# Cybersecurity Issues in Robotic Platforms

Adrián Campazas-Vega[1,*], Alberto Miguel-Diez[1], Mario Hermida-López[1], Claudia Álvarez-Aparicio[1], Ignacio Samuel Crespo-Martínez[1] and Ángel Manuel Guerrero-Higueras[1]

[1] *Grupo de Robótica de la Universidad de León, Campus de Vegazana, 24071 León, Spain*

**Abstract**

The use of robots has increased dramatically in recent years. Currently, there are multiple types of robots, from service robots, designed to help people in any kind of environment (home, work, hospitals...), to quadruped platforms, developed for critical infrastructures or the military field. Security in those platforms is crucial, since robots present vulnerabilities, they can pose a risk to both their integrity and that of the people/objects around them. In this work, a security evaluation of the Unitree A1, a quadruped robot, and the humanoid robot Pepper has been carried out, to know the security flaws that may be present, as well as the implications that it may have for the user, the environment, or the integrity of the robot. The final goal of the work is that the vulnerabilities found will be taken into account by other researchers or companies that develop that kind of robot and take into account those security problems.

**Keywords**

Pentesting, robot, security, Unitree A1, Pepper

## 1. Introduction

The use of robots has exponentially increased in the last decade. Throughout the year 2022, the utilization and deployment of industrial robots increased by 40% in the United States and 6% in Spain, according to the Spanish Association of Robotics (AER) [1]. Industrial robotics has traditionally focused on the precise repetition of tasks, surpassing the capabilities of a human being. However, in recent years, there has been a particular emphasis on the development of robotic platforms capable of performing tasks that are difficult or dangerous for humans. In this regard, the most impactful robotic platforms are quadruped robots. These robots are characterized by supporting their weight on four legs, typically mimicking the morphology of a dog. The design of these devices offers advantages over bipedal robots due to their versatility in adapting to various types of terrains. The characteristics of quadruped robots enable them to undertake tasks considered challenging or hazardous for humans. These tasks include bomb inspection and deactivation, radiation detection, and critical infrastructure maintenance. In addition to their civilian applications, these robots are actively utilized in the military domain [2]. Similarly, the use of service robots has also significantly increased in recent years. These robots are designed to interact and communicate with humans to assist in the completion of everyday tasks.

Similarly, to other types of devices, cybersecurity in robotic environments is an important aspect that becomes critical when a robot is involved in highly sensitive tasks or interacts with people. Many issues with these platforms arise because manufacturers often prioritize manufacturing cost or design over conducting product security testing [3]. In addition to the lack of device security by manufacturers, it is worth noting that most of these robotic platforms are "plug and play," meaning that end users often do not pay proper attention to configuring the device correctly. This includes changing default passwords, which poses an additional security challenge.

This paper aims to address some of the security issues presented by both quadruped robotic platforms and social robots. Specifically, a security evaluation has been conducted on the quadruped robot Unitree A1 and the semi-humanoid robot Pepper, with the objective of identifying potential vulnerabilities and risks that could affect both humans and the robot itself, as well as the environment in which it is deployed. The severity of the discovered vulnerabilities has been assessed using the CVSSv3 (Common Vulnerability Scoring System version 3) standard. This work and the methods employed can serve as a starting point for other researchers interested in evaluating the security risks of other models of quadruped robots and social robots.

The rest of the article is organized as follows: In Section 2, related works are presented. Section 3 introduces

*BISEC'23: 14th International Conference on Business Information Security, November 24, 2023, Niš, Serbia*

[*] Corresponding author.

acamv@unileon.es (A. Campazas-Vega); amigud00@estudiantes.unileon.es (A. Miguel-Diez); mherml00@estudiantes.unileon.es (M. Hermida-López); calvaa@unileon.es (C. Álvarez-Aparicio); icrem@unileon.es (I. S. Crespo-Martínez); am.guerrero@unileon.es (Á. M. Guerrero-Higueras)
0000-0001-8237-5962 (A. Campazas-Vega); 0000-0002-7465-8054 (C. Álvarez-Aparicio); 0000-0002-3154-0144 (I. S. Crespo-Martínez); 0000-0001-8277-0700 (Á. M. Guerrero-Higueras)

CEUR Workshop Proceedings (CEUR-WS.org)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

the architecture and characteristics of the robots Unitree A1 and Pepper, along with the method for assessing the severity of discovered vulnerabilities. Section 4 provides details on the various experiments conducted and the implications of exploiting the vulnerabilities in a real-world environment. Finally, Section 5 offers the current conclusions.

