# *Crypto CAN't* – Confidentiality and Privacy for CAN/ISOBUS Networks in Precision Agriculture

Jan Bauer°, René Helmke°, Till Zimmermann°, Alexander Bothe°, Michel Löpmeier•, Nils Aschenbruck°

°University of Osnabrück, Institute of Computer Science
Wachsbleiche 27, 49076 Osnabrück, Germany
Email: {bauer, rhelmke, tzimmermann,
bothe, aschenbruck}@uos.de

•Competence Center ISOBUS e.V.
Albert-Einstein-Str. 1, 49076 Osnabrück, Germany
Email: michel.loepmeier@cc-isobus.com

***Abstract*—Modern agricultural machines and implements are equipped with numerous embedded sensors, producing extensive machine and environmental data, which often contains personal and privacy-sensitive information. Data streams are transmitted via ISOBUS, an internal vehicle bus that relies on the Controller Area Network (CAN) standard. However, neither ISOBUS nor CAN take privacy aspects into account. Thus, particularly with respect to the increasing interconnectivity of machinery, serious privacy concerns arise. In this paper, we briefly introduce our modular privacy framework *CAN't* that allows to purposefully filter, manipulate, and encrypt CAN data streams for the sake of privacy in the context of Precision Agriculture. The demo will present an open source prototype, realized using low-cost off-the-shelf hardware. Its technical feasibility and its benefits will be showcased by exemplary privacy filters applied to personal or business information, using both a commercial ISOBUS simulator and a custom simulator based on a video game.**

***Index Terms*—ISO 11783; ISOBUS; Controller Area Network; Privacy; Data Sovereignty; Precision Agriculture**

## I. Introduction

Modern agricultural is driven by a variety of technological innovations and continuously evolving through novel opportunities of information technology. The availability of global navigation satellite systems, such as GPS, has already revolutionized today's farm management and enabled Precision Agriculture applications with site-specific field treatments [1], [4]. Nowadays, sensor, communication, and Internet of Things (IoT) technologies allow the interconnection of machinery, the agricultural orchestration of processes, and a resource-friendly management. In addition, farm management information systems (FMISs) facilitate integrated planning and automatic documentation of agricultural processes. These innovations highly contribute to yield increases, resource optimizations and, thus, steadily improve sustainability.

In this context, modern machines are equipped with a plurality of Electronic Control Units (ECUs) such as controllers for various machine components and sensor devices, also for environmental information [1], [4]. For data communication between these ECUs, ISOBUS [6] is used, which relies on Controller Area Networks (CANs) [7], a network bus embedded into machines. Due to the broadcast nature of these networks, however, confidential and privacy-sensitive information is passed unprotected to all ECUs. Thus, when machines and implements of different parties collaborate with each other or machines are connected to the Internet, serious privacy and also data sovereignty concerns arise. To this end, we have developed *CAN't* [2], an effective privacy framework for the CAN-based ISOBUS.

Although *CAN't* is in principle also applicable in the automotive sector and without loss of generality, our demo is focused on two typical scenarios in agricultural practice. In the first scenario, a contractor is carrying out a certain process on a farmer's field. During this process the contractor not only collects machine-internal information but also precise environmental information that potentially allows to derive business-related conclusions on the fertility of the field and expected yields. In a second scenario, the machine collects personal driving information (e.g., speed and position) of an employee in road traffic, resulting in further questions regarding labor law.

The demo contribution is twofold: (1) We present our privacy framework, *CAN't* [2], and, using commercial ISOBUS hardware, we showcase how it enables data sovereignty and privacy in both scenarios. (2) We also introduce a lightweight cryptographic extension for our framework allowing confidential ISOBUS communication based on the Corrected Block eXtended Tiny Encryption Algorithm (XXTEA)[1]. This extension provides the opportunity to further increase privacy by protecting confidential data against eavesdropping.

The remainder of this paper firstly gives some background information about CAN and ISOBUS (Sec. II) and a very brief overview on related work in the context of CAN security and privacy (Sec. III). Then, the *CAN't* architecture (Sec. IV) and its privacy filters (Sec. V) are introduced. Finally, our demo contribution is presented (Sec. VI).

