# Security Assessment of the Advanced Cyclist Assistance System (ACAS)

## Project Summary

Out of the 1,230 cyclist deaths in the USA in 2021, 853 were caused by motor-vehicle crashes[1]. Although various technologies have been deployed to ensure greater road safety for all, these are typically focused on cars. For example, we can cite automatic braking and rear-view cameras. For other road users, such as cyclists, very few tools exist.

The Advanced Cyclist Assistance System (ACAS) developed by Continental bridges the gap in safety solutions for bike users. At the core of the system is a camera with danger-detection through Artificial Intelligence (AI). In this project, we integrated the smart camera to a phone application. In addition, a vibration system was added to a high-visibility vest worn by the user and connected to the application.

The various elements of the project are connected using Bluetooth Low Energy (BLE) and Wi-Fi. It is also required to download a phone application. All these aspects must be considered when considering the security of the overall project. An assessment of what is currently implemented in each of these aspects, as well as improvements required for commercialisation, will be conducted in this report. We will rely on the knowledge we have acquired in security in Internet of Things (IoT) through our course at INSA.

First, we will clarify the key items of the project and the way they are connected. Then, we will discuss the security measures in place for each of them. Finally, we will discuss improvements.

## Layout of the Project

The ACAS is a system that adds onto the user's bike. It takes the form of a camera gathering footage from behind the cyclist, like a rear view mirror. The footage is shown on the cyclist's camera. Additionally, it goes through an AI to detect dangers (such as a car coming close and fast). If a danger is detected, an alarm is sent on the phone and in the form of vibrations on a safety vest worn by the user. Finally, it is possible to parameter the system through the application, such as the kind of vibration sent.

An overview of the system is available on Fig. 1.

[1] https://injuryfacts.nsc.org/home-and-community/safety-topics/bicycle-deaths/

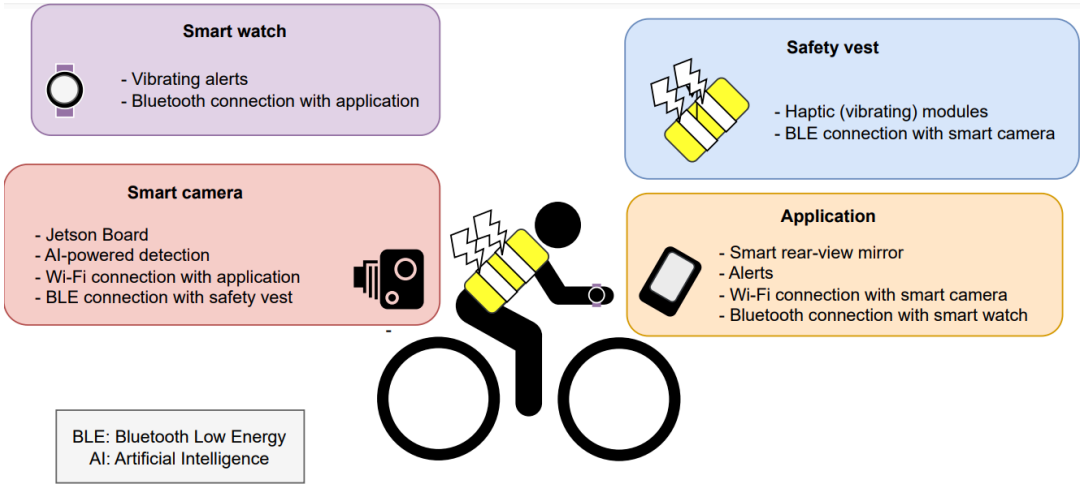


Figure 1: Overview of the system and its connections

# EBIOS Risk Manager

In order to analyze the main security threats on our project, we will follow an EBIOS-like method. The first step we will follow is identify the business values associated with our product and the possible dreaded events and security criterias that would be altered by said events.

| **Business Value** | **Dreaded Event** | **Security Criteria** |
|---|---|---|
| Protect cyclists | Disabling alarms and/or vibrations | Availability |
| | Sending false notification | Integrity |
| Protect the project reputation | Show vulnerabilities of the project | Confidentiality |

The second step is to identify the main risks and quantify them so we can determine if they are relevant to be considered in our security analysis.

| Identification | | Quantification | | Evaluation of Relevance | |
|---|---|---|---|---|---|
| Risk Source | Targeted Objective | Motivation | Resources | Relevance Level | Validation of Criteria |
| Security experts | Show the vulnerabilities of the product | +++ | +++ | High | ☑ |
| Personal Enemy of the User | Sending false notification | + | + | Low | ☐ |
| Personal Enemy of the User | Disabling alarms and/or vibrations | + | + | Low | ☐ |

From both tables we deduct that the security experts form the highest risk for the credibility of our product and it has to be taken into account. In order to protect ourselves from this risk source, we decided to secure the Wi-Fi communication between the Jetson and the application as well as the Bluetooth communication between the Jetson and the vest.