## 2. Related Works

Despite the growing popularity of quadruped robots, there is limited research on the cybersecurity of these robots. Most research in this field focuses on the physical security of robots, such as collision prevention [4] and stability on different terrains [5]. However, there are some works that examine overall security in robotic devices. In [6], the authors analyzed potential security issues that different types of robots might have and listed some generic recommendations that could be implemented to enhance the overall security of robotics. One of the conclusions reached by the authors is that cyberattacks on robots used in critical infrastructures and military environments are the most damaging and dangerous. It's important to note that the current use of quadruped robots primarily focuses on these two areas. Another work related to robotic security is presented in [3]. In this work, the authors identified security threats in the field of robotics, classified them based on the affected layer of the robot's architecture, and analyzed their impact and potential countermeasures. Other works, such as [7] and [8], discuss security issues associated with ROS (Robot Operating System). ROS is a set of software libraries and tools that help create applications for robots. While Pepper and Unitree A1 do not come with ROS by default, it is possible to install ROS on the latter.

Finally, regarding the specific analysis of the Pepper robotic platform, in [9], the authors conducted a security evaluation of the semi-humanoid robot "Pepper" from SoftBank Robotics. The authors demonstrated that this robot had critical vulnerabilities that needed to be addressed by the manufacturer. This article expands on the work done in [9], confirming that years later, the vulnerabilities identified by the authors still exist and uncovering new vulnerabilities in the platform.

## 3. Materials and Methods

In this section, the characteristics of the robots analyzed in this work are presented. Additionally, the methodology used to conduct the experiments and the evaluation method for these experiments are described.

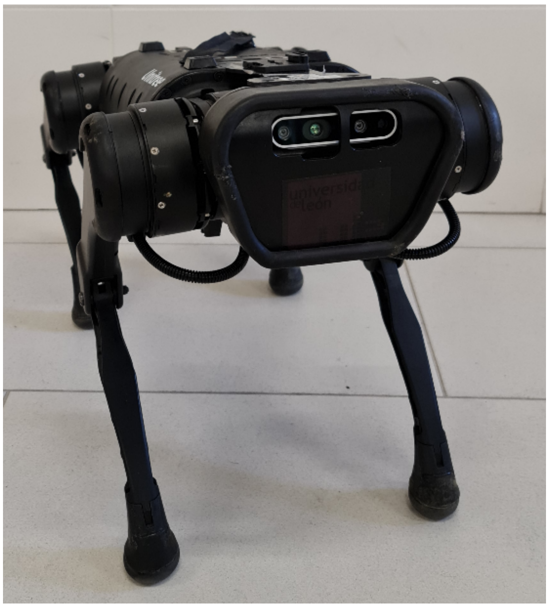


**Figure 1:** Unitree A1 of the Robotics Group of the University of León.

### 3.1. Unitree A1

As mentioned in Section 1, to conduct the cybersecurity evaluation of quadruped robots, the Unitree A1 robot, as shown in Figure 1, has been utilized. The Unitree A1 is manufactured by Unitree Robotics, a Chinese company that has been producing quadruped devices since 2016 [10].

The Unitree A1 robot can reach a maximum speed of 3.3 m/s at a particular moment and can carry objects with a maximum weight of 5 kg. Additionally, it is equipped with sensors that enable it to maintain proper balance during operation, preventing the robot from falling on uneven terrain. The device has a battery life ranging from 1 to 2.5 hours, depending on the mode in which it is used [11].

