International Journal of Mathematics and Computer Research
ISSN: 2320-7167
Volume 09 Issue 05 May 2021, Page no. – 2277-2286
Index Copernicus ICV: 57.55, Impact Factor: 7.184
DOI: 10.47191/ijmcr/v9i5.01

Analysis of IoT security issues used in Higher Education Institutions

Alexei Arina[1], **Alexei Anatolie**[2]
[1,2]Technical University of Moldova, bd. Stefan cel Mare 168, Chisinau, Republic of Moldova

ARTICLE INFO

Published Online:
05 May 2021

Corresponding Author:
Alexei Arina

ABSTRACT

The holistic analysis conducted in this article aimed to identify vulnerabilities, cyber-attacks and ways to mitigate them in IoT systems, which are increasingly implemented in Higher Education Institutions (HEIs). Industry 4.0 is gaining ground in academia, which governs with a large volume of sensitive data, and the growing use of IoT systems brings a number of undeniable benefits and growing challenges. The creation of smart university campuses, of innovative laboratories that allow the simulation of technological processes but also of smart classes that use a mix of technologies for the education of professionals in Industry 4.0, represents the future of education. That is why the security analysis of IoT systems is very important to ensure the confidentiality, integrity and availability of academic data.

KEYWORDS: HEIs, security, attack, vulnerability, IoT, mitigation

I. INTRODUCTION

Industry 4.0 tends to revolutionize the world and change all technological and manufacturing processes in the next few years, by digitizing organizations and automating business processes. The education industry is no exception, and changing the student-centred study process and at the same time sustainably developing useful skills for the future ha[1]s become an absolute necessity. The impact of the remote study imposed by the pandemic situation in 2020, reflected the need to virtualize classes and study resources, so that physical presence is no longer necessary.

The concepts of Industry 4.0 used in the higher-level educational organizations in order to evolve towards University 4.0 to provide better responses to the adaptation of learning and the differentiation of the pedagogical path of each learner [1]. Education 4.0 is a development based on Industry 4.0 applied concepts and digitalisation of higher education institutions and of teaching and learning practices [2].

In general, the industry 4.0 has included several technologies as presented in figure 1 [2].

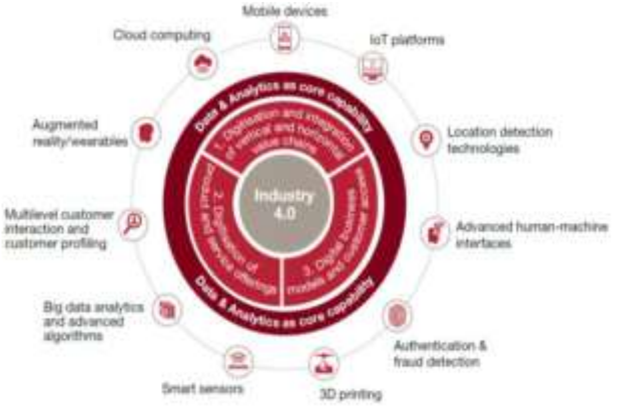
Mobile devices
Cloud computing
IoT platforms
Augmented reality/wearables
Location detection technologies
Multilevel customer interaction and customer profiling
Advanced human-machine interfaces
Big data analytics and advanced algorithms
Authentication & fraud detection
Smart sensors
3D printing
Data & Analytics as core capability
1. Digitisation and integration of vertical and horizontal value chains
2. Digitisation of product and service offerings
3. Digital business models and customer access
Industry 4.0
Data & Analytics as core capability

Fig.1 Industry 4.0 Framework

Industry 4.0 supported by innovative technologies such as Internet of Things, Cloud technology, Augmented and Virtual Reality will also play an important role in manufacturing education, supporting advanced life-long training of the skilled workforce [3].

Some industry 4.0 technologies will be implemented in the future, while technologies such as IoT are already widely implemented in academia. These include all devices connected to the Internet capable of exchanging data and communicating, online learning platforms (LMS) but also facilities for storing and managing data in the Cloud.

