCS has a great variety of possible configurations in the signalling equipment used on the existing or new lines. Because of this, ETCS has been conceived with several application levels: 0, NTC (national train control), which is the former STM (specific transmission module), 1, 2, and 3. Next, the different ETCS levels are described.

2.1.1. ETCS Level 0. ETCS level 0 covers the operation of ETCS equipped trains on lines that are not equipped with ETCS or national systems. On this lines, lineside signals are used to give movement authorities to the trains. This level has been defined to ensure the proper transition between ETCS equipped and nonequipped trains. The operation of this level is shown in Figure 1.

2.1.2. ETCS Level NTC. ETCS level NTC is used to run ETCS equipped trains on lines equipped with national train control and speed supervision systems. The train control information that is generated trackside by the national train control system is transmitted to the train via the communication channels of the underlying national system and transformed onboard into information interpretable by ETCS. Depending on the functionality and the configuration of the specific national system installed onboard, the ERTMS/ETCS onboard system may need to be interfaced to it, in order to perform the transitions from/to the national system and/or in order to give access to ERTMS/ETCS onboard resources. This can be achieved through a device called STM. The operation in this level is presented in Figure 2.

2.1.3. ETCS Level 1. In the application ETCS level 1, ETCS is overlaid to the traditional signalling equipment. The train position is detected by the traditional trackside devices, which are linked to the interlocking through the interface Lineside Encoder Unit (LEU). The interlocking is the wayside equipment control. Lineside signals are kept, and data is transmitted to the onboard equipment by means of Eurobalises, which are transponders placed between the rails of the railways. The operation of this level is shown in Figure 3.

2.1.4. ETCS Level 2. In application level 2, GSM-R radio is used to exchange data between the RBC and the trains. EuroRadio protocol is implemented in these communication channels, which is based on a 3DES cryptographic system. Movement authorities to the trains are sent via this channel, and besides a continuous speed supervision is made. For this communication, the Base Transceiver Station (BTS) of the Control Centre communicates with the onboard unit (OBU) of the onboard equipment.

However, the train detection is performed by the trackside equipment, so it is out of the scope of ERTMS/ETCS. In this level, lineside signals could be suppressed. The operation is described in Figure 4.

2.1.5. ETCS Level 3. Finally, the operation level 3 is a radio based train control system. Movement authorities are generated trackside and transmitted to the train via EuroRadio, as in level 2, but in this level, train position is also performed by the trackside RBC. Eurobalises are just used for location referencing. Lineside signals could be suppressed in this level too. The operation is described in Figure 5.

2.2. GSM-R. During the course of their standardization activities, the UNISIG group realized that in order to ensure security of the railways in GSM, certain spectrum bands needed to be allocated. However, GSM could not fulfil all the requirements needed for an efficient railway service, and therefore, some specific functional features were added to the GSM specifications.

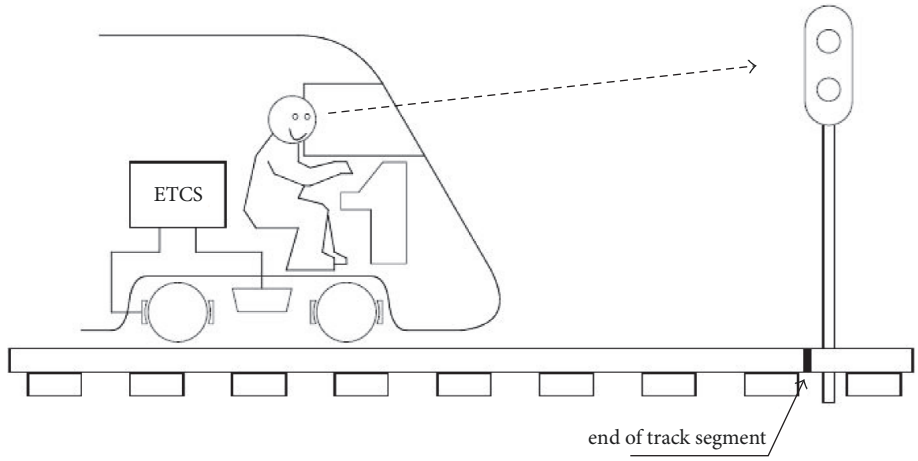

Figure 1: ETCS level 0 operation [9].