Regarding the cameras and sensors, the Unitree A1 is equipped with a RealSense camera [12], located on its "head." This camera features a depth sensor that utilizes a combination of infrared and laser technologies to measure the distance between objects and the camera. This enables it to capture 3D images and detect objects in real-time. In the field of robotics, these types of cameras are used to implement autonomous functions in the robot, allowing it to navigate around obstacles and create a 3D map of the area in which the robot is deployed [13, 14]. At the connectivity level, the quadruped robot has several ports on the upper part of its "body" that the user can utilize to interact with various interfaces of the robot. These connections include four USB ports, two HDMI ports, and two Ethernet ports.

Teleoperation of the robot can be performed using a mobile application developed by the manufacturer or by using the controller that comes with the robot. The controller includes two joysticks and a directional pad (D-pad) for easy robot maneuvering. According to the manual, the controller connects directly to the robot's control board via radio frequency. On the other hand, Unitree's mobile application is compatible with both iOS and Android devices. The app allows users to control the robot, view the real-time camera feed, and utilize a simulator of the Unitree A1. However, despite the robot being available for commercial use since 2020, some features of the app may not work correctly or require specific parameter configurations. Furthermore, Unitree provides users with a Software Development Kit (SDK) to develop custom code for the robot. This SDK enables developers to create their own applications and functionalities for the Unitree A1.

## 3.2. Pepper

Pepper is the world's first social humanoid robot capable of recognizing human faces and basic emotions. It is optimized for interaction and can engage with people through conversation or its touchscreen interface. Pepper is designed for intuitive and natural interaction. It finds common applications in various fields such as hospitality, retail, healthcare, education, entertainment, and personal assistance. Its appearance is depicted in Figure 2.

Pepper has 20 degrees of freedom to achieve more natural and expressive movements. Additionally, it features voice recognition available in 15 languages and perception modules to recognize and interact with the person in front of it. In terms of physical sensors, the robot is equipped with touch sensors, LEDs, microphones for multimodal interaction, infrared sensors, bumpers, an inertial unit, and 2D and 3D cameras to enable autonomous and omnidirectional navigation. Pepper provides an API that allows for the development of custom applications and functionalities for this robotic platform.

## 3.3. Evaluation

To assess the severity of the discovered vulnerabilities, the Common Vulnerability Scoring System (CVSS) version 3 has been employed [15]. CVSS, or Common Vulnerability Scoring System, is an open and widely used framework that defines metrics for communicating the characteristics, impact, and severity of vulnerabilities affecting security elements. It provides a standardized way to evaluate and communicate the seriousness of security vulnerabilities.

CVSSv3 categorizes vulnerabilities with a numerical value between 0 and 10. A vulnerability with a score between 0.1 and 3.9 is considered to have low severity. Vulnerabilities with a score between 4.0 and 6.9 are classified as having moderate severity. Finally, vulnerabilities with a score between 7.0 and 10.0 are considered to have high severity. This scoring system provides a clear way to assess the seriousness of vulnerabilities and helps organizations prioritize their remediation efforts.

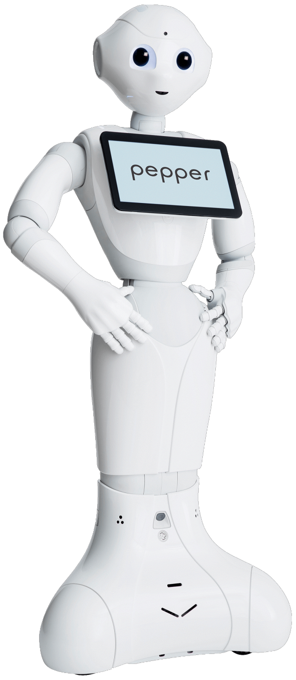


**Figure 2:** Appearance of the Pepper service robot.

CVSS defines metrics to assess the likelihood that a vulnerability will be exploited. The metrics defined by the CVSSv3 standard can be seen in Table 1.

