A Novel IoBT Security Assessment Framework: LoRaWAN Case Study

Federico Mancini[a] and Frank T. Johnsen[a]

[a]Norwegian Defence Research Establishment (FFI), Kjeller, Norway

ABSTRACT

The Internet of Things (IoT) is interesting because this potentially disruptive technology can also be used as a low-cost approach to augment information in military systems. This makes following IoT trends important when considering the future of command and control systems. However, it is critical to make sure that employing IoT for defense purposes, the so-called Internet of Battlefield Things (IoBT), does not introduce unacceptable security risks into the missions. Because of the large variety of devices and the different ways in which they can be used, there is no one-size-fits-all security solution, and individual assessments would, ideally, have to be performed for each specific case. This can quickly become an unmanageable overhead in the mission-planning phase, which can lead to either discarding the technology or using it in an insecure way.

In this paper, we discuss a more systematic and efficient approach to assess and manage the security risks associated with IoBT. We outline a modular framework where mission-dependent security requirements can be derived and tested against actual security assessments of relevant IoT devices and technologies. Residual risk can then be identified and systematically reduced to an acceptable level where possible. We evaluate the framework by assessing the security of two implementations of the Long Range Wide Area Network (LoRaWAN) protocol in two different military scenarios.

Keywords: IoT, security, LoRaWAN

Topic 7: Other C2 Related Research and Analysis

Paper ID 12

Point of contact

Federico Mancini
Norwegian Defence Research Establishment (FFI)
P.O. Box 25, 2027 Kjeller, Norway
E-mail: Federico.Mancini@ffi.no

1. INTRODUCTION

The recent proliferation of affordable civilian consumer electronics has enabled a wide array of new possibilities that were beyond reach not many years ago. This trend, which can be referred to as the *Internet of Things* (IoT), is characterized by *the interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data.*[1] Also, *Makerspaces are proving to be breeding grounds for a new wave of novel IoT devices*,[2] as these venues provide an arena for information exchange, hands-on development and rapid prototyping of new ideas. The driving force is a desire to innovate, as well as using the IoT devices as a means to simplify or improve tasks in daily life. Typical areas that have been identified to benefit from applying IoT include manufacturing, supply chains, transportation systems, healthcare, infrastructure, and industrial automation. Conversely, we can also envision that employing IoT for defense and public safety will yield a lot of new opportunities for different applications as well. So, it is important to consider IoT and related technologies as capabilities enabling new functionality for the future of command and control (C2) systems. In,[3] target scenarios for mission-critical IoT are identified to include energy management, surveillance, collaborative and crowd sensing, smart cities, personal sensing (including healthcare), logistics, fire-control systems, and C4ISR (Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance).

Another term for mission-critical IoT is that of Internet of Battlefield Things (IoBT), which is the term we will be using in this paper. It is important to note that there could basically be two fundamentally different approaches to IoBT: 1) Develop military-specific IoT devices, or 2) Employ commercial off-the-shelf (COTS) consumer electronics IoT approaches for military purposes.

In this paper we pursue the second approach. The motivation for this is to leverage the momentum civilian IoT has, both in the proliferation of affordable devices, but also from a cost perspective in that consumer electronics has negligible costs when compared to hardened military-specific devices. We can envision many different applications, but must emphasize that adequate security is critical if missions become dependent on these devices. What makes COTS IoT devices attractive (low cost and many potential applications), is also what makes them insecure, as there are few incentives for the manufacturers to provide additional security mechanisms out of the box. It is also clear that devices and missions are very diverse, have different goals and capabilities, so that risks and possible security controls to manage these risks need to be evaluated case by case.

We need an approach that allows us to perform these assessments and quickly decide if IoT technology is secure enough for a given mission, and if not, what additional security controls may be implemented to reduce the risk to an acceptable level. An additional advantage would be to be able to perform the assessments in a consistent and systematic manner, so that previous results can be easily integrated in new assessments. The current research on security of IoT is often very specific for a given technology or uses different terminologies and classifications for the types of threats and risks they analyze. In any case, it focuses on commercial applications and therefore presupposes a different risk picture than military operations. In this paper, we propose a framework that allows to integrate existing security research and best-practices over a common basis. The novel framework is specifically devised for a military context and allows for rapid and repeatable risk and security assessments. What we present here, is the initial version of the framework. To verify that it serves its purpose and identify potential improvements, we apply the framework to evaluate whether two specific IoT devices implementing the Long Range Wide Area Network (LoRaWAN) protocol for communication, can be securely employed. We consider these devices for two different scenarios: 1) Unit tracking with GPS, and 2) surveillance reporting when detecting intruders in a remote area. The framework combines a top-down risk assessment to derive mission-specific security requirements, with a bottom-up assessment of relevant technology through field experiments and technology studies. The result of the overall assessment is then refined by designing and testing additional security to reduce the residual risk to an acceptable level, if possible.

The remainder of this paper is organized as follows: Section 2 covers related work. Section 3 presents our framework and its components in detail. In Section 4 we describe and analyze the two chosen scenarios through the top-down step of the framework. In Section 5 we describe the LoRaWAN protocol, the two IoT devices, supporting experiments and apply the bottom-up step to derive a security assessment. The overall assessment to evaluate whether the devices can be securely employed in the described scenarios, and additional security controls to further reduce the residual risk, are presented in Section 6. Finally, Section 7 concludes the paper.