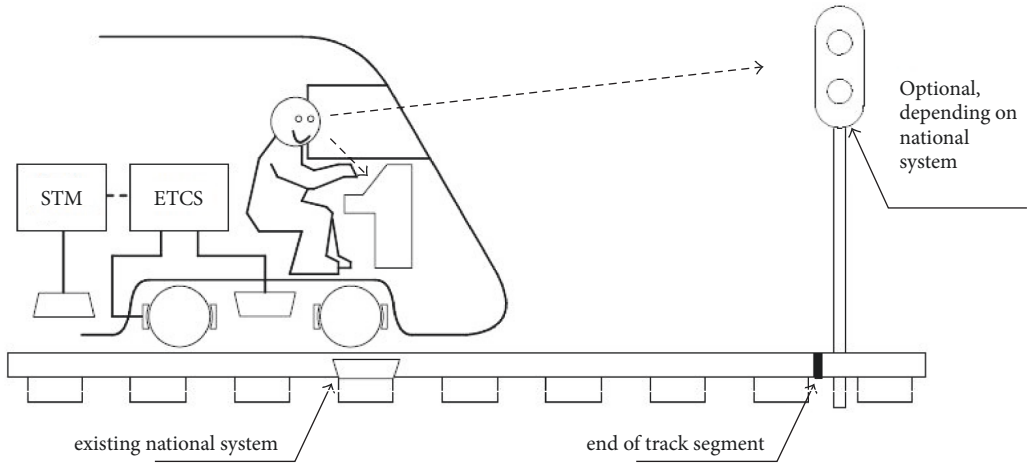

Figure 2: ETCS level NTC operation, based on [9].

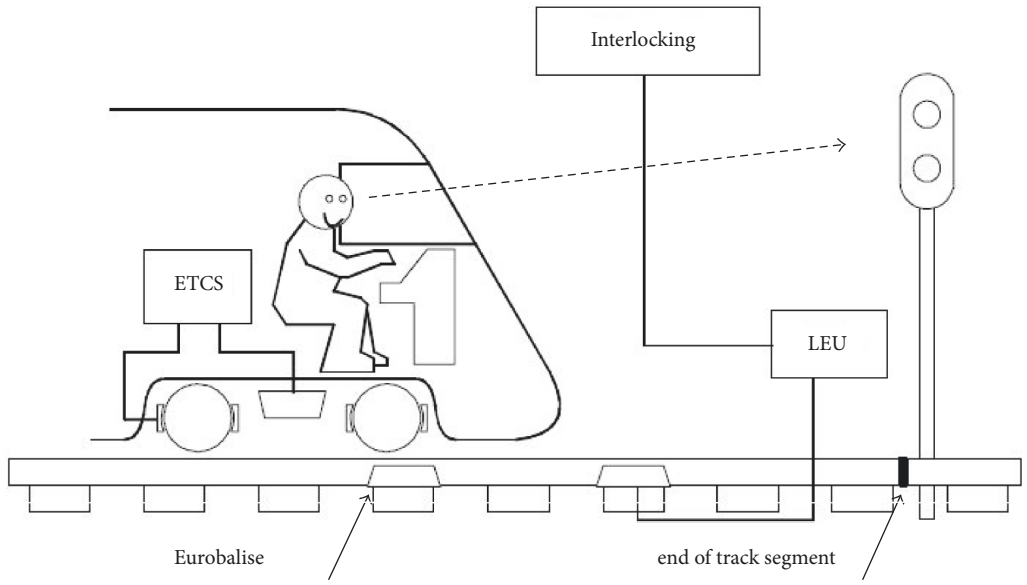

Figure 3: ETCS level 1 operation [9].

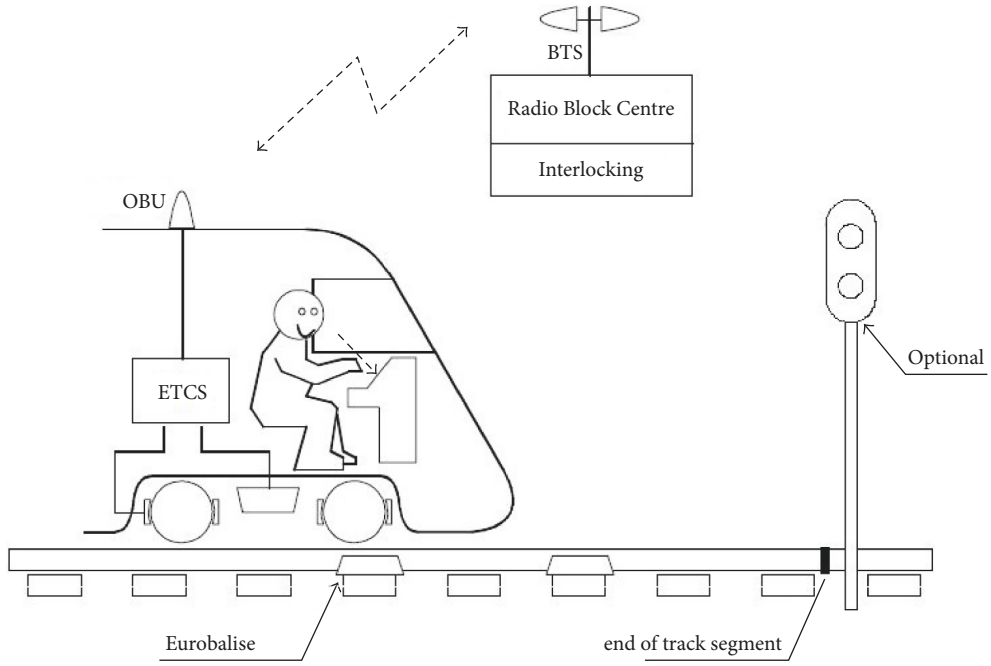

Figure 4: ETCS level 2 operation, based on [9].