## 3.4. Methodology

The methodology used for the analysis of robotic platforms is similar to that employed in conventional computer systems. Below, we outline the three stages carried out to assess the security of the Unitree A1 robot and the Pepper service robot:

- **Information Gathering**: In this step, information is collected about the robotic platform, including the type of hardware and sensors used by the device, the operating system it runs on, the services it executes, and the nature of the communications that take place.
- **Vulnerability Analysis**: Tests are conducted to identify vulnerabilities in the robotic system. This analysis encompasses both hardware and

**Table 1**
Metrics associated with the CVSS vector in version 3

| Symbol | Description |
|---|---|
| **AV** | **Attack Vector**: Determines how the vulnerability can be exploited, assessing the accessibility requirements. The values of this metric are:<br>• Network (N)<br>• Adjacent (A)<br>• Local (L)<br>• Physical (P) |
| **AC** | **Attack Complexity**: Determines the attack complexity required to make use of the vulnerability. The values of this metric are:<br>• Low (L)<br>• High (H) |
| **PR** | **Privileges Required**: Determines the level of privileges an attacker must have before he can successfully exploit a vulnerability. The values of this metric are:<br>• None (N)<br>• Low (L)<br>• High (H) |
| **UI** | **User Interaction**: Determines if user intervention is necessary for successful exploitation of the vulnerability. The levels of this metric are:<br>• None (N)<br>• Required (R) |
| **S** | **Scope**: Determines whether successful exploitation of the vulnerability can indirectly affect other components outside the scope of the system or application. The values of this metric are as follows:<br>• Unchanged (U)<br>• Changed (C) |
| **C** | **Confidentiality Impact**: Confidentiality is the ownership of a document, message or data that is only authorized to be read or understood by certain persons or entities. The values of this metric are as follows:<br>• None (N)<br>• Low (L)<br>• High (H) |
| **I** | **Integrity Impact**: Integrity is the property of a document, message or data that guarantees the veracity of the information. The values for this metric are as follows:<br>• None (N)<br>• Low (L)<br>• High (H) |
| **D** | **Availability Impact**: Availability is the property of a system, service, or application that is accessible without impediments. The values for this metric are as follows:<br>• None (N)<br>• Low (L)<br>• High (H) |

software aspects, as well as the systems deployed by the robot.

- **Exploitation of Identified Vulnerabilities**: Finally, identified vulnerabilities are exploited to determine the extent to which these security flaws pose a risk to the safety of the robot itself and its surrounding environment.

# 4. Experimentation and Discussion

The evaluation conducted on these robots aims to identify vulnerabilities that may be present in the devices and could be extrapolated to other robotic platforms. The following will demonstrate how both robots share common vulnerabilities. All vulnerabilities listed below are associated with an impact vector generated using the CVSSv3 standard, as discussed in Section 3. The discovered vulnerabilities, which are explained below, are presented in Table 2.

## 4.1. Common vulnerabilities in both robots

In this subsection, we present the vulnerabilities that are common to both robots.

**Table 2**
Vulnerabilities of the evaluated robots

| Vulnerability | Impact | Robot |
|---|---|---|
| Lack of protection against brute force attacks in SSH protocol | High | Unitree A1<br>Pepper |
| Lack of verification against MiTM attack | High | Unitree A1<br>Pepper |
| Denial of service to the robot's Web server | Moderate | Unitree A1<br>Pepper |
| Unsecured physical ports | High | Unitree A1 |
| Web server without authentication | Moderate | Unitree A1 |
| API access without authentication | High | Pepper |
| Communication with the web server without encryption | Moderate | Pepper |

### 4.1.1. Lack of protection against brute force attacks in SSH protocol

One way to access the embedded computers inside the robot is through the SSH protocol. This connection allows for configuring certain aspects of the robot, such as the AP password, and even controlling the robot using the installed SDK. Both the Unitree A1 robot and Pepper do not implement security measures to prevent brute-force attacks on the SSH servers installed in the robot. To verify that the SSH servers are vulnerable to dictionary attacks or brute-force attacks, the open-source tool Hydra has been used [16].