2. RELATED WORK

The use of IoT for defense applications is being actively researched in many fora. Fraga-Lamas et al.[3] have performed a review of IoT for defense and public safety. Their work points to several different application areas, including those we pursue further in this paper. Notably, the paper focused on architectural approaches and gave several examples of supporting technologies, but LoRaWAN was not mentioned or addressed in that work.

The NATO Research Task Group (RTG) IST-147 titled "Military Application of Internet of Things" examined applying COTS civilian IoT approaches for military purposes. Typically, the use case was centered around a humanitarian assistance and disaster relief (HADR) coalition operation in a Smart City, where IoT information from the city could be used as additional sensor input to the military situational awareness and hence C2 systems.[4] Following that group's conclusion in 2019, this work now continues in NATO RTG IST-176 titled "Federated Interoperability of Military C2 and IoT Systems". That group continues work on Smart Cities, but also broadens the scope to include such use cases as our recent work on crowdsourcing and crowdsensing.[5]

More specific on securing these IoT devices in a defense context, Wrona[6] emphasizes that security is a paramount challenge that needs to be addressed at every level of IoT, from the high volume of endpoint devices that gather data and execute tasks, to cloud-based control systems through network infrastructure. Some of these aspects are what we aim to provide a systematic approach to, by proposing the security framework we present in this paper.

The NATO RTG IST-164 "Securing Unmanned and Autonomous Vehicles for Mission Assurance" is developing a framework to support mission dependent risk assessments and derive consistent security requirements across different types of military operations and autonomous vehicles.[7,8] The framework presented here is inspired by the same approach, but it is geared towards IoBT rather than autonomous vehicles, and focuses on the assessment of existing devices, rather than on the creation of guidelines for the design of new devices specifically thought for military missions. This implies also that this work is not a duplication of the existing results on the security of IoT, but rather we see the existing literature as a resource to improve the design of the framework and populate it with relevant information.

For instance, Mekki et al.[9] have performed a comparison of Low Power Wide Area Networking (LPWAN) technologies, including LoRaWAN. They point to LoRa's main strengths being battery lifetime, capacity, and cost. In their view, LoRa will serve as the lower-cost device, with very long range (high coverage), infrequent communication rate, and very long battery lifetime. Further, LoRa will also serve the local network deployment and the reliable communication when devices move at high speeds. They identify the following application areas as suitable for using LoRa: Smart farming, manufacturing automation, smart buildings, and tracking for logistics. Conversely, they envision that Narrowband IoT (NB-IoT) will serve other applications in the higher-value IoT markets that are willing to pay for very low latency and high quality of service.

The military application aspect has been investigated by Michaelis et al.,[10] who used the USA version of LoRaWAN (in the 915 MHz band) to track vehicles in an urban environment (downtown Montreal, Canada). Their findings show a usable range of LoRaWAN of up to 5 Km under the conditions tested. This information can be used to improve the security assessment of the technology in question.

3. FRAMEWORK FOR RISK AND SECURITY ASSESSMENT

The framework presented here to support the risk and security assessment of IoT devices, is partly based on the underlying approach of the framework developed by the IST-164 for autonomous vehicles. It uses a top-down approach to derive mission-specific risk assessments that are then used as the basis to formulate security requirements for the devices. However, it is more contained since IoT devices are simpler and usually have a smaller role in the mission than autonomous vehicles. In addition, it includes a bottom-up component that is used to derive security assessments of a concrete device and the technologies it implements through actual tests and available documentation. This assessment is compared to the top-down requirements to evaluate the risk of using the device in the given mission, and if this risk is unacceptable, it is possible to consider additional security controls and repeat the security assessment. The framework organization is shown in Figure 1. Each component of the framework contains some modules with IoT specific guidelines, reference models, best-practices and so on, to support the different kinds of assessment, which we describe in more detail below.

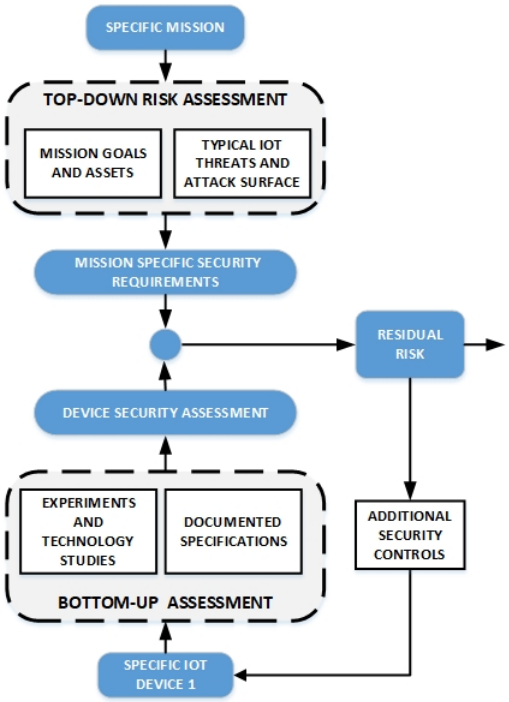

Figure 1. The security evaluation approach used in this paper.