The integration of new systems and their increased hypothetical potential third-party access mean that a whole new range of security issues arise in this context [4]. It is important to ensure the security of the university infrastructure that manages with a large volume of sensitive data, and implementing IoT systems for managing it. HEIs are targeted by cyber-attacks because of the information they hold. Information that is of interest for attackers are:

- Intellectual property, in particular institutions that have conducted studies for the development of a vaccine against Covid-19 or various studies in this field. As with many institutions in the UK, which, according to a study by VMWare, who did research to explore the extent of cyber-attacks and the implementation of the IT security standard within HEIs in UK, at least 25% of universities have suffered intellectual property theft [5].
- Personal data of students, including dissertation materials, but also exam results, according to the same study [5], 43% of institutions experienced.
- Research data also represents a major vulnerability, about 28% of institutions have such experience.

Gartner's survey showed that security is the most significant data governance challenge for those organizations planning and implementing IoT solutions [6]. Data governance is a priority for all industries, but the value of academic data is special. Using IoT systems increase security risks. IoT solutions are collecting, analysing and storing huge amount of very valuable data that presents risk for HEIs. At the device level, it is necessary to identify those solutions that will allow the secure use of IoT, and the vulnerabilities and attacks to which IP networks are susceptible are also valid for communications in an IoT system. IoT databases are quite vulnerable to attacks, as are data-driven web applications.

The paper is organized in inter-related sections, in the first section we will discuss about IoT integration in HEIs, the second section contains analyse of the main vulnerabilities and attacks at the different layers of IoT system, at the end the mitigation methods will be discussed.

II. MATERIAL AND METHODS

A. IoT integrations in HEI's

The Internet of Things (IoT) is a global physical network which connects devices, objects and things seamlessly to the Internet infrastructure to communicate or interact with the internal and the external environment, for the purpose of exchanging information [7].

This journal [8] categorized IoT as can be applied to different sectors of academia, which were;

-IoT-based Smart classroom; this involved the use of IoT devices and technology for lecturing and learning processes in academic organizations all over the world which provides new innovative approaches to education and classroom management. Examples of IoT devices found in a classroom to further education include; Interactive Whiteboards, Tablets and Mobile devices, Student ID Cards, 3-D Printers, Wireless door locks, Temperature Sensors, Security Cameras, Electric Lighting, Smart HVAC systems, Attendance Tracking Systems, Room Temperature Sensors, etc.

-IoT-based Smart lab IoT- is implemented by embedding the electrical appliances such as lights, air-conditioners, fans and projectors in the laboratory with sensors and network connectivity. This software enabled physical objects collect and exchange the real time data [8]. Along with the status and energy consumption of individual devices, temperature & humidity status of the laboratory can also be monitored using sensors and viewed in dashboard and mobile application, those reducing their energy consumption.

-IoT-based Smart Campus includes: Smart Street Light System; Smart Parking; Smart Automation; Smart Gardening; Smart Air Quality, Noise Monitoring and Weather Monitoring System; Smart Library; Smart Canteen; Smart Office [9]. Some of the applications described above can be managed through various remote applications. But surely comfort and accessibility will increase greatly and will save HEI's money and resources.

B. Security issues of IoT systems

IoT security requirements can be grouped into the following categories: system-wide, device, communication and Application. Table 1 will reflect the requirements for each category.

Table 1. IoT security requirements

Category	Requirement	Argument
System-wide	Ensure data privacy	Keeping all data private should be a general requirement for confidentiality.
	Minimize attack surface	All potential entry points into the network should be secured, to minimize the risks to gain access in the system by a hacker.
	Log critical events	Logging suspicious activities will allow the administrator to monitor and detect illegal activities.
	Provide at least minimal security operations support	Training security staff is important. They must be able at least to monitor for security incidents.
	Secure boot and system	Devices should have measures to ensure that operating systems

Device	integrity	and software are not tampered with by hackers or malware
	Hardened and secure system	Unnecessary network services should be deactivated, to reduce using them like pathway to access an IoT
	Secure firmware and operated system updates	Secure mechanism for updating these devices over the network must be deployed. It is a critical requirement that device firmware and operating systems can be updated when vulnerabilities are discovered.
Communication	Secure communications	Systems must use techniques to verify that data that is received comes from authentic sources.
Application	No default or weak credentials	Default credentials must be changed prior to putting the device into service. Passwords should conform to security policies for length and composition.
	Secure web interfaces	Login facilities could be vulnerable to various types of cyberattacks. Credentials in use between IoT devices and web applications should be protected from attack.