If an attacker gains access to the robot's internal computers, they could potentially control the robot remotely and even delete system files, rendering the device inoperable. Furthermore, since the default password for both devices is considered insecure today and is present in a wide range of online dictionaries, this vulnerability is deemed severe with a score of 9 and the following CVSS vector: AV:A/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:H.

### 4.1.2. Lack of verification against MiTM attack

Neither the quadruped robot Unitree A1 nor the social robot Pepper implement security measures to prevent an attacker with access to the robot's network from performing a Man-in-the-Middle (MitM) attack. This would allow the attacker to intercept unencrypted communications and manipulate them at will. Here's an example of the vulnerability in the Unitree A1 robot: The A1 robot deploys a web server that serves images from the robot's camera, allowing an operator to teleoperate the device remotely.

An attacker who has access to the network deployed by the robot can carry out a MitM attack, altering the video transmission from the robot's camera with another feed controlled by the attacker, without the victim noticing any difference. If the robot is used in critical situations, the operator controlling the robot will not perceive the actual situation, potentially enabling an attacker to cause harm to the robot itself or its surrounding environment.

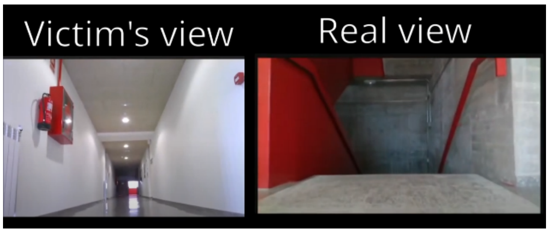


**Figure 3:** On the left, view of the teleoperator after being attacked. On the right, real image of the robot's situation.

To exploit this vulnerability, an ARP Spoofing attack was conducted using the "arpspoof" tool [17]. This attack is considered one of the most dangerous on LAN networks [18]. The attacker manipulates both the robot's and the victim's ARP tables, associating their MAC address with the victim's IP address, thereby redirecting all traffic to a machine controlled by the attacker. Subsequently, the attacker redirects the traffic arriving from the user to a web server identical to the robot's but under the attacker's control. In this case, the web server deployed by the Unitree is MJPG-Streamer, which is publicly available on GitHub [19].

The consequences of such attacks can be critical in certain environments. For instance, in Figure 3, can see that the person operating the robot perceives an obstacle-free corridor, while in reality, the robot is in a hazardous situation near a set of stairs.

This vulnerability has a high impact with a score of 8.0 and the following associated CVSSv3 vector: AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H. A video has been created to replicate the experiment performed [20].

### 4.1.3. Denial of service to the robot's Web server

The web servers deployed by both robots are vulnerable to denial-of-service (DoS) attacks. The process to exe-

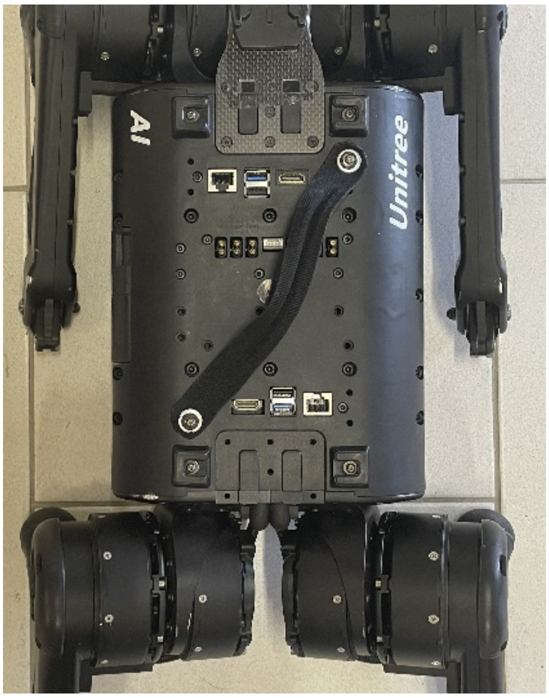


Figure 4: Top view of Unitree A1.

cute this attack is quite similar to the previous one, as it relies on the ARP Spoofing technique in both cases. To exploit this vulnerability, the attacker must manipulate the victim's and robot's ARP tables to intercept traffic. Once the attack is successfully carried out, all packets are received by the attacker, who will then discard these packets, causing the legitimate user to lose the connection to the web server. This vulnerability has a moderate impact with a score of 5.7 and the following associated CVSSv3 vector: AV:A/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H.