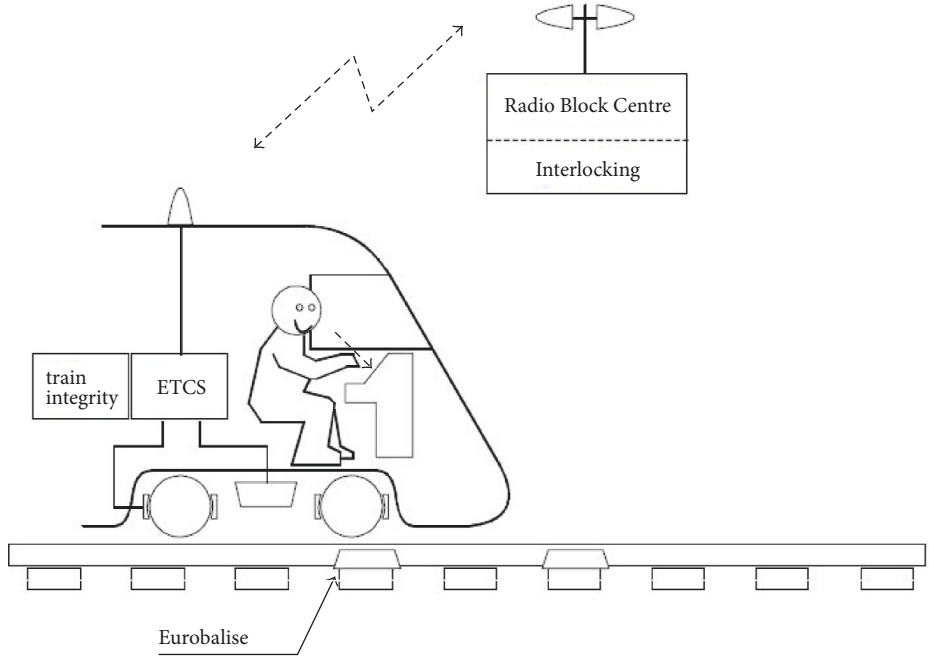

Figure 5: ETCS level 3 operation [9].

The frequencies allocated in Europe for GSM-R are close to the GSM 900 band of the public operators. A 4 MHz spectrum with 19 frequencies is available for these communications.

In order to guarantee the confidentiality of the network, the A5/1 stream cipher is used in GSM (and GSM-R) networks. A stream cipher is a symmetric key cipher where plain-text digits are combined with a key stream.

2.3. Integrity and Authenticity in ERTMS. From the second level ETCS, integrity and authenticity in ERTMS are accomplished with two different security mechanisms: A5/1 for GSM-R and EuroRadio for ETCS. This is the target level of our framework.

Regarding A5/1, even if at the beginning the encryption system was kept in secret, it became public knowledge through leaks and reverse engineering [2]. A number of

serious weaknesses in the cipher were identified [10]. Hence, rainbow tables able to decrypt encrypted messages are available on the Internet.

On the other hand, the EuroRadio protocol uses 3DES keys to encrypt the messages [11]. The keys used in the communications (KTRANS, K-KMC, KMAC, and KSMAC) are created by the KMC entity, with the exception of the session key, KSMAC. The KTRANS and K-KMC keys are transport keys used to ensure the safe distribution of the KMAC keys from the KMC to ERTMS entities. This distribution is made off-line; this means it requires personnel to manually deliver the messages.

The KMAC key is used in the session establishment process to negotiate the KSMAC session key between ERTMS entities. Three messages are exchanged between the ERTMS entities in this phase for the authentication of both entities and the key generation. In these messages, R_A and R_B random numbers are sent, which are used for computing the KSMAC key together with the KMAC key. Considering KSMAC = $K_S = K_{S1}, K_{S2}, K_{S3}$, the three 64-bit DES keys K_{S1}, K_{S2}, and K_{S3} are calculated according to the following formulas:

$$\begin{aligned} K_{S1} &= \mathrm{MAC}\left({R_A}^L \mid {R_B}^L, K_{AB}\right) \\ &= \mathrm{DES}\left(K_3, \mathrm{DES}^{-1}\left(K_2, \mathrm{DES}\left(K_1, {R_A}^L \mid {R_B}^L\right)\right)\right) \\ K_{S2} &= \mathrm{MAC}\left({R_A}^R \mid {R_B}^R, K_{AB}\right) \\ &= \mathrm{DES}\left(K_3, \mathrm{DES}^{-1}\left(K_2, \mathrm{DES}\left(K_1, {R_A}^R \mid {R_B}^R\right)\right)\right) \\ K_{S3} &= \mathrm{MAC}\left({R_A}^L \mid {R_B}^L, K'_{AB}\right) \\ &= \mathrm{DES}\left(K_1, \mathrm{DES}^{-1}\left(K_2, \mathrm{DES}\left(K_3, {R_A}^L \mid {R_B}^L\right)\right)\right). \end{aligned} \tag{1}$$

where L means left and R means right, and therefore, $R_A = {R_A}^L \mid {R_A}^R$ and $R_B = {R_B}^L \mid {R_B}^R$.