3.1 Framework modules

In order for the framework to enable systematic and consistent risk and security assessments, it needs to offer the tools to perform the assessments in an efficient and repeatable way. The overall organization defines the steps to be taken and the order in which they should be performed, but each step provides also specific support in the form of reference models, standardized guidelines and established best practices. This will provide a common basis for the assessments, guarantee that fundamental security concerns are considered and properly handled, and that different assessments can be compared against each other. Furthermore, modules can be added as needed or updated with the newest information from either standardization and regulatory bodies, technology trends, experiments or academic work. Here, we propose a preliminary set of modules, shown as white boxes in Figure 1, that can constitute the basis for the framework.

3.1.1 Mission goals and assets

Missions vary in both goals, duration, resources and organization, so we cannot provide a generic reference model for the mission risks that covers all possible scenarios. What we can define are guidelines for systematically identifying what are the relevant assets to protect when employing IoT devices, and use this information as the starting point for the risk assessment. The first step is to identify which mission goals are supported by the information produced, stored, processed and communicated by IoT devices. Once this is clear, the second step consists in analyzing how a compromise of the confidentiality, integrity and availability (C,I,A) of this information can affect the mission. Additionally, other properties like the amount, freshness and life-cycle of the information assets can be considered to produce an even more detailed criticality assessment. This will constitute the starting point to evaluate the threats to these information assets.

3.1.2 Typical IoT threats and attack surface

Risk is typically defined by the likelihood and the impact of a threat, and while the impact is defined in the previous module by assessing the criticality of the assets, the likelihood is dependent on how easy it is for a threat to exploit the device vulnerabilities in the given setting to cause that impact. This module supports the assessment of the likelihood by providing a list of typical threats against the C,I,A of information assets

that should be considered when employing IoT devices. We compile this preliminary list by analyzing a generic system model inspired by the Raspberry Pi system architecture.[11] The model is shown in Figure 2.

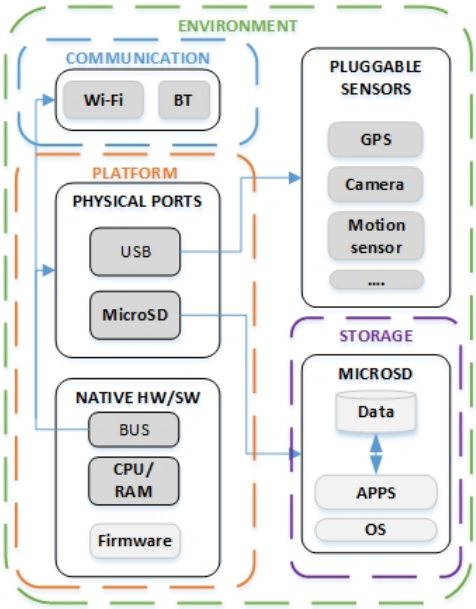

Figure 2. Generic system model for IoT devices.

Relevant threats are mainly the same as for any other information system and have been already discussed in the literature, but it is important to see them in a typical military setting. To make it easier to go systematically through the threats during the assessment, we categorize them based on the attack surface of the device as shown in Figure 2 (platform, network, storage and environment) and list them in Table 1. The greatest threats to the platform are those exploiting physical access to the device by connecting to the exposed physical ports, by damaging the device, or by removing or even replacing parts. These threats are particularly likely if the devices are left unattended in an uncontrolled area, and the fact that they are low-cost systems may mean that they lack even basic security controls or support and resources to develop effective ones. Compromising the integrity of the platform undermines the integrity of the whole system, and if no integrity verification is implemented, it is not only difficult to prevent this, but also just detecting the it happened. Threats associated to the network are mostly about lack of proper security which allows for eavesdropping, manipulating and spoofing the data transmitted on the channel. Remote access to the device can also be possible, but in a military setting this threat may be less serious if local private networks are used. Threats to the storage are those that can lead to compromise of data stored on the device, despite additional security controls to compensate for the lack of platform protection. For instance, encrypting data could help if the removable storage unit with data on it is stolen, but an additional threat could be sloppy implementation, where the decryption key is stored in clear on the same unit. Finally, environmental threats are those taking place outside the device and are the most difficult to defend against. They include jamming, detection, and tracking of electromagnetic signals and sensors spoofing.

3.1.3 Bottom-up modules

This is the less mature part of the framework, and it is more difficult to develop as it requires in-depth knowledge of existing technology, acquired either through actual field or lab experiments or through studying existing technical documentation. Ideally, a library should be created with all the assessments and field tests performed on different types of devices, so that it can be re-used for different missions. In this paper, we use experiments performed with LoRaWAN and available documentation on the devices as our basis.

Table 1. Typical threats associated to IoT devices that can compromise the C,I and A of the data they process, store and communicate.

Threats on the C,I,A of IoT sensor data	
Attack surface	Threats
Platform	Physical damage to device (A) Removal or replacement of parts (C,I) Access through physical interfaces (C,I,A) Lack of built-in security (C,I,A) Lack of support for security (C,I,A) Lack of frequent updates (C,I,A) Lack of integrity verification (C,I,A)
Network	Use of clear text (C) Lack of integrity protection (I) Lack of mutual authentication (C,I) Unauthorized remote access (C,I,A)
Storage	Lack of dedicated storage protection (C,I) Weak protection mechanisms (C,A)
Environment	Signal analysis (C) Sensor spoofing (I) Jamming (A)

3.1.4 Iterative concept and experimentation

By comparing the assessed security from the bottom-up step to the top-down requirements, we should be able to evaluate the residual risk of using a given device in the current mission. If the risk is too high, it is possible to either discard the device or select additional security controls that may be applied to further reduce the risk to an acceptable level. Ideally, the framework should provide a selection of established controls, either at a conceptual level or in the form of a ready-to-use product. In either case, one would need to integrate or implement the control on the device and test its effect, hence repeating the bottom-up step, until the risk is acceptable or the device is deemed unfit for the purpose of the mission.

