




TRAKS: A Universal Key Management Scheme for ERTMS

Richard J. Thomas, Mihai Ordean, Tom Chothia and Joeri de Ruiter[§]

University of Birmingham, [§]Radboud University Nijmegen

R.J.Thomas@cs.bham.ac.uk

 *cs.bham.ac.uk/~rjt195/acsac2017*



Motivation

- Current scheme for ERTMS key management is over 20 years old
 - Results in a solution that doesn't scalable
- Key material is sent in plaintext when a train/RBC is built
 - All keys potentially compromised from the outset
- Proposed solution does not offer any forward secrecy
 - Capture now, break later in a quantum world
 - Not quantum-secure either
- Security compliments Safety
 - but it's not part of the development process



Motivation

- ❑ Current scheme for ERTMS key management is over 20 years old
 - Results in a solution that doesn't scalable
- ❑ Key material is sent in plaintext when a train/RBC is built
 - All keys potentially compromised from the outset
- ❑ Proposed solution does not offer any forward secrecy
 - Capture now, break later in a quantum world
 - Not quantum-secure either
- ❑ Security compliments Safety
 - but it's not part of the development process



Matthew Green
@matthew_d_green

After looking at the signature and ciphertext sizes for these NIST PQ submissions, I think I need to upgrade my Internet plan.

04/12/2017, 19:13



Motivation



[Home](#) > [Threats](#) > [Reports](#)

Report

Weekly Threat Report 1st December 2017

Created: 01 Dec 2017

Updated: 01 Dec 2017

Cyber criminals target the Regional Transit System in Sacramento, California

Cyber criminals have reportedly compromised the corporate IT system of the Sacramento Regional Transit District (SacRT), deleting internal operations data. SacRT is the sole operator of local public bus and tram services in the Sacramento area of California, but reports suggest services were unaffected by the breach.

The attack began when hackers defaced SacRT's website, stating that they were "good hackers" seeking to help the organisation fix website vulnerabilities and requested SacRT contact them. When contacted, the attackers said they had access to corporate systems and demanded \$7000 worth of Bitcoin be paid to prevent deletion of data. SacRT refused to pay the ransom resulting in approximately 30% of its data being deleted. This affected the organisation's internal operations including the ability to dispatch employees and assign buses to routes.

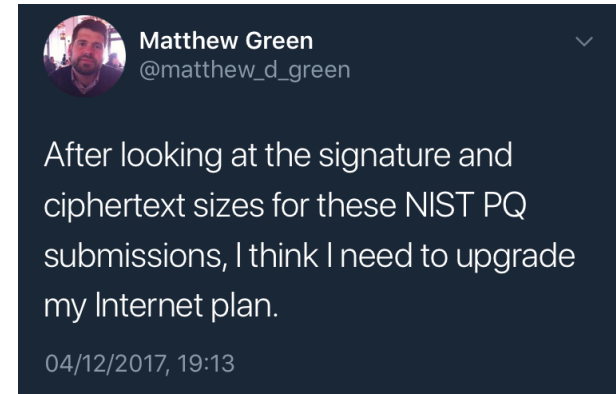
SacRT was able to make use of backups to restore the deleted data. The organisation also took down its website and shut down systems used to process credit card payments as a precaution. Passengers were still able to pay fares using cash and through SacRT's mobile app that is hosted separately on a cloud-based system. It is reported that customer data was unaffected by the breach and that no data was stolen.