In 2008, the European Telecommunications Standards Institute (ETSI) creates a framework for understanding the placement of various standards and protocols in an IoT system. This framework contains three domains or layers: Application, Network and M2M device&gateway domain. Figure 2 presents a pictorial view of ETSI architectural reference model [10].

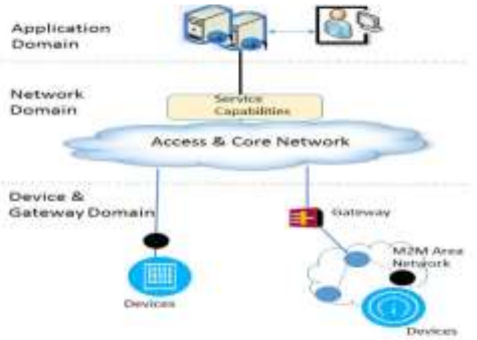

Fig.2 IoT reference model

A simplified version proposed by Cisco is also based on the ETSI model, which facilitates a deeper understanding of the processes that take place [11].

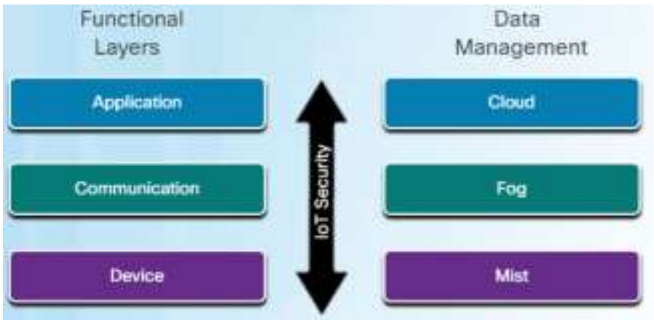

Fig.3 IoT simplified reference model

Regardless of functional connectivity and data management aspects of the IoT system, security must permeate throughout, as shown by the arrow in the figure 3.

1) Device layer

- *Vulnerabilities*, OWASP (Open Web Application Security Project) has compiled a list of vulnerabilities that should be addressed for each attack surface within the IoT system. The vulnerabilities described by the OWASP Attack Surfaces that apply to the devices layer of the IoT Protocol Suite are further defined in Table 2.

Table 2. Device layer vulnerabilities

Hardware sensors	**Device memory**	**Device Physical Interfaces**	**Device Firmware**	**Firmware Update Mechanism**
Environment manipulation Tampering Damage	Default username and password Sensitive data Plaintext usernames Plaintext passwords Encryption keys	Removal of storage media Reset to insecure state Device ID/Serial number exposure Serial interface connections User and Administrative access Privilege escalation	Backdoor accounts Hardcoded credentials Encryption keys Firmware version display Firmware version last update date Vulnerable services Security related function API exposure	Update sent without encryption Update not signed Update location writable Update verification Update authentication Malicious update Missing update mechanism No manual update mechanism

The most common IoT devices used in HEIs that have the vulnerabilities described above are: cameras, NAS devices, networking devices, medical devices, printers.

- Attacks that exploit the vulnerabilities identified above can be classified as follows:

- Physical attacks or hardware attacks [12];
- Firmware attacks.

Table 3 reflects attacks on device layer of the IoT architectural reference model.

Table 3. Attacks on device layer

Physical attacks	**Firmware attacks**
Node tampering	Distributed Denial of Service (DDoS)
RF Interference	
Node Jamming	Backdoor Installation
Malicious Node Injection	
Physical Damage	Buffer Overflow
Social Engineering	
Sleep deprivation Attack	
Malicious Code Injection on the Node	

a. Physical attacks

These attacks affect hardware components of the IoT system, the attacker should be physically close or into the IoT system.

- *Node tampering* the attacker can cause damage to a sensor node, by physically replacing the entire node or part of its hardware or even electronically interrogating the nodes to gain access and alter sensitive information, such as shared cryptographic keys (if any) or routing tables, or impact the operation of higher communication layers [13].

- *RF Interference on RFIDs*, a Denial of Service attack can be implemented on any RFID tag by creating and sending noise signals over the Radio Frequency signals which are used by the RFIDs for communication [14]. The noise signals will interfere with the RFID signals hindering communication [12].