## II. Background

CAN [7] is a robust bus standard that is commonly used in the automotive industry due to its simplicity, the resilience against physical interferences, and the operational safety. Depending on the individual version and application, CAN offers fixed data rates of up to 1 Mbit/s. The medium access of CAN is based on CSMA/CR with message prioritization. Each basic CAN frame consists of a unique identifier (which also determines its priority), a length field, and a relatively small maximum payload of 64 bit.

[1] https://github.com/xxtea/xxtea-nodejs/blob/master/lib/xxtea.js

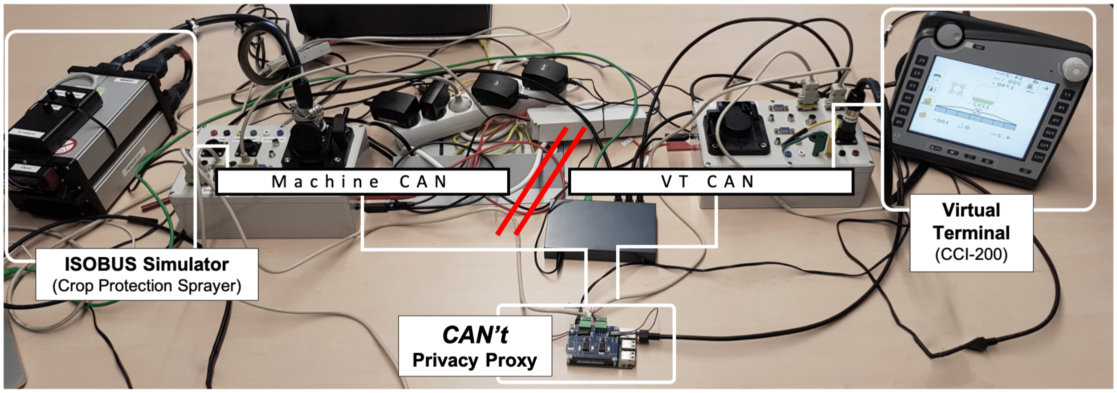

Fig. 1. Segmented CAN with attached ECUs, connected by a *CAN't* prototype to demonstrate the technical feasibility and the impact of privacy filters.

Especially for embedded communication between ECUs in agricultural machinery, CAN is extended by ISOBUS [6]. This standard provides versatile, non-proprietary, and manufacture-independent interoperability between machines and their implements. It is based on CAN 2.0 B, i.e., with the Extended Frame Format (EFF) and 29 bit identifiers and a nominal bitrate of 250 kbit/s. ISOBUS also specifies data type identifiers, so-called Parameter Group Numbers (PGNs) that structure the payload of the underlying CAN frame into a predefined number of data types that are semantically related to each other. Therefore, for each data type within a certain PGN, a second internal identifier called Suspect Parameter Number (SPN) is defined. The PGN is transmitted during the arbitration phase of a CAN frame, while the corresponding SPNs are internally mapped in the software implementation of an ECU [6]. Details on ISOBUS data types and both identifiers can be found in the VDMA ISOBUS Data Dictionary[2].

ISOBUS also standardizes protocols for various key entities on agricultural machines. These entities include a Task Controller (TC), which is used for the coordination of other ECUs in predefined tasks [6, Part 10]. Furthermore, there is the Virtual Terminal (VT) that is typically located in the cabin of a tractor and provides a universal graphical user interface (GUI) with flexible terminal functionalities [6, Part 6]. Some entities, particularly the VT, record, process, and aggregate information transmitted from other ECUs via ISOBUS, not only from the machine itself but also from attached implements that may belong to third parties. For this reason in particular, and because of the broadcast nature of the network in general, ISOBUS is vulnerable to common network-based attack vectors. Since neither CAN nor ISOBUS are designed with regard to security or privacy, no countermeasures to those attacks exist in the standards.

## III. Related work

Security for CANs has been increasingly investigated over the recent years and many approaches have been proposed in the literature. These approaches largely focus on operational safety in the automotive sector and propose solutions for message authenticity and integrity, cf. e.g., [5], [9], [10]. However, they rarely aim at privacy or data sovereignty and the integration into existing machines is often only partially possible.