In the following part, we will describe all the measures that were taken and the ones that could be taken in order to maximize the security and reduce as much as possible the risk.

# Bluetooth Low Energy connection

### Security options in BLE

Multiple security options come with BLE protocol. Specifically, Bluetooth differentiates between security modes and levels, with one mode containing multiple levels.[2]

- Level 1 of security mode 1 provides no added security.
- Other levels of security mode 1 enable encryption and authentication (with AES128-CCM).
- Security mode 2 enables data encryption with a special key to verify data authenticity. As a guideline, this mode is described as “sufficient” when confidentiality of data is not a key requirement.
- Security mode 2 and 3 have a different pairing mechanism. When using authenticated pairing, information is shared out-of-band. This would cause Man-in-the-middle (MITM) attacks to fail as the attacker would not know about this additional information.

Whilst newer versions attempt to mitigate these risks, BLE is still vulnerable to numerous attacks. Nevertheless, an older and more vulnerable pairing procedure (known as legacy pairing) can still be used in Bluetooth connections. It would be safer to rely on the new pairing procedure known as LE secure connection (LE-SC). LE-SC uses elliptic curve Diffie-Hellman (ECDH) key exchange to allow both parties to agree on a shared key without transmitting it over the air: thus the key exchange cannot be eavesdropped.

However, for compatibility reasons, legacy pairing is used if one of the devices does not support LE-SC.

### Used libraries and their limits

We used two libraries for this project: one from the Jetson board, SimpleBLE, and one from the Arduino point of view with the XIAO nRF52840, Bluefruit nRF52. Neither of them offer LE-SC as a pairing mechanism, which is a significant security flaw.

They both support *some* sort of higher security mode, but neither of them fully implement either of them. It makes it very difficult to use security modes with them. In BLE, when devices try to implement different security modes, the lower of the two is kept.

To mitigate this issue, we would have to maybe use a lower level approach to Bluetooth that would allow us to use a higher security level. Alternatively, an application-level protocol could

[2] A survey on Bluetooth Low Energy security and privacy

be used.

# Wi-Fi connection

Wi-Fi is a secure protocol for wireless communications. It comes with multiple security features built-in. Using secured connection with WPA2 or the newer WPA3 encryption helps secure the communications between the camera and the smartphone making MITM attacks more complicated. To take the security of the communications even further, we could use a secure protocol for transferring the video. For example, we use the RTP protocol but a secure version called SRTP exists and could have been implemented. However, doing so probably isn't necessary as the Wi-Fi is already safe.
The safest solution would be to use a cable making it impossible to execute any MITM attack on our system.

# Design of a secured communication protocol

In addition to using the native security parameters of WiFi and Bluetooth we could have designed our own applicative communication protocol on top of these layer 2 (OSI Model) protocols. The final decision of our team was to not implement it as it would have been a lot of effort for a small increase of the security of the product. In addition to that, the consequences of an attack would be negligible. Indeed, our product is a device that aims to increase the security of the cyclist, but the basic management of the security is still left to the cyclist that should not, in any case, reduce his focus on the road.

However to leave a track of what could have been possible to implement, we thought about implementing some security mechanisms.
First of all, we thought about the possibility for the phone (client) to send a code to the jetson (AP) when it connects in order to be authenticated as a legitimate device. That would prevent illegitimate users from connecting to the Jetson. This would provide better authentication to our system.
We then thought about implementing some sequence number in our packets in order to prevent replay attacks but it is already implemented by the 802.11 (WiFi) standard.
Finally, we could have encrypted our communication also at the application layer to limit the impact of MITM attacks. However, this solution would have been hard to implement as it would cost a lot of bandwidth that we cannot afford due to the real time video transfer context.

# Conclusion

To conclude on this report, the security risk in our project is very limited. The attack surface is very limited as neither of the devices are connected to the internet. In addition to that, the

consequences of attacking our system are very limited as an attacker can only block the notification which does not represent a danger for the user.
Moreover, as the system is mobile with short range communication protocols (WiFi, Bluetooth) it hardens the hacking of the system over time.

Nevertheless, we still implemented some basic security using the native security that our communication protocols. Particularly, the implementation of the WPA2 protocol for WiFi and legacy pairing for Bluetooth.