The calling entity (B) creates a R_B random number and sends it to the called entity (A) in plain text. Consequently, the A entity creates a R_A random number and computes the KSMAC key with both random numbers and the KMAC. Afterwards, the A entity sends the created R_A random number and a CBC-MAC code computed with the KSMAC and both random numbers to the B entity. Finally, the B entity computes the KSMAC key and verifies that it is correct, creating also a CBC-MAC with it, which is sent to the A entity, for the complete authentication. Random numbers are exchanged in plain text and, thus, an attacker could save them.

3. Related Work

A lot of research and innovation projects are being completed with the funds of the European Union regarding the cybersecurity in railways. Some of those projects are described in [12].

Different analyses of the ERTMS protocol's security have also been done. These analyses have pointed out vulnerabilities that the ERTMS cryptographic mechanisms have. A high-level security analysis of ERTMS is made in [13] but does not present the vulnerabilities of the EuroRadio protocol that will be exploited with this framework. Different vulnerabilities of the EuroRadio protocol are pointed out in [3] by performing an analysis of it with the ProVerif tool. These vulnerabilities include, for instance, the ability of including high-priority messages or deletion of messages, since the session establishment process does not use timestamps and, therefore, these messages could be replicated. In this case, once the session is established, the train does not verify the identity of the RBC anymore, so a vulnerability that could be exploited exists.

Additionally, [14] pointed out that since the distribution of the KMAC key is made off-line and this requires personnel to manually deliver the keys from the KMC to the ERTMS entities, many operators decide to simplify the process by using the same KMAC for large train fleets, amplifying the risk of having an attack. Therefore, if the attack is performed during the session establishment process and the same key is shared between different parties, the whole system could be compromised: an attacker could take the identity of many trains in other session establishments.

On the other hand, [15] pointed out the ability of making a key collision attack to DES and [16] described how a Related-Key Attack (RKA) can be done in ERTMS. A method for doing these two attacks in ERTMS networks is presented in [17] and concludes that the EuroRadio protocol is not secure if large amounts of data and, therefore, long session lengths are used. Thus, the Meet-in-the-Middle attack presented in this article could be more feasible.

However, all of these analyses present vulnerabilities of the protocols used in ERTMS but do not describe how these vulnerabilities could be exploited in order to later find countermeasures for those vulnerabilities. This paper will contribute by presenting a framework that describes how an attack could be performed and figuring out if it is feasible to do it in real time.

4. Proposed Method for Vulnerability Detection

In this section, we present our framework and the method for vulnerability detection describing also its limitations. Finally, the process we will follow to know if an attack could be successful in real time is described.

4.1. Description of the Framework and Limitations. The scenario that this framework will consider is shown in Figure 6. The train is used to connect to the Control Centre in order to receive movement authorities, but first, the train sends a position report to the Control Centre.

In the scenario that we are considering, we force the train to connect to the malicious Control Centre, instead of the real one, but before doing this, we calculate the keys used in the communication between the train and the RBC, with the attacker presented in Figure 7.

Once we have gotten the keys and forced the train to connect to our malicious Control Centre, the position report

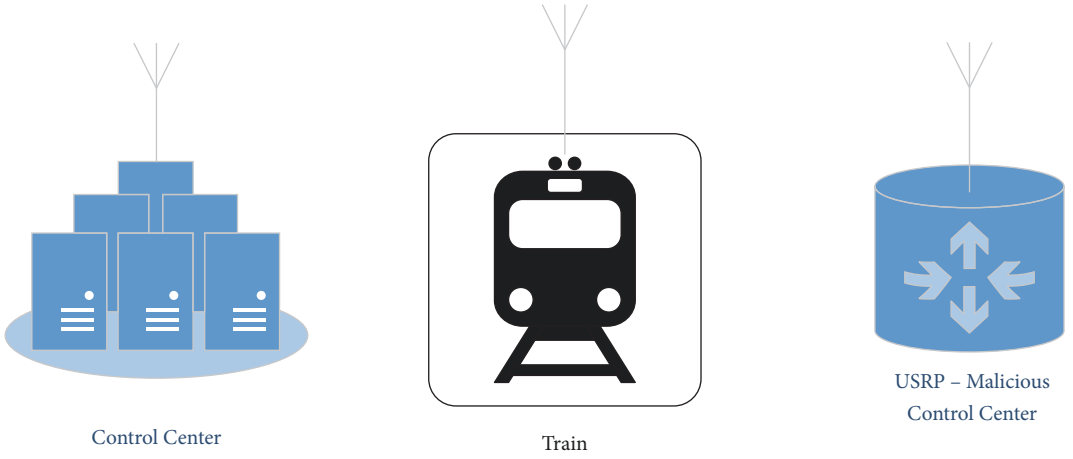

Figure 6: Scenario.