It is over 20 years old

RBC is built
outset

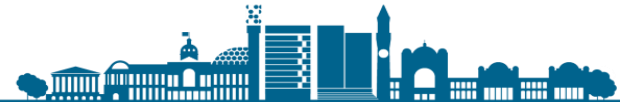
1 secrecy
rld

is



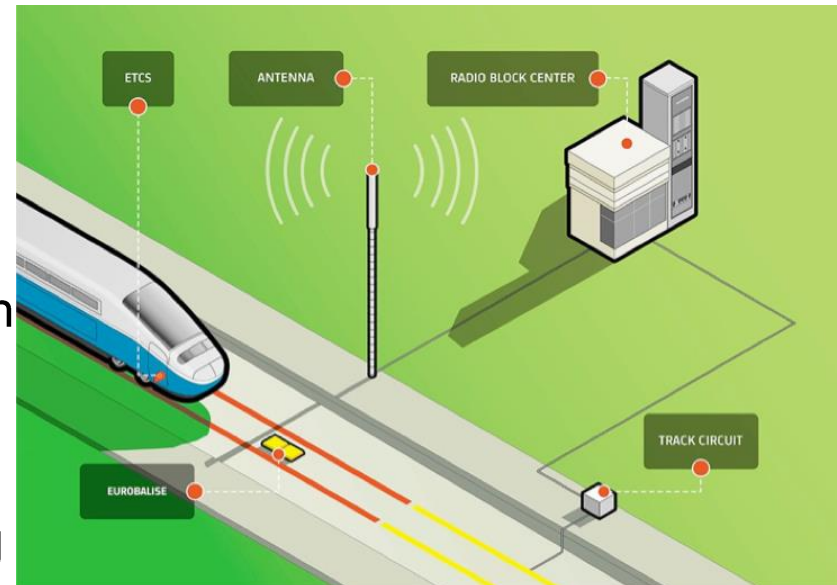
Outline

- The European Rail Traffic Management System (ERTMS) - overview
- Cryptography in ERTMS – the who, what and where
- ERTMS Key Management
- Formally Defining ERTMS Key Generation
- *TRAKS*: A Universal Key Management Scheme for ERTMS
 - Architecture and goals of the scheme
 - Security Analysis
 - Enforcing a responsible key lifecycle and distribution
 - Case Studies
- Conclusion



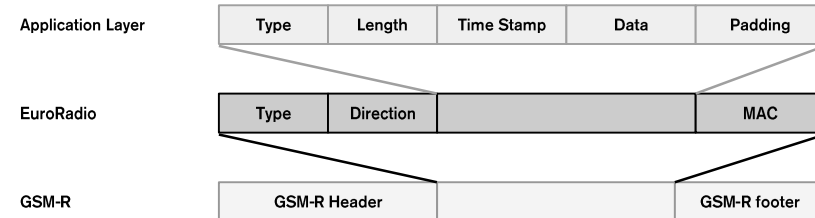
ERTMS Overview

- European Rail Traffic Management System
 - Responsible for **Train Management** and **Signalling**
 - **Wholly digitised**, supported by a number of protocols, e.g. EuroRadio
 - End 2014 – 80,000km of track covered by an ERTMS implementation
 - 50% located in Asia
- Moves signalling from line-side signalling to in-cab systems
 - Open standard, with a stack based on GSM-R, EuroRadio (for authentication) and an application layer



ERTMS Protocols

- Communications Protocol – GSM-R
 - Based on the GSM standard, with additional functionality (e.g. pre-emption and group calling)
- EuroRadio
 - Provides integrity and authenticity for safety-critical messages through a MAC
 - Currently a ISO-9797-based MAC
 - MACs keyed using a unique symmetric key between train and RBC
- Application Layer
 - Ensures timely receipt of messages, using pre-defined structures



Previous/Relevant Work

- Formal Analysis of EuroRadio and Application Layers (de Ruiter et al., 2016) showed session key negotiation is secure, but a cryptographic analysis of the MAC Algorithm showed it was vulnerable to a collision attack (Chothia et al., 2017).
 - No analysis of ERTMS Key Management from a security perspective to date.
- EU Rail Infrastructure Managers, e.g. ProRail propose an alternative part-PKI solution in 2012, but is superceded by SUBSET-137, an online Key Management Scheme for ERTMS.
- Fuloria et al., 2010, 2011 consider Key Management from an Energy Perspective, but lacks portability.