## 4.2. Unitree A1 robot vulnerabilities

This section shows vulnerabilities that exclusively affect the Unitree A1 robot.

### 4.2.1. Unsecured physical ports

Figure 4 shows the port distribution of the robot. The main vulnerability lies in the fact that the robot does not request any form of authentication when connected through the provided ports.

The lack of authentication poses several security implications, even without connecting standard input and output devices such as a keyboard and monitor. Currently, there are USB-like devices that function as input and output devices, enabling the execution of commands simply by plugging them in. These devices are referred to as Rubber Ducky [21]. Furthermore, the exposure of USB ports also makes the robot vulnerable to attacks carried out with a USB killer device [22]. This type of device discharges a high-voltage surge, damaging the components of the connected device. This vulnerability has a high impact with a score of 7.5, and the associated CVSSv3 vector is AV:P/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L.

### 4.2.2. Web server without authentication

Access to the live video feed from the robot's camera does not have an authentication system. Therefore, any user connected to the network emitted by the robot can view the real-time image either through the device's web server or via the mobile application. To be considered secure, this functionality should require authentication.

This vulnerability has a moderate impact with a score of 5.7 and the following CVSSv3 vector: AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N.

## 4.3. Pepper robot vulnerabilities

In this section, the vulnerabilities that exclusively affect the social robot Pepper are presented.

### 4.3.1. API access without authentication

The API implemented by Pepper allows for complete control of the device. Access to the API occurs without any form of authentication, so an attacker only needs to be on the same network as the robot. Interaction with the API is done through port 9559 using the Python programming language, although C++ and Java are also supported.

This vulnerability has a high impact with a score of 7.5, and the associated CVSSv3 vector is: AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H

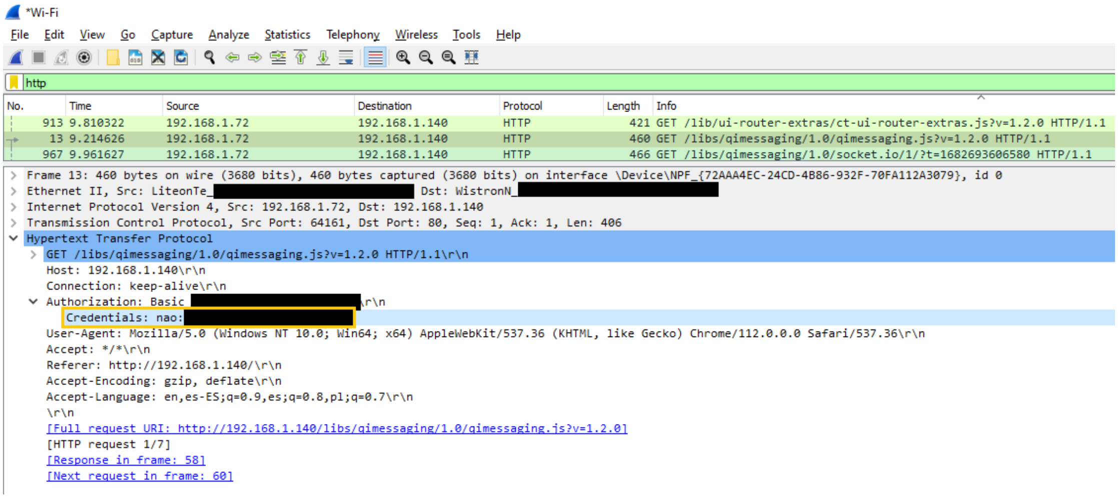
### 4.3.2. Communication with the web server without encryption

The web server used by the robot utilizes unencrypted HTTP communication. An attacker connected to the network can sniff the traffic and obtain the access credentials for the web server, as depicted in Figure 5.