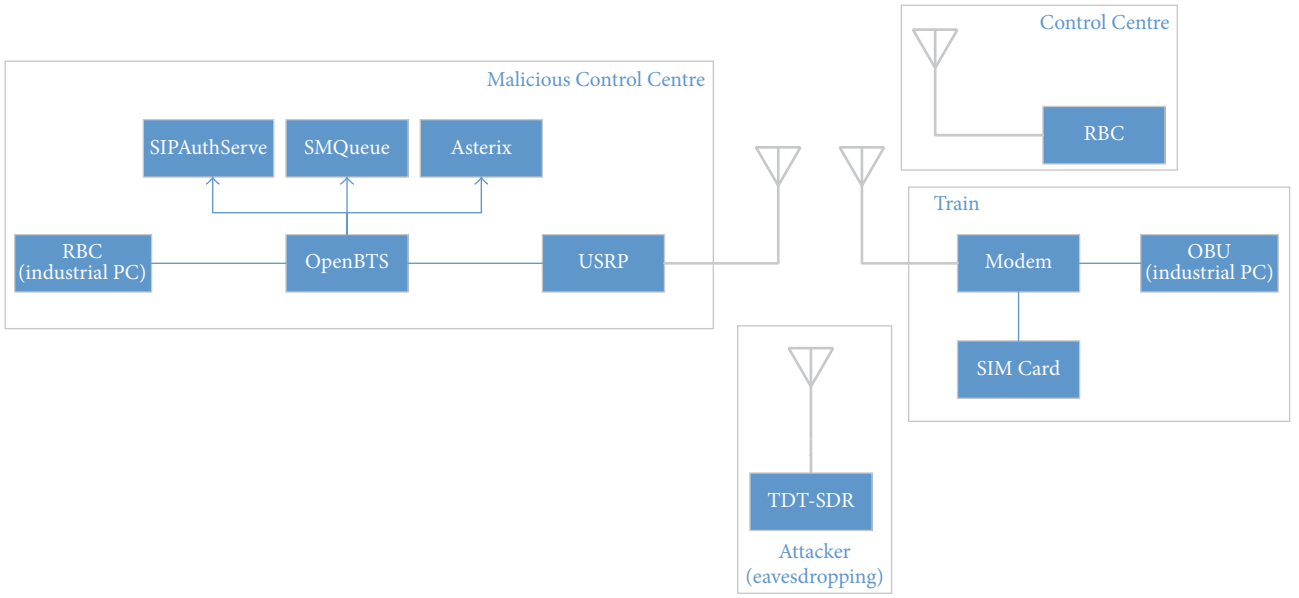

Figure 7: Framework.

that the train sends will arrive to the malicious Control Centre. This position report is encrypted by different encryption systems in different levels, but in this point, we have already obtained the keys used in the communication and, therefore, we are able to decrypt the message. Accordingly, we are able to change the position report and send it to the real Control Centre, by taking the identity of the train. The Control Centre receives the fake message and creates a movement authority depending on the position report, which is sent to the train.

The framework we have considered is described in Figure 7. It is composed of the malicious Control Centre, the train, an attacker that will eavesdrop, and the real Control Centre.

The malicious Control Centre is formed by the false RBC, which will be an industrial PC; the OpenBTS (Open Base Transceiver Station) software [18], which is composed of the SIPAuthServe, SMQueue, and Asterix servers; and the USRP N210 Software Defined Radio (SDR) [19].

The OpenBTS is a software-based GSM access point, allowing standard GSM-compatible mobile phones to be used as SIP endpoints in Voice over IP (VoIP) networks. The software controls the transceiver, makes calls, and sends SMSs. The SIPAuthServe is the server that processes SIP register requests that OpenBTS generates when a handset attempts to join the GSM network. It supports three types of authentication:

(i) AUTH type 2: unauthenticated. The handset is connected to the OpenBTS network but it does not exist in the register server.

(ii) AUTH type 1: cached authentication. The handset is connected to the network and it does exist in the

register server, but a simple encryption method based on TMSI is used.

(iii) AUTH type 0: full authentication. The SIM card is full authenticated in OpenBTS and, therefore, K_i key is provided to the authentication server and used for the encryption. The K_i is a 128-bit value used to authenticate the SIMs on a GSM mobile network. Each SIM holds a unique and secret Ki assigned by the operator. This authentication method uses proper GSM encryption over the network.