Beyond authenticity and integrity, missing confidentiality has additionally been identified as a potential privacy threat by Hoppe et al. [5]. Confidential CAN communication is also addressed by Bruton [3] who proposes different encryption schemes such as AES and RC4. However, either a segmentation of CAN frames or an additional synchronization mechanism is required, which significantly induces additional bus load. As an alternative, Jukl and Čupera introduce an approach for ISOBUS message encryption with TEA [8]. Due to TEA's block size of 64 bit, payload segmentation is not required. This approach is very similar with the *CAN't* extension but is implemented for a specific microcontroller, whereas our solution is more generic. Overall, there is a lack of feasible CAN/ISOBUS confidentiality and privacy solutions that can be easily retrofitted into existing agricultural machinery.

## IV. System Architecture

*CAN't* provides a privacy framework for ISOBUS networks. It is based on a small and cost-effective ECU, implemented as a proxy. Therefore, the CAN is physically divided into two bus segments. The *CAN't* proxy operates as a man-in-the-middle (or as a network interconnection unit in the ISOBUS context [6, Part 4]) and bridges both segments, as visualized by the demonstration setup in Figure 1. Hence, it is able to relay, filter, and intentionally manipulate information when being forwarded from one segment to the next. In this way, broadcasts from ECUs of different segments can be suppressed or modified only selectively in order to ensure data privacy for sensitive information. At the same time, core functionalities of the machine can be maintained by forwarding relevant messages with unchanged information. To hide personal or business-related information, the recommended injection point for this proxy in our architecture is directly in front of a logging device, such as the VT, cf. Figure 1.

The proxy comprises four main components. Privacy filter and encryption algorithms are implemented in its *core* component. Here, the PGN of each incoming frame is identified and matched against a blacklist in its *data base* component,

[2] https://www.isobus.net/isobus/pGNAndSPN

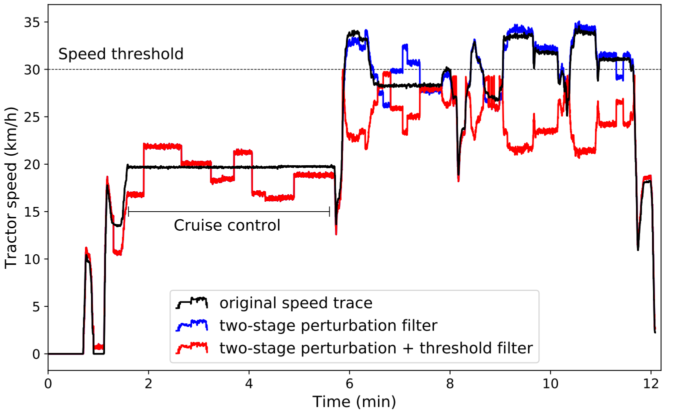

Fig. 2. Speed manipulation filter example applied to a real-world CAN trace.

which associates each PGN with possible privacy filters. In case one or more filters are activated for a particular PGN, the core component iteratively applies them to the content of the frame, before this frame is either dropped or being forwarded to the target segment. Using the *GUI* component, filters can be wirelessly configured and managed during the operation. In addition, the GUI also provides communication statistics using the *logging* component of the proxy. See [2] for further details on the *CAN't* framework, its components, and a performance evaluation.

## V. Privacy Filters

*CAN't* provides a modular set of different privacy filters offering basic operations. These filters can be applied to each PGN or SPN, respectively, and individually adapted. They can also be combined with other filters to create more complex filter strategies. Our current filter set is categorized into three different types. *Blocking* filters drop frames based on their PGN, independently from their payload. However, because some information is vital for the machine's operability, sometimes simply dropping certain frames might be impracticable. Therefore, there are also *value-centric* filters that apply some kind of mathematical operation to the original payload. Such operations include threshold or rounding operations in order to reduce the level of information content of corresponding messages. In addition, payload encryption allows the encryption of data that is logged by the VT as well as an encrypted communication between a pair of *CAN't* proxies. The third category are *perturbation* filters, e.g., filters that add noise to individual values of a specific SPN.