4. SCENARIOS AND TOP-DOWN ASSESSMENT

We consider two scenarios where IoT devices are used to collect and report different kinds of sensor data in two different operational settings: Unit tracking and area control. For these, we perform a risk assessment by evaluating the criticality of the C, I and A of the sensor data for the mission goals, and the risk associated to the different threats in Table 1. Based on this, we formulate the security requirements for each scenario, which are more detailed statements about the risks one should protect against.

4.1 Unit tracking

Moving units, like convoys transporting supplies, military vehicles patrolling an area, or dismounted soldiers operating in an urban setting, can be tracked by using GPS sensors that report their position to a C2 center. The main goal is to be able to track the units in near real-time and with sufficient accuracy, in order to build an updated operational picture and be able to react quickly and effectively in a coordinated manner. However, it is also important to consider that tracking units is but an activity in the overall operation, which may have other high-level goals, like locate and neutralize a threat or safely transport valuable goods to a destination.

4.1.1 Risk assessment

From the IoT perspective, the main assets supporting the operation are the GPS data fetched and reported by the devices placed on the units to be tracked. The confidentiality of the units' positions is not important for the tracking activity itself, but for the overall mission it may be critical that the enemy cannot use this information to avoid detection or even set up an ambush. The integrity and correctness of the GPS measurements are essential if tracking is to be of any use. The availability of the units' positions is critical if no other means of

communications are available, but we assume that since all units are manned, it is possible to contact them to assess the situation in case of loss of signal.

Since the devices are used to track manned units, we assume that they are constantly under some form for physical control, so that threats to platform and storage that require physical access to the device, do not constitute a high risk. Network threats are of course very relevant for both C, I and A of the information when reported to the C2 center, so lack of network protection constitutes a high risk. We are left with environmental threats, that are indeed problematic in this scenario. Communication can be intercepted and used to indirectly locate the units by triangulating it, while the GPS signal can be spoofed before it is measured by the device. Jamming and device malfunctions are likely threats that can compromise availability, but we assessed that it is not a high priority.

4.1.2 Security requirements

The most relevant requirements based on the risk assessment are about the protection of the communication. In particular: protect the confidentiality and integrity of the data transmitted over the communication channel and protect against reporting from unauthorized sources or to malicious servers. Availability should be protected whenever possible by using robust communication channels and techniques to detect GPS spoofing. Some protection against locating the units through signal tracking should be realized by choosing transmission patterns and frequencies that are difficult to detect.

4.2 Area control

This scenario is taken from an extended version[12] of the Anglova scenario,[13] where ground sensors are spread in an area for surveillance and a drone is sent in periodically, or just when needed, to collect the sensor data. The main goal in this scenario is to be able to detect intruders entering a controlled area and possibly track their movements.

4.2.1 Risk assessment

Here, there are no real-time requirements as data is fetched by a drone at given intervals, so availability is not crucial all the time, but only at collection time. Confidentiality of the data is not very critical, as an opponent would already know where they have been, but confidentiality of the position of the sensor is important, as knowing this information could allow to evade detection or find and compromise the sensors. Integrity and trustworthiness of sensor data is the most critical aspect.

Threats that exploit physical access to the devices to compromise the integrity and availability of data either directly by manipulating the storage or indirectly through platform compromise, are very high-risk in this scenario. Consequently, threats to the confidentiality of the device position, like signal tracking, should also be managed so that it is not trivial to trace and access the devices. Threats to the network are all relevant here, except maybe data confidentiality, but also environmental threats to the availability of the communication during data collection or the location confidentiality. Threats to availability in the form of deletion of data or destruction of the device are of course a risk, but less critical either than fake measurements, as lack of data or contact with the device can in itself expose the presence of an intruder.

4.3 Security requirements

The main security requirements for the IoT devices in this operation, are about the protection of the data integrity, both in storage and when communicated to the drone, and about the protection of the device location. The first can be achieved by storing data in a way that any modification can be detected, no false data can be generated and made look genuine, and making sure that only authorized devices can report to the authorized drone. Confidentiality of the device position can be protected to a certain extent by requiring minimal radio activity, but a radio signal with sufficient reach is also needed to allow the data collection.

5. TECHNOLOGY, EXPERIMENTS AND BOTTOM-UP ASSESSMENT

In this section, we present specific technologies that have been evaluated through experiments, namely the LoRaWAN protocol, and two of its implementations used by, respectively, Pycom and Raspberry Pi devices. We use the collected data together with additional available documentation on the devices, to derive an assessment of their security capabilities. The experiments were performed in the context of the unit tracking scenario, so only the GPS component was tested, but one of the strengths of the framework is that the collected data can be stored and systematized in the bottom-up modules for reuse in other assessments. Therefore, we can speculate how the same technology and device could fare when using a motion sensor, rather than a GPS sensor, in the Area control scenario.