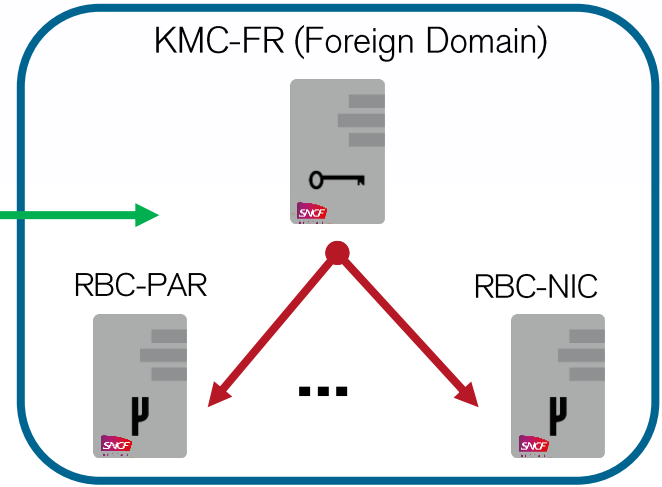


Current ERTMS Key Management

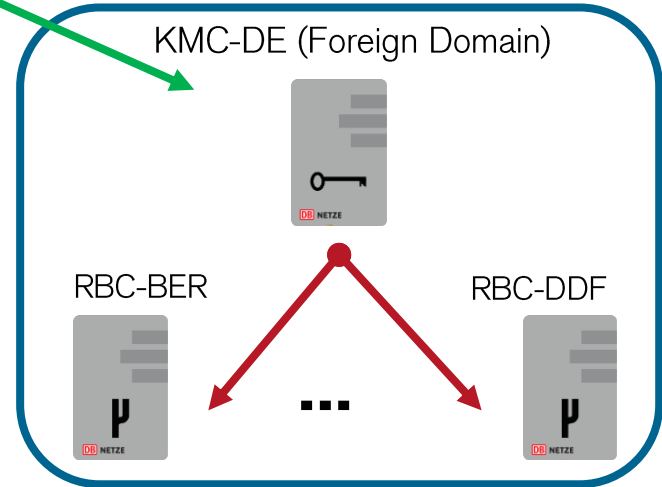
KMC-GB (Home Domain)



K-KMC1_{GB-FR}
K-KMC2_{GB-FR}



K-KMC1_{GB-DE}
K-KMC2_{GB-DE}



K-KMC1_{BHM}
K-KMC2_{BHM}

RBC-BHM



K-KMC1_{LDN}
K-KMC2_{LDN}

RBC-LDN



K-KMC1_{GB-390123}
K-KMC2_{GB-390123}

K-MAC_{BHM-390123}
KSMAC

K-MAC_{LDN-390123}
KSMAC



390123



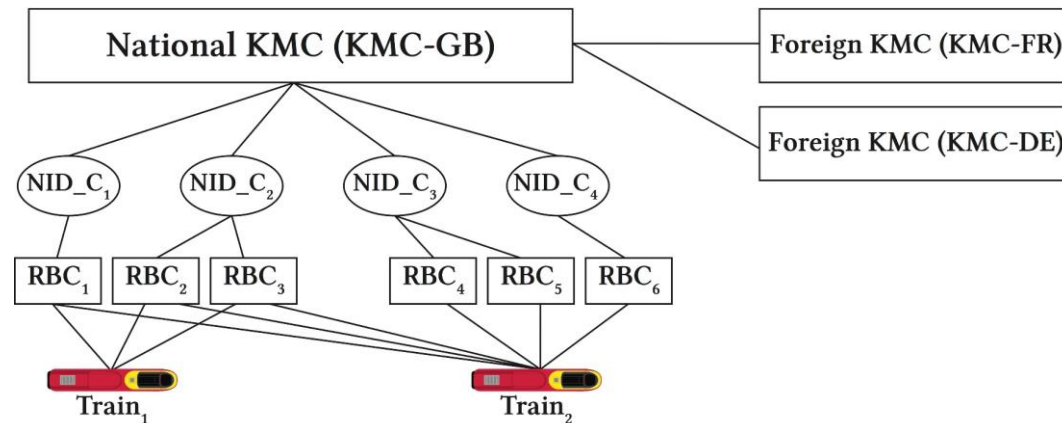
Issues with this scheme

- ❑ Scheme is highly inefficient
 - During 'National Deployment', engineers will be required to visit every RBC for a new train
- ❑ Trains and RBCs initially are given their 'transport' keys in plaintext, as they have no trust anchor
- ❑ Cross-border operation requires significant work for all parties involved for new trains
- ❑ Soon insecure as it depends on 3DES encryption and MACs, with no proposed alternative
- ❑ Proposed online solution relies on RSA Certificates and encryption, which is not post-quantum secure



TRAKS: A Universal Key Management Scheme for ERTMS

- Backwards-compatible with the existing standards
- Offers improved cross-border key management
- Reduces key management complexity and operational overheads
- Provides longevity to ERTMS Key Management with post-quantum security



ERTMS Keys	Current	TRAKS
NID.C Secret	×	$knid_c$
RBC Derivation Key	×	$km_{rid,null}$
Train Key	$K_{MAC_{rid,oid}}$	$km_{rid,oid}$
Balise Secret	×	km
Balise NID.C Area Key	×	$km_{NID.C,null}$
Balise MAC Key	×	$km_{NID.C,bgid}$



Formalising ERTMS Key Management and TRAKS

□ Defining ERTMS Key Generation

- $knid_c \leftarrow SGen(1^\lambda)$
- $ID_t \leftarrow INIT.ID(EDB, t)$
- $km_{id, id'} \leftarrow GEN.KMAC(id, id', knid_c)$