The other servers in OpenBTS are SMQueue and Asterisk, as it has been pointed out before. While the SMQueue processes SIP message requests that OpenBTS generates when a handset sends an SMS, Asterisk is a VoIP switch responsible for handling SIP INVITE requests, establishing the individual logs of the call, and connecting them together [20].

The hardware part of the OpenBTS software is a SDR, USRP N210 in our case, with two GSM antennas to create the network. USRP N210 has been chosen because it supports GSM-R networks and provides high-bandwidth and a high-dynamic range processing capability.

The train in the framework will be simulated with a PC, a Modem, and a programmable SIM card. A programmable SIM card [21] is needed because it is necessary to know the K_i key of the SIM card in order to get the full authentication in OpenBTS and use a GSM encryption over the network. The Modem will be used for being able to connect the PC to the GSM network.

Finally, the attacker that will perform the eavesdropping attack in Figure 7 will be supplied by a TDT-SDR. It is a SDR that captures the traffic in the GSM network together with the Universal Radio Hacker (URH) software. With this SDR we will be able to investigate the wireless protocol, and with the rainbow tables we will be able to get the A5/1 keys used in the communication between the train and the real Control Centre. Thus, since the SIM card we use has a programmable K_i, we will be able to configure the same A5/1 key and, therefore, once we force the train to connect to the malicious Control Centre, we will be able to decrypt the sent messages.

As mentioned before, our framework is composed of OpenBTS and USRP N210 and, therefore, since we use the OpenBTS software, we are able just to create GSM-R and GPRS networks. In the case railway networks evolve to LTE (Long Term Evolution) network, we could use OpenLTE open source project [22], but this project cannot be used in the hardware USRP N210 because of clock incompatibility reasons. Therefore, a sample rate conversion on the host would have to be done in the USRP.

4.2. Specification of the Procedure. The flow chart that this attack follows is described in Figure 8. As can be seen, after installing the framework we are able to get the data from the train to the real Control Centre. This traffic is ciphered by A5/1. Since we have the rainbow tables in the attacker performing the eavesdropping attack, we are able to decrypt the messages on GSM level.

The attack in the EuroRadio protocol will be performed against the DES KSMAC key and will be a Meet-in-the-Middle attack. In this attack, all possible keys are tested.

Since all the messages that are exchanged between the train and the RBC in ERTMS are defined in [11], we assume that we are able to obtain a known plain-text and a cipher-text pair (P_1, C_1). The Meet-in-the-Middle (MTM) attack with in ciphers like $C = \mathrm{DES}(K_1, \mathrm{DES}(K_2, P))$ works as follows:

We build a list containing the pair (I_1, K_1) for every possible value of K_1, 2^{56} for DES. The I_1 values will be gotten by brute force, $I_1 = \mathrm{DES}(K_1, P_1)$.

On the other hand, we will obtain I_2 values by performing $I_2 = \mathrm{DES}^{-1}(K_2, C_1)$. This operation is performed until the I_2 value matches a I_1 value that is stored in the table.

In order to be sure that the computed keys are correct, it is possible to obtain another known plain-text and a cipher-text pair (P_1, C_1) and calculate $C = \mathrm{DES}(K_1, \mathrm{DES}(K_2, P_2))$. If C and C_2 are equal, it means the we have find the correct keys.

In triple-DES systems where there are three different keys, the ciphers work following the next relation:

$$C = \mathrm{DES}\left(K_1, \mathrm{DES}^{-1}\left(K_2, \mathrm{DES}\left(K_3, P\right)\right)\right). \quad (2)$$

For the Meet-in-the-Middle attack in triple-DES with three different keys, we define $\mathrm{DES}(K_2, P) = \mathrm{DES}^{-1}(K_2, \mathrm{DES}(K_3, P))$, so we just need to apply this for the calculation of I_1.

The calculation of I_1 needs 2^{112} operations, because it is a double-application of DES, and on the other hand, the calculation of I_2 needs 2^{56} operations. Thus, the attack requires $2^{112} + 2^{56} \approx 2^{112}$ operations.

Afterwards, if we are able to calculate the three keys we are going to obtain $K_S = (K_{S1}, K_{S2}, K_{S3})$, we need to calculate the time we need for performing the whole attack, as Figure 8 describes. With the measured time, we know whether this attack could be performed in real time or not.

In the case the attack can be performed in real time, that is, if all the keys are obtained during the operation of the train, these keys could be used to carry out different attacks that involve the identity theft of the train or the RBC. The attacker, for instance, could pass himself off as the RBC in order to send false movement authorities to the train. On the other hand, the attacker could also falsify the trains position control that is made by the RBC in ERTMS level 3, by sending false position information to the RBC while he impersonates the train. All this false information created by the attacker could involve the collision between different trains.