5.1 LoRaWAN

Often, the terms LoRa and LoRaWAN are used interchangeably, but there is an important distinction between the two: LoRa defines the layer 1 (physical layer) standard, whereas LoRaWAN adds on the remaining functionality (MAC layer and application standards). Whereas there is no security built into LoRa itself (apart from the chirp mechanism of the protocol, which makes it somewhat resilient towards interference), LoRaWAN defines an encryption scheme to protect the payload when it is in transit. LoRa excels in the areas of battery life, cost efficiency, coverage and range. Further, it provides very good deployment options, since you can deploy your own LoRaWAN without relying on existing infrastructure.

For the two operational scenarios we are targeting in this paper, unit tracking and area control, LoRaWAN's characteristics make it a seemingly fitting protocol from a communication viewpoint. So, we discuss the security-relevant aspects of this protocol.

Figure 3 gives an overview of a typical LoRaWAN deployment. Here, multiple end-devices may communicate with one or more gateways using sub-GHz radio frequencies (RF). The frequencies being used vary across the world, according to what is set aside for unlicensed spectrum. For example, in USA LoRa uses 915 MHz, whereas in the EU LoRa most commonly uses 868 MHz, though 433 MHz is also allowed. The devices can be grouped in one of three communication classes (A, B, and C), dependent on how they approach interacting with the gateway:

- A: Can only receive a message at the time the device is sending a message.
- B: Same as A but listens for incoming messages on regular intervals.
- C: Continuously listens for incoming messages.

For the unit tracking scenario, class A is sufficient since information is flowing from the end device (GPS or other sensor) to the application server. For the area control scenario, B or C may be more appropriate.

The addressing approach of LoRaWAN devices is as follows:

- Each device has a unique 64 bits hardware ID called a *DevEUI* (analogous to the MAC address of an IP device).
- Further, each device has a 32 bits address assigned (or chosen specifically) for use on the network. This is called the *DevAddr* (analogous to an IP address).
- Finally, the 64 bits *AppEUI* uniquely identifies the application provider of the device.

Regardless of class, once information is received in the gateway, it may transfer it onto a back-haul network (a typical approach being a 4G Internet connection). Ultimately, the information ends up in an Application Server, which can decipher and make use of the payload. The payload is secured in transit by employing both a network session key and an application session key:

- The network session key (NwkSKey) is used to encrypt the whole frame and sign the message, allowing the network server to verify the identity of the sender.

- The payload is encrypted with the application session key (AppSKey). The Network Server does not need this key.

These characteristics give both mutual authentication of device and servers, and confidentiality and integrity protection during communication. The way keys are generated, used and stored, is critical for how secure this approach really is.

There are two main approaches to activating a device on a LoRaWAN network: Activation By Personalization (ABP) and Over-The-Air Activation (OTAA).

Of these two, ABP is the simplest approach. Typically, ABP involves hardcoding the DevAddr as well as the security keys (NwkSKey and AppSKey) in the device. The benefit of this approach is that devices are ready to use as soon as they are powered on, there is no OTAA procedure involved. The drawback is the static nature of this approach, which requires pre-planning and pre-distribution of keys and addresses.

OTAA is considered the preferred and most secure way to connect a device to a LoRaWAN gateway. Here, devices perform a join-procedure with the network that involves assigning a dynamic DevAddr as well as negotiating security keys. Devices are pre-loaded with a 128 bit *AppKey*, which must not be confused with the AppSKey which is another entity. The OTAA process starts with a LoRa device sending a JOIN_REQUEST signed with the aforementioned AppKey. The request contains the AppEUI, DevEUI and a DevNonce (a randomly generated number). The Network Server calculates the AppSKey and NwkSKey based on the JOIN_REQUEST it receives and some additional information (e.g., AppNonce, NetID). Then, the Network Server can issue a JOIN_ACCEPT including AppNonce (which is another random number). Finally, the device receives the JOIN_ACCEPT (encrypted with the AppKey), from which it can obtain the DevAddr, NetID, and some other information. At this point, both the device and the server have the same information, so that the device also can calculate the NwkSkey and AppSKey. As we can see, at no point are keys exchanged over the air, only the parts needed to perform the calculation of said keys. The idea here is that even if it is possible to intercept the air traffic, it should be impossible for a third party to get the keys. LoRaWAN uses 128-bit AES for its encryption.

Since, in practice, some key will have to be pre-loaded on the device in every case, we consider only the OTAA approach, which gives most security. As long as the key on the device is physically secure, this protocol provides good security for data transmission. Experiments on the performance of actual implementations of this protocol on different devices, are described in the next section.