□ Applying to ERTMS and TRAKS:

ERTMS Keys	Current	TRAKS
NID_C Secret	×	$knid_c$
RBC Derivation Key	×	$km_{rid, null}$
Train Key	$K_{MAC_{rid, oid}}$	$km_{rid, oid}$
Balise Secret	×	km
Balise NID_C Area Key	×	$km_{NID_C, null}$
Balise MAC Key	×	$km_{NID_C, bgid}$

Algorithm 1: Offline ERTMS key generation

Input: id, id'

Output: $km_{id, id'}$

```

1 function GEN.KMAC( $id, id', null$ )
2    $km_{id, id'} \leftarrow SGen(1^\lambda)$ 
3   return  $km_{id, id'}$ 

```

Algorithm 2: TRAKS key generation

Input: id, id', s

Output: $km_{id, id'}$

```

1 function GEN.KMAC( $id, id', s$ )
2   /* for computing keys using  $s = knid\_c$  */
3   if  $id \neq null$  then
4      $km_{id, id'} \leftarrow PRF(id, s);$ 
5     if  $id' \neq null$  then
6        $km_{id, id'} \leftarrow PRF(id', km_{id, id'});$ 
7     /* for computing OBU-RBC keys using
8        $s = km_{rid, null}$  */
9     else if  $id = null$  then
10       $km_{id, id'} \leftarrow PRF(id', s);$ 
11    return  $km_{id, id'}$ 

```



Security Analysis of TRAKS

- Proposing a new solution is all well and good...
 - but is it secure?
- Formal security analysis is required to show it is secure
 - Game-based approach to prove negligible advantage for an adversary.
 - Challenger generates a set of IDs id and id' , generates a new secret to generate unique keys for GEN.KMAC, for all but the last elements of id and id' .
 - Coin flipped, and if $b=0$, we generate a 'valid' key, otherwise generate a random key.
 - Attacker has to successfully determine which 'world' they are in based on the last key.
- If PRF is shown to be insecure, we are able to swap out the algorithms as needed with a proven secure alternative.
- Compromise of ERTMS entities – what happens?

```
Exp_{\mathcal{A}}^b(KMAC)
ID ← {i | i ∈ INIT.ID(EDB, t)}
ID' ← {i | i ∈ INIT.ID(EDB, t')}
s ←R SGen(1λ)
for id ∈ ID, id' ∈ ID' do :
  if (id, id') ≠ (last(ID), last(ID')) :
    km_{id, id'} ← GEN.KMAC(id, id', s)
  endif
endfor
if b = 0 :
  km_{last(ID), last(ID')} ← GEN.KMAC(last(ID), last(ID'), s)
else :
  km_{last(ID), last(ID')} ←R \mathcal{K}
endif
b' ← \mathcal{A}((km_{id, id'})_{id ∈ ID, id' ∈ ID'}, ID, ID')
return b'
```