The results obtained with the framework will help in looking for countermeasures, since the fact of acquiring the keys in real time means A5/1 and 3DES security mechanisms are not strong enough for railway environments. In consequence, those mechanisms should be enforced or changed in order to continue using ERTMS systems in a secure manner. A possible countermeasure for the system could be to update the 3DES security mechanism to a more secure system such as AES, since AES uses larger block sizes and longer keys. Therefore, it will be more costly to perform the attack in real time. In fact, 3DES keys length is 112 or 156 bits, whereas in AES, the length of the keys is variable: 128, 192, or 256

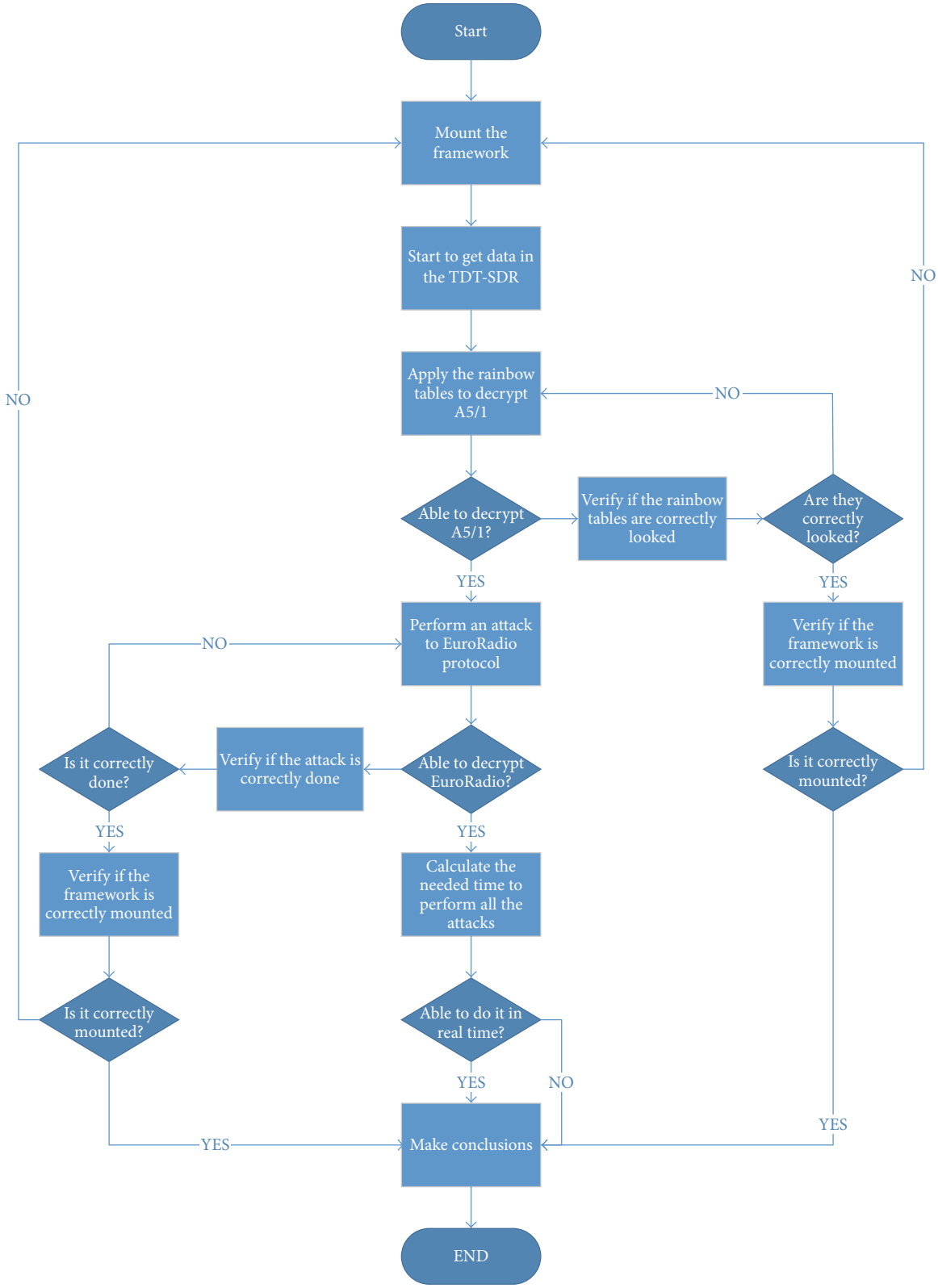

Figure 8: Flow chart of the process.

bits. However, the use of AES should be first evaluated by the framework presented in Figure 7.