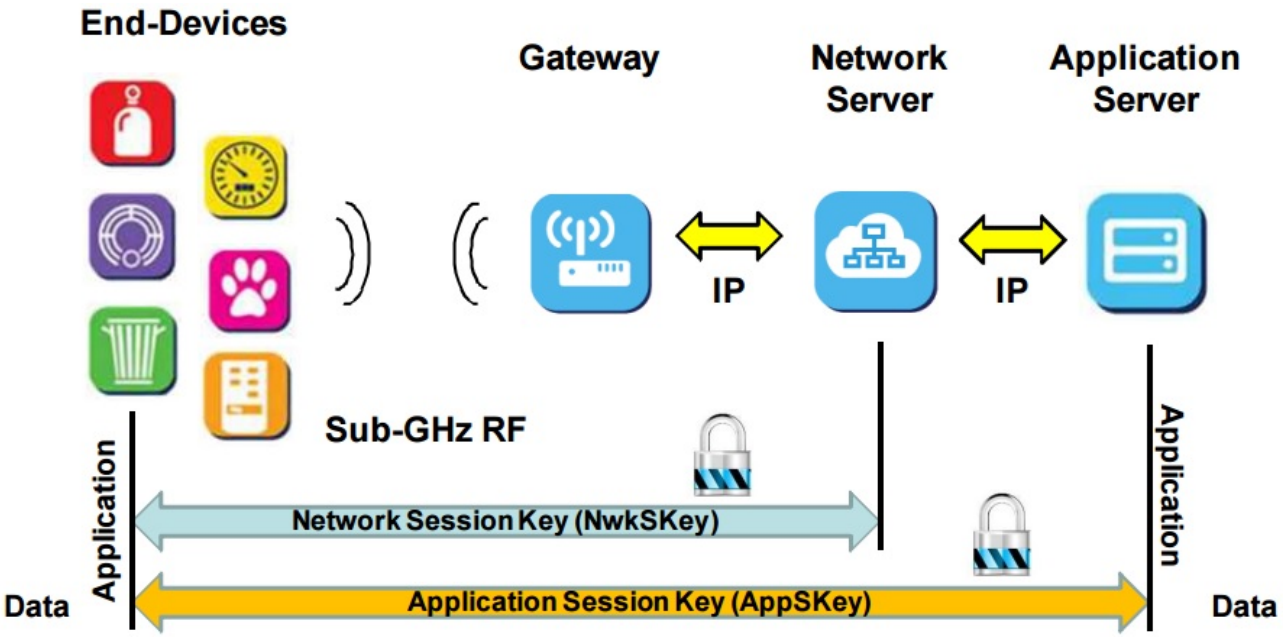

Figure 3. The LoRaWAN architecture (adapted from[14])

5.2 IoT devices

We present Pycom and Raspberry Pi below, their setup for the experiments described in Section 5.3 and their technical specifications.

5.2.1 Pycom

The LoPy4 is a quadruple bearer development board supporting LoRa, Sigfox, WiFi, and Bluetooth.[15] This board supports one programming language, MicroPython. The LoPy4 is merely a communication module, so it needs to be coupled with another board from Pycom. Different use cases will require different boards, e.g., the Pytrack[16] board for unit tracking applications, or the PIR motion sensor[17] for area control. In addition one would also need a case, a battery (or other power source), a MicroPython program to make use of the provided tracking and communication services, and an external LoRa antenna to get a complete tracking device.

Key LoPy4 device specifications are as follows (see[15] for complete details):

- Processing: Espressif ESP32 chipset, with main processor entirely free to run the user application
- LoRa Specification: Semtech LoRa transceiver SX1276, LoRaWAN stack, Class A and C devices
- Interfaces: 2 x UART, SPI, 2 x I2C, I2S, micro SD card
- Security: SSL/TLS support, WPA Enterprise security
- Hash/encryption: SHA, MD5, DES, AES
- Memory: 4 MB RAM, 8 MB flash

5.2.2 Raspberry Pi

The Raspberry Pi is a prolific IoT prototyping platform. There exist several different devices with varying form factors and capabilities, including the Raspberry Pi 3 Model B that we are using. Key specifications are as follows (see[11] for complete details):

- Processing / Memory: Quad Core 1.2 GHz Broadcom BCM2837 64 bit CPU, 1 GB RAM
- Connectivity: 40-pin extended GPIO, 4 USB 2 ports, 4 Pole stereo output and composite video port, full size HDMI, CSI camera port, DSI display port, Micro SD card compatibility
- Networking: 100 Base Ethernet, BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board

This device has no operating system (OS) built into it, as opposed to Pycom. It is a generic ARM-based computer, and an OS like Raspbian Linux is to be installed through the MicroSD card or USB. Then, it is possible to develop advanced applications using the programming language of your choice. The security offered is also almost non-existent and needs to be implemented at the OS and application layer, but additional security modules like Zymbit[18] could greatly improve this aspect. The Raspberry Pi has no built in LoRa or GPS support, but this can be added on by obtaining a 3rd party offering like the Dragino HAT[19] that we have been using. Key properties of the Dragino HAT are (see[19] for complete details):

- LoRa specification: SX1276/SX1278 transceiver, either 868 MHZ, 433 MHZ, or 915 MHZ (pre-configured in factory)
- Compatible with Raspberry Pi 2/3 Model B
- Positioning compliant with GPS, SBAS.

5.3 Experiments

An example of recent experiments performed using the LoPy4 and Pytrack combination is,[10] where they, as mentioned in the related work section, achieved ranges up to 5 Km with LoRaWAN in an urban environment. This is in contrast to the maximum possible range, which Pycom lists as 40 Km (node range) or 22 Km (using the LoPy4 as a nano-gateway). The experiments in[10] used the Pycom devices as nodes, and had a separate, industry-strength gateway deployed, and should theoretically have a maximum range of 40 Km.