This vulnerability has a moderate impact with a score of 6.5 and the following CVSSv3 vector: AV:L/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:N.

# 5. Conclusions

The use of robotics is becoming increasingly widespread; however, it is essential that progress in this field is accom-



```
No.   Time        Source         Destination     Protocol   Length  Info
  913 9.810322    192.168.1.72   192.168.1.140   HTTP          421  GET /lib/ui-router-extras/ct-ui-router-extras.js?v=1.2.0 HTTP/1.1
   13 9.214626    192.168.1.72   192.168.1.140   HTTP          460  GET /libs/qimessaging/1.0/qimessaging.js?v=1.2.0 HTTP/1.1
  967 9.961627    192.168.1.72   192.168.1.140   HTTP          466  GET /libs/qimessaging/1.0/socket.io/1/?t=1682693606580 HTTP/1.1
```

```
> Frame 13: 460 bytes on wire (3680 bits), 460 bytes captured (3680 bits) on interface \Device\NPF_{72AAA4EC-24CD-4B86-932F-70FA112A3079}, id 0
> Ethernet II, Src: LiteonTe_[redacted] Dst: WistronN_[redacted]
> Internet Protocol Version 4, Src: 192.168.1.72, Dst: 192.168.1.140
> Transmission Control Protocol, Src Port: 64161, Dst Port: 80, Seq: 1, Ack: 1, Len: 406
v Hypertext Transfer Protocol
  > GET /libs/qimessaging/1.0/qimessaging.js?v=1.2.0 HTTP/1.1\r\n
    Host: 192.168.1.140\r\n
    Connection: keep-alive\r\n
  v Authorization: Basic [redacted]\r\n
      Credentials: nao:[redacted]
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36\r\n
    Accept: */*\r\n
    Referer: http://192.168.1.140/\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en,es-ES;q=0.9,es;q=0.8,pl;q=0.7\r\n
    \r\n
    [Full request URI: http://192.168.1.140/libs/qimessaging/1.0/qimessaging.js?v=1.2.0]
    [HTTP request 1/7]
    [Response in frame: 58]
    [Next request in frame: 60]
```

**Figure 5:** Capture of Pepper's traffic showing the robot's plaintext credentials.

panied by a thorough review of potential vulnerabilities in these devices.

In this work, a security evaluation has been conducted on the quadruped robot Unitree A1 and the service robot Pepper. Several potential vulnerabilities have been identified that could be exploited by an attacker to gain unauthorized access to the robot or control its movements and actions. For each of the vulnerabilities discovered in this work, a Common Vulnerabilities and Exposures (CVE) has been requested. The CVE program's mission is to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities.

To continue advancing in the field of robotics, it is necessary to implement security measures such as user authentication and authorization, encryption of device communications, and regular security testing to detect and address potential vulnerabilities in the software of various robotic platforms. It is important to emphasize that the cybersecurity of quadruped and social robots is a critical issue that must be addressed by manufacturers, developers, and users of these devices to ensure their proper functioning and protect them against potential malicious attacks that could pose a security risk to the robot itself or to people in its vicinity.

## Acknowledgment

This research has been partially supported under the grant PID2021-126592OB-C21 funded by MCIN/AEI/10.13039/501100011033 and by ERDF A way of making Europe and under the Grant TED2021-132356B-I00 funded by MCIN/AEI/10.13039/501100011033 and by the "European Union NextGenerationEU/PRTR.

## References

[1] A. E. de Robótica y Automatización, La importancia de la ciberseguridad en la industria 4.0, https://www.aer-automation.com/wp-content/uploads/2023/01/Ciberseguridad_AERPaper.pdf, 2023.