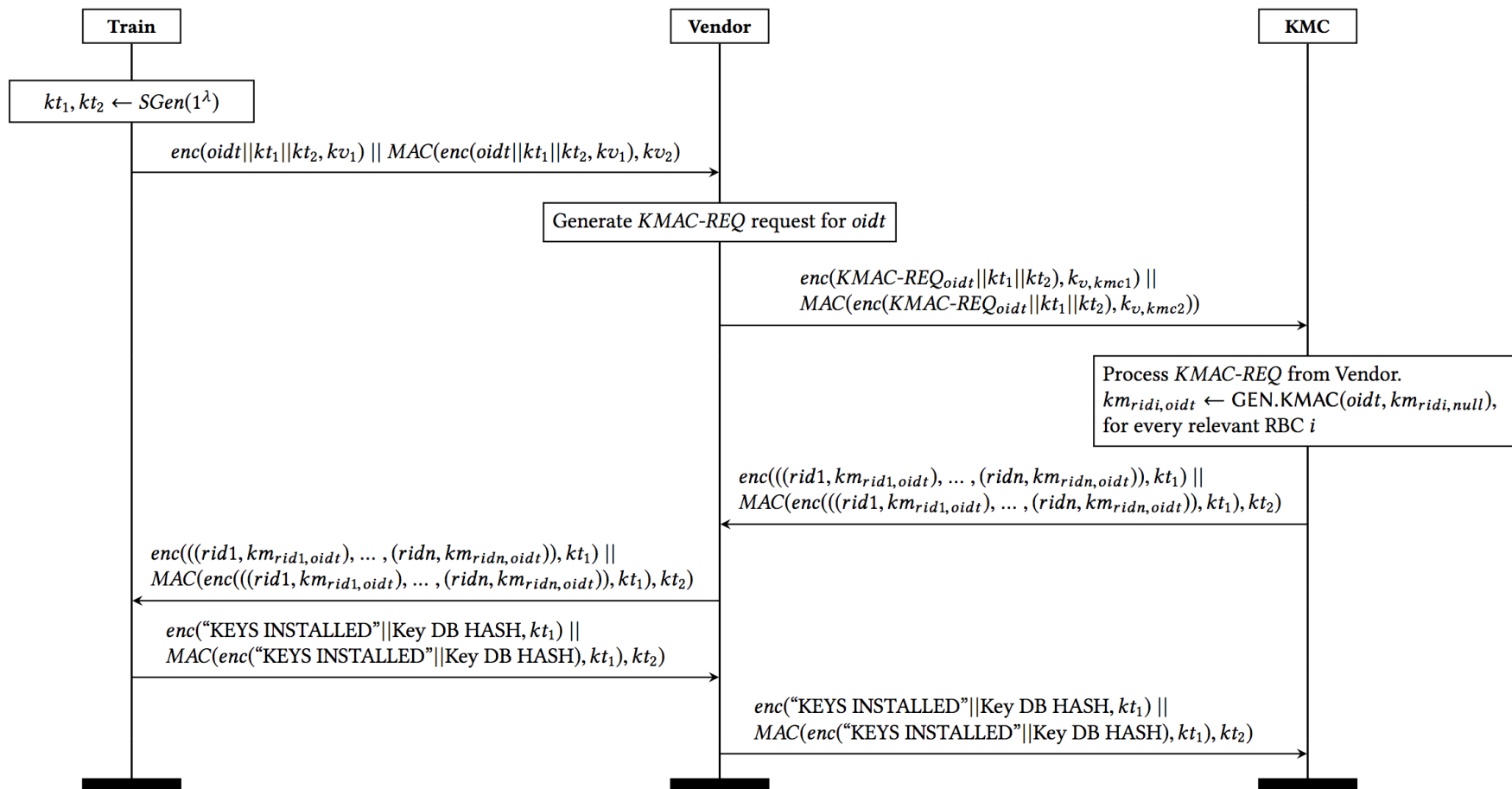


Enforcing Responsible Key Lifecycles and Distribution

- ❑ Current standard allows arbitrary lifespans which can be extended for convenience.
- ❑ Initial keying material is transmitted in the clear
 - Dishonest employee at a vendor can obtain all future keys for a given train.
- ❑ Consider four key stages of lifecycle:
 - Key Allocation, Distribution and Usage, Revocation, and Disposal/EoL
- ❑ Propose an alternative scheme which prevents MiTM attacks for safety-critical cryptographic material.



Ensuring a Responsible Management of Keys



Uses of *TRAKS* outside of EuroRadio

□ EuroBalises

- Trusted by trains and RBCs for accurate location, track profile and (in the UK) tilting data (Packet 44)
- Statically set payloads with CRC
- Malicious attacker could swap out balises without detection, with safety-compromising data
 - RBC will validate location against track-circuits but not speed/'packet 44'.
- *TRAKS* can provide appropriate keying for MACs
 - Provides true integrity and authentication validation of payloads
 - Trains carry out a derivation similar to an RBC for EuroRadio and validate the MAC.
 - If a bad MAC is found, it is recorded for the infrastructure manager/vendor to review
- Balises can now be made secure, where data presented forms part of a safety-critical decision

□ Wider ICS environments

- *TRAKS* partitions systems, where ICS environments can be broken down by function
- PLCs increasingly capable of doing crypto – MACs between devices can limit the effect of attacks against these devices



Conclusion

- ERTMS Key Management as a standard is over 20 years old
 - Volume of systems significantly higher today
 - Not scalable as a result, especially during National Deployment
 - Hampers cross-border operation with considerable operational overheads

- TRAKS reduces operational overheads and offers:
 - Backwards compatibility to the current ERTMS standards with few changes required to support
 - Post-quantum security against future threats
 - Improved cross-border operation by removing the burden on peer KMC managers
 - Flexibility outside of EuroRadio with portability for other applications






TRAKS: A Universal Key Management Scheme for ERTMS

Richard J. Thomas, Mihai Ordean, Tom Chothia and Joeri de Ruiter[§]

University of Birmingham, [§]Radboud University Nijmegen

R.J.Thomas@cs.bham.ac.uk

 *cs.bham.ac.uk/~rjt195/acsac2017*