5. Conclusions and Future Work

Different vulnerabilities in the security mechanisms of the ERTMS system have been described in this article, but even if they are identified, without an attacking framework we do not know whether they are exploitable in practice or not. In consequence, the presented framework will give information about the exploitability of the A5/1 and 3DES security mechanisms and, therefore, will determine if countermeasures should be applied to improve the security of the system or not. Moreover, the resulting information of the framework will constitute the basis of the countermeasures that should be applied to the system.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research work was supported by the Spanish Government through the SAREMSIG TEC2013-47012-C2-1-R project, Proyectos de I+D+I Retos Investigación 2013, and by the Cyber Security on Rails project with Construcciones Auxiliar de Ferrocarriles, Investigación y Desarrollo, S.L. 2015-2016.

References

[1] U. P. Winter, *Compendium on ERTMS*, Eurail press, 2009.

[2] M. Kalenderi, D. Pnevmatikatos, I. Papaefstathiou, and C. Manifavas, "Breaking the GSM A5/1 cryptography algorithm with rainbow tables and high-end FPGAs," in *Proceedings of the 22nd International Conference on Field Programmable Logic and Applications, FPL 2012*, pp. 747–753, nor, August 2012.

[3] R. J. T. Joeri de Ruiter and T. Chothia, "A formal security analysis of ertms train to trackside protocols," *Reliability, Safety, and Security of Railway Systems - Modelling, Analysis, Verification, and Certification*, pp. 53–68, 2016.

[4] UIC, "2005, UIC Code 518 OR: Testing and approval of railway vehicles from the point of view of their dynamic behaviour: safety, track fatigue, ride quality".

[5] A. Faivre, A. Lapitre, A. Lanusse et al., "Two methods for modeling and verification of safety properties of railway infrastructures," in *Proceedings of the International Conference on Industrial Engineering and Systems Management, IEEE IESM 2015*, pp. 48–54, esp, October 2015.

[6] M. Franeková, K. Rástocn, A. Janota, and P. Chrtiansky, "Safety analysis of cryptography mechanisms used in gsm for railway," *Annals of the Faculty of Engineering Hunedoara*, vol. 9, no. 1, p. 207, 2011.

[7] L. J. Valdivia, I. Adin, S. Arrizabalaga, J. Anorga, and J. Mendizabal, "Cybersecurity-The Forgotten Issue in Railways: Security Can Be Woven into Safety Designs," *IEEE Vehicular Technology Magazine*, vol. 13, no. 1, pp. 48–55, 2018.

[8] G. Hatzivasilis, I. Papaefstathiou, and C. Manifavas, "Real-time management of railway CPS secure administration of IoT and CPS infrastructure," in *Proceedings of the 6th Mediterranean Conference on Embedded Computing, MECO 2017*, mne, June 2017.

[9] U. SUBSET-026-2, "Ertms/etcs system requirements specification chapter 2 basic system description," Tech. Rep., 2014.

[10] C. Manifavas, G. Hatzivasilis, K. Fysarakis, and Y. Papaefstathiou, "A survey of lightweight stream ciphers for embedded systems," *Security and Communication Networks*, vol. 9, no. 10, pp. 1226–1246, 2016.

[11] U. SUBSET-037, Ertms/etcs euroradio fis , tech. rep,.

[12] É. Masson and C. Gransart, "Cyber Security for Railways – A Huge Challenge – Shift2Rail Perspective," in *Proceedings of the Communication Technologies for Vehicles: 12th International Workshop, Nets4Cars/Nets4Trains/Nets4Aircraft '17*, pp. 97–104, Toulouse, France.

[13] R. B. I. Gashi, R. Bloomflied, and R. Stroud, "How secure is ertms?, Computer Safety, Reliability, and Security: SAFECOMP 2012 Workshops," in *Proceedings of the SAFECOMP 2012 Workshops: Sassur, ASCoMS, DESEC4LCCI, ERCIM/EWICS, IWDE*, pp. 247–258, Magdeburg, Germany, 2012.

[14] I. Lopez and M. Aguado, "Cyber security analysis of the European train control system," *IEEE Communications Magazine*, vol. 53, no. 10, pp. 110–116, 2015.

[15] E. Biham, "How to forge des-encrypted messages in 228 steps," Tech. Rep., Technion Computer Science Department, 1996.

[16] F. Pépin and M. G. Vigliotti, "Risk assessment of the 3des in ERTMS," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9707, pp. 79–92, 2016.

[17] T. Chothia, M. Ordean, J. De Ruiter, and R. J. Thomas, "An Attack against message authentication in the ERTMS train to trackside communication protocols," in *Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security, ASIA CCS 2017*, pp. 743–756, are, April 2017.

[18] R. Networks, "Openbts," 2018, http://openbts.org.

[19] E. Research, "Usrp n210," 2018, http://www.ettus.com/product/details/UN210-KIT.

[20] Digium, "Asterisk," http://www.asterisk.org/.

[21] Sysmocom, "sysmousim-sjs1 sim card," 2018, http://www.sysmocom.de/products/sysmousim-sjs1-sim-usim.

[22] "Openlte," 2018, http://sourceforge.net/p/openlte/wiki/Installing.