For instance, in a road traffic scenario (cf. Section I), a possible filter strategy could be applied to privacy-sensitive driving data, e.g., the driving speed (PGN 65096/SPN 1862). Such a strategy could firstly use a perturbation filter that obfuscates periods with active cruise control and could secondly use a threshold filter to keep the maximum speed below a configured threshold, as exemplarily visualized in Figure 2.

## VI. Demo Contribution

In our demo, we will show a prototype of *CAN't*. The prototype is realized on a Raspberry Pi platform that is extended by low-cost MCP2551 CAN transceivers. The software implementation uses *Google Go* and *SocketCAN*, an open-source set of CAN drivers included in the Linux kernel. Our software was recently released as open source software and is publicly available[3]. It contains a modular and expandable set of basic predefined filters. An extensive PGN/SPN data base can be imported from VDMA's Online Data Base[4]. Furthermore, it comprises an intuitive HTML5 GUI. Implemented as a single page application based on *ReactJS* and *TwitterBootstrap*, it allows the configuration and monitoring of filter rules.

Using a commercial ISOBUS infrastructure, comprising a VT manufactured by Competence Center ISOBUS (CCI) and a specific implement simulator, as depicted in Figure 1, the impact of *CAN't* privacy filters will be demonstrated during the simulated process. Additionally, in a second setup, an arcade racing video game will involve the audience by applying illustrative filters and a selective encryption to the driving information of the user. Therefore, speed and position data from the video game is translated to ISOBUS messages and transmitted via CAN. Finally, also the limitations of our approach, its potential application to the automotive sector, and future work will be discussed at the demonstration.

## Acknowledgments

This work was partially funded by the German Federal Ministry of Education and Research (BMBF) within the Program "Innovations for Tomorrow's Production, Services, and Work" (02K14A194) and managed by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the contents of this publication.

## References

[1] H. Auernhammer, "Precision farming – the environmental challenge," *Comput. Electron. Agr.*, vol. 30, no. 1–3, pp. 31–43, 2001.

[2] J. Bauer, R. Helmke, A. Bothe, and N. Aschenbruck, "CAN't track us: Adaptable Privacy for ISOBUS Controller Area Networks," *Comput. Stand. Inter.*, vol. 66, p. 103344, 2019.

[3] J. A. Bruton, "Securing CAN Bus Communication: An Analysis of Cryptographic Approaches," Master's thesis, National University of Ireland, Galway, August 2014.

[4] S. Cox, "Information technology: the global key to precision agriculture and sustainability," *Comput. Electron. Agr.*, vol. 36, no. 2–3, pp. 93–111, 2002.

[5] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive CAN networks – Practical examples and selected short-term countermeasures," *Reliability Engineering & System Safety*, vol. 96, no. 1, pp. 11–25, 2011, Special Issue on Safecomp 2008.

[6] Int. Organization for Standardization, "Tractors and machinery for agriculture and forestry – Serial control and communications data network – Parts 1–14," ISO 11783-{1–14}:2007–17, 2007.

[7] ——, "Road vehicles - Controller area network (CAN) – Part 1: Data link layer and physical signalling," ISO 11898-1:2015, 2015.

[8] M. Jukl and J. Čupera, "Using of tiny encryption algorithm in CAN-Bus communication," in *Research in Agricultral Engineering*, vol. 62, 06 2016, pp. 50–55.

[9] C. W. Lin and A. Sangiovanni-Vincentelli, "Cyber-Security for the Controller Area Network (CAN) Communication Protocol," in *Proc. of 2012 Int. Conference on Cyber Security*, 2012, pp. 1–7.

[10] Y. Wu, Y.-J. Kim, Z. Piao, J. G. Chung, and Y.-E. Kim, "Security protocol for controller area network using ECANDC compression algorithm," in *Proc. of the IEEE Int. Conference on Signal Processing, Communications and Computing (ICSPCC)*, 2016, pp. 1–4.

[3] http://sys.cs.uos.de/cant

[4] https://www.isobus.net/isobus/attachments/isoExport_csv.zip