Together with an industry-strength gateway, using the Raspberry Pis for unit tracking, we have achieved usable ranges of up to 6 Km in an urban environment using the 868 MHz version.[20] Dragino provides Python code examples and a Python library to work with the HAT and the LoRaWAN stack. We have been using this library on the Raspbian Stretch Lite OS for our experiments. In the same setting, we have been able to detect and eavesdrop on the LoRaWAN communication. This confirmed that encryption was successfully enabled and prevented trivial eavesdropping attempts, but also that signal detection was possible when up to 1 Km away from the nodes (due to practical limitations on positioning of equipment, theoretically only limited by the transmission range of LoRa).

5.4 Security assessment

Based on the information we can gather on the security of the devices and technology used, and their experimental setup, we can derive a preliminary security assessment. Ideally, this should be done in a systematic manner by using some reference model that can be incrementally refined as more information is gathered, but the framework is not that mature yet. For the purpose of this paper, we use the attack surface components shown in Table 1.

Considering the platform, there does not seem to be any intrinsic security. It is possible to connect to the LoPy4 without any authentication if physical access is gained. Raspberry Pi has the possibility to enable password protected access and SSH connections, but the SD-card can be physically removed with data on it, while attached sensors can be replaced with malicious ones as there does not seem to be any authentication. Communication seems to be adequately protected when using LoRaWAN. Both mutual authentication, confidentiality and integrity can be achieved. LoRa has some protection against interference due to the chirp spread spectrum technique it employs. Also, there is the possibility to reduce the frequency of communication, so that it may be more difficult to intercept in some modes, although not impossible, as experiments showed. Experiments also showed that the effective reach of the tested implementations should allow the communication with a passing drone, but targeted experiments are necessary to confirm this. The infrastructure setup would need to be more carefully analyzed to identify possible security gaps that may be introduced there. Being a commercial protocol, LoRaWAN may also easily blend with other local traffic if used in populated urban areas. Storage protection can be achieved through encryption since LoPy4 provides encryption libraries with AES, and Raspberry Pi can run an advanced OS that can provide the same, or better, functionality. However, no dedicated protected storage is offered for cryptographic keys, so this approach may not be very effective for devices in uncontrolled areas.

6. EVALUATION OF RESIDUAL RISK

Here, we put together the assessments performed so far to derive some conclusion on whether the given devices are, or can be made, secure enough in the described scenarios. We summarize the assessment in Table 2.

The main weakness of the IoT devices we considered, is the lack of physical platform protection, which also means that the security provided for storage protection may not be very effective in practice. On the communication side, instead, the LoRaWAN seems to provide sufficient security for most applications, given that keys and setup are chosen carefully. This means that for the unit tracking scenario, where threats exploiting physical access are not a very serious risk, both the overall risk of using these two commercial devices may boil down to which one is cheaper, easier to configure and most reliable. However, some additional security could be implemented on the C2 center to detect attempts of GPS spoofing by analyzing historical data and jamming by detecting suspicious loss of communication and securing alternative communication channels. For the area control scenario, the risk of using these devices as they are, is too high. If the devices are located before they can relay their measurements to the drone, they could be destroyed or compromised due to the lack of physical security. The worst case is if the measurements can be replaced with fake ones showing no activity in

the area. Here, a possible additional security control could be the addition of an hardware security module to store keys that also has some tamper-resistant capability, like Zymbit.[18] This is available only for Raspberry Pi, so the LoPy4 device would need a different solution. A possibility can be to implement an encryption scheme to provide perfect forward secrecy (PFS), like the one proposed in.[21] In this way, we would at least have an archive of measurements that cannot be forged. How much this approach would reduce the risk of using a LoPy4 device, would depend on the frequency at which measurements are collected by the drone, but given the system specifications of the device, it seems at least possible to implement and test.

Table 2. Overall security assessments.

UNIT TRACKING		
Top-down security requirements	Bottom-up assessment	Additional security
Protect against GPS spoofing	No protection	Anomaly detection at C2
Protect integrity of reporting	Integrated protection at link layer	Secure key management
Protect confidentiality of reporting	Integrated protection at link layer	Secure key management
Protect against false reporting	LoRaWAN integrated authentication	Not needed
Protect availability of communication	Robust and wide coverage	Alternative communication
Protect against indirect tracking	Blending with other LoRa traffic	Not identified
AREA CONTROL		
Top-down security requirements	Bottom-up assessment	Additional security
Protect integrity of stored data	No protection against physical access	Use security module/PFS
Prevent location of devices	Transmit only on request	Not identified
Protect integrity of reports	LoRAWAN encryption	Not identified
Protect against injection of false data	Use mutual authentication	Not identified

7. DISCUSSION AND CONCLUSIONS

In this paper we presented the first version of a framework to perform effective, quick and repeatable risk and security assessments for the application of IoT devices to military operations. The goal was to get a first evaluation of its potential, and this paper showed that one can quickly produce preliminary security assessments and identify additional security controls that can be tested through further design and experimentation activities. A more technical analysis could also have been performed by taking into account the technical specifications of the protocol and devices to speculate on other potential security challenges. For instance, the use of different LoRa frequencies in different countries could become an interoperability threat in federated missions, while the use of a specific OS or libraries could be linked to known security vulnerabilities that could be exploited in a certain setting.