[2] K. Geldenhuys, Killer robots are real, Servamus Community-based Safety and Security Magazine 116 (2023) 20–22.

[3] G. W. Clark, M. V. Doran, T. R. Andel, Cybersecurity issues in robotics, in: 2017 IEEE conference on cognitive and computational aspects of situation management (CogSIMA), IEEE, 2017, pp. 1–5.

[4] R. Singh, T. Bera, Walking model of jansen mechanism-based quadruped robot and application to obstacle avoidance, Arabian Journal for Science and Engineering 45 (2020) 653–664.

[5] Y. H. Lee, Y. H. Lee, H. Lee, L. T. Phan, H. Kang, U. Kim, J. Jeon, H. R. Choi, Trajectory design and control of quadruped robot for trotting over obstacles, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2017, pp. 4897–4902.

[6] C. Cerrudo, L. Apa, Hacking robots before skynet, IOActive Website (2017) 1–17.

[7] S.-Y. Jeong, I.-J. Choi, Y.-J. Kim, Y.-M. Shin, J.-H. Han, G.-H. Jung, K.-G. Kim, A study on ros vulnerabilities and countermeasure, in: Proceedings of the Companion of the 2017 ACM/IEEE International

Conference on Human-Robot Interaction, 2017, pp. 147–148.

[8] R. White, D. H. I. Christensen, D. M. Quigley, Sros: Securing ros over the wire, in the graph, and through the kernel, arXiv preprint arXiv:1611.07060 (2016).

[9] A. Giaretta, M. De Donno, N. Dragoni, Adding salt to pepper: A structured security assessment over a humanoid robot, in: Proceedings of the 13th International Conference on Availability, Reliability and Security, 2018, pp. 1–8.

[10] U. Robotics, Unitree, https://m.unitree.com/, 2022.

[11] U. Robotics, Unitree a1 user manual, https://www.mybotshop.de/Datasheet/UnitreeA1_User_Manual_v1.0.pdf/, 2020.

[12] F. L. Siena, B. Byrom, P. Watts, P. Breedon, Utilising the intel realsense camera for measuring health outcomes in clinical research, Journal of medical systems 42 (2018) 1–10.

[13] J. Bayer, J. Faigl, On autonomous spatial exploration with small hexapod walking robot using tracking camera intel realsense t265, in: 2019 European Conference on Mobile Robots (ECMR), IEEE, 2019, pp. 1–6.

[14] J. Hu, Y. Niu, Z. Wang, Obstacle avoidance methods for rotor uavs using realsense camera, in: 2017 Chinese Automation Congress (CAC), IEEE, 2017, pp. 7151–7155.

[15] INCIBE, Métricas de evaluación de vulnerabilidades: Cvss 3.0, https://.incibe-cert.es/blog/cvss3-0/, 2023.

[16] V. Hauser, Hydra, https://github.com/vanhauser-thc/thc-hydra/, 2022.

[17] D. Song, arpspoof - intercept packets on a switched lan, https://manpages.ubuntu.com/manpages/bionic/man8/arpspoof.8.html, 2022.

[18] G. Jinhua, X. Kejian, Arp spoofing detection algorithm using icmp protocol, in: 2013 International Conference on Computer Communication and Informatics, IEEE, 2013, pp. 1–6.

[19] jacksonliam, Servidor web mjpg-streamer, https://github.com/jacksonliam/, 2021.

[20] A. Miguel, Ataque man in the middle al unitree a1, https://bit.ly/3JGCGDl, 2023.

[21] INCIBE, Rubber ducky, ¿una simple memoria usb?, https://www.incibe.es/empresas/blog/rubber-ducky-simple-memoria-usb, 2023.

[22] O. Angelopoulou, S. Pourmoafi, A. Jones, G. Sharma, Killing your device via your usb port, in: Proceedings of the Thirteenth International Symposium on Human Aspects of Information Security & Assurance (HAISA 2019), The Centre for Security, Communications and Network Research (CSCAN), 2019, pp. 61–72.