Future work will focus on refining the framework by defining more formally how the different modules should support the assessment process, possibly by formalizing and automating some of the steps. Secondly, one should populate the modules with more refined standards, catalogs of threats and controls, best practices and patterns to guarantee that well-established security solutions are used whenever possible and to manage the right risks. The bottom-up modules should also be populated with the systematized results of experiments and other assessments. Finally, these assessment will have to be seen in the larger context of mission assurance, where further security aspects are taken into consideration.

REFERENCES

[1] Oxford Dictionary, "Definition of Internet of Things," https://en.oxforddictionaries.com/definition/internet_of_things, Accessed 2020-02-14.

[2] C. Bedell, "Why Makerspaces Are Democratizing IoT Development," https://www.iotworldtoday.com/2019/08/10/why-makerspaces-are-democratizing-iot-development/, Accessed 2020-02-14.

[3] P. Fraga-Lamas, T. M. Fernandez-Carames, M. Suarez-Albela, L. Castedo, and M. Gonzales-Lopez, "A Review on Internet of Things for Defense and Public Safety," Sensors2016,16, 1644; doi:10.3390/s16101644, 5 October 2016.

[4] F. T. Johnsen, Z. Zielinski, K. Wrona, N. Suri, C. Fuchs, M. Pradhan, J. Furtak, B. Vasilache, V. Pellegrini, M. Dyk, M. Marks, and M. Krzyszton, "Application of IoT in Military Operations in a Smart City," 2018 International Conference on Military Communications and Information Systems (ICMCIS), Warsaw, Poland, 22 - 23 May 2018.

[5] M. Pradhan, F. T. Johnsen, M. Tortonesi, and S. Delaitre, "Leveraging Crowdsourcing and Crowdsensing Data for HADR Operations in a Smart City Environment," IEEE Internet of Things Magazine, Volume: 2 , Issue: 2 , June 2019.

[6] K. Wrona, "Securing the internet of things a military perspective," In Proceedings of the IEEE 2nd World Forum on Internet of Things (WF-IoT), Milan, Italy, 12-14 December 2015.

[7] F. Mancini, S. Bruvoll, T. Verhoogt, R. Wiegers, J. Melrose, R. Been, R. Ernst, K. Rein, and F. Leve, "Securing autonomous and unmanned vehicles for mission assurance*," in *2019 International Conference on Military Communications and Information Systems (ICMCIS)*, 2019, pp. 1–8.

[8] F. Mancini, S. Bruvoll, J. Melrose, L. Mailloux, F. Leve, S. Fioravanti, D. Merani, R. Ernst, K. Rein, and R. Been, "A security reference model for autonomous vehicles in military operations," in *Submitted*, 2020.

[9] K. Mekki, E. Bajica, F. Chaxel, and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," ICT Express Volume 5, Issue 1, March 2019, Pages 1-7.

[10] J. Michaelis, A. Morelli, A. Raglin, D. James, and N. Suri, "Leveraging LoRaWAN to Support IoBT in Urban Environments," IEEE World Forum on Internet of Things (WF-IoT) 2019, 16 April 2019, Special session on military applications of IoT, Limerick, Ireland.

[11] Raspberry Pi Foundation, "Raspberry Pi 3 Model B," https://www.raspberrypi.org/products/raspberry-pi-3-model-b/, Accessed 2020-03-02.

[12] N. Suri, K. M. Marcus, C. van den Broek, H. Bastiaansen, P. Lubkowski, and M. Hauge, "Extending the Anglova Scenario for Urban Operations," 2019 International Conference on Military Communications and Information Systems (ICMCIS), Budva, Montenegro, 14 - 15 May 2019.

[13] N. Suri, J. Nilsson, A. Hansson, U. Sterner, K. Marcus, L. Misirlioğlu, M. Hauge, M. Peuhkuri, B. Buchin, R. in't Velt, and M. Breedy, "The angloval tactical military scenario and experimentation environment," 2018 International Conference on Military Communications and Information Systems (ICMCIS), Warsaw, Poland, 22 - 23 May 2018.

[14] P. Mannion, "Connect the Dots: LoRaWAN Makes IoT Deployment Easy," https://www.iotsolutionprovider.com/industrial/connect-the-dots-lorawan-makes-iot-deployment-easy, accessed 2020-02-27.

[15] Pycom Ltd., "LoPy4," https://pycom.io/product/lopy4/, Accessed 2020-03-02.

[16] ——, "Pytrack," https://pycom.io/product/pytrack/, Accessed 2020-03-02.

[17] Kiwi Electronics, "PIR sensor," https://www.kiwi-electronics.nl/PIR-Motion-Sensor, Accessed 2020-03-02.

[18] Zymbit, "Security Module for Raspberry Pi," https://www.zymbit.com/blog-security-module-raspberry-pi/, Accessed 2020-03-02.

[19] Dragino, "LoRa GPS HAT for Raspberry Pi," Retrived July 22nd, 2019, from http://www.dragino.com/products/module/item/106-lora-gps-hat.html.

[20] P. Ø. Puente and F. T. Johnsen, "Asset tracking experiments with lorawan," FFI-Eksternnotat 19/01760, 2019.

[21] M. A. Simplicio, L. H. Iwaya, B. M. Barros, T. C. M. B. Carvalho, and M. Näslund, "Secourhealth: A delay-tolerant security framework for mobile health data collection," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 2, pp. 761–772, 2015.