- *Node Jamming in WSNs* is similar to the Radio Frequency Interference physical attack explained earlier for the RFIDs with the difference that this attack is based on the WSNs. The attacker can interfere with the radio frequencies of the wireless sensor nodes, jamming the signals and denying communication to the nodes. If the attacker manages to jam key sensor nodes, he can successfully deny service of the IoT [15].

- *Malicious Node Injection* otherwise called Man in The Middle, it involves obtaining unauthorized access and infiltrating a node into the IoT system to capture data.

- *Physical Damage*, involves the intentional failure of the IoT device, based on two reasons: the interruption of the availability of communications with the device or its replacement with the device controlled by the attacker

- *Social Engineering* is one of the most used attacks, because it is quite easy to perform, it is based on human nature, the attacker obtains information from the users of the IoT system. Even after all studies and information campaigns in this regard, social engineering remains at the top of the attacks that take place in information systems.

- *Sleep Deprivation Attack*, most sensor nodes in the IoT system are powered by replaceable batteries and are programmed to follow sleep routines to extend their battery life. This attack, keeps the nodes awake which will result in a more power consumption, and will cause the nodes to shut down [12].

- *Malicious Code Injection on the Node* involves installing malicious code on the IoT device to produce various effects, such as: encrypting storage units, deleting system files, altering information stored on the device, etc. this can be done locally by using an external storage unit, which contains the code, connected to the device.

b. Firmware attacks

The firmware attacks are carried out exploiting the vulnerabilities exposed in table 2. These attacks affect the availability of the IoT system but also its privacy. Making it impossible to connect authorized users at the right time.

- *Distributed Denial of Service (DDoS) attack* might be accomplished by bombarding a server with huge number of requests to consume all available system resources, by passing the server malformed input data that can crash a process, by infiltrating a virus, or by destroying or disabling a sensor in a system, not allowing it to operate normally [16]. Industry 4.0 relies on a great number of interconnected systems and processes; the DoS attacks are a very important threat in such environments.

- *Backdoor Installation* usually occurs after the attacker gains remote access to the IoT device. On a Linux-based operating system, the attacker could run the netcat command in the background and execute malicious commands on this system remotely from anywhere in the world. In addition, network diagnostic and testing tools are sometimes left behind in the firmware by the IoT device manufacturer. These tools can make the devices more exploitable if unauthorized entry occurs [11].

- *Buffer Overflow* attack can cause corrupt data, a denial of service or cloud allow malicious code to run on the target

IoT system. Can occur whit vulnerable software when the programmer does not account for the size of the input that a user might enter.

2) Communication layer

- *Vulnerabilities*, communications define the nature of IoT devices. IoT is made up of sensors, actuators and other small things that are connected to applications through communication channels. It is the communications that underlie the operation and utility of these systems. The communication layer allows IoT devices to communicate device to device or device to application. Data in transit can be captured, modified and deleted. That's why network-level security is essential for IoT devices. OWASP [17] defines the most common vulnerabilities on communication layer, it’s shown in the table 4.

Table 4. Communication Layer Vulnerabilities

Device Network Services	Network Traffic
Information disclosure	LAN traffic
Injection	LAN to internet traffic
Denial of Service (DoS)	Short range
Unencrypted Services	Non-standard protocols
Poorly implemented encryption	Wireless (Wi-Fi, Z-wave, XBee, Zigbee, Bluetooth, LoRA)
Test/Development Services	Protocol manipulation (protocol fuzzing)
Vulnerable UDP Services	
Replay attack	
Lack of payload verification	
Lack of message integrity check	

It is proposed in this paper to analyse IoT attacks based on IP protocols and the TCP / IP suite, for the communication layer. Many devices do not fully support the stack of TCP / IP transport protocols, so IP services are often enabled for sending data from IoT devices to the applications that manage them. IoT uses IP networks to transmit data, which is why the security of IP networks is very important and absolutely necessary to ensure.

Vulnerable in this regard are:

- The sensor networks
- The IoT gateway
- The enterprise IT network
- The uplink to the internet

- *Attacks*, at the communication layer, cyber-attacks can be classified into attacks targeting IP networks and attacks on the suite of TCP / IP transport protocols. Those mentioned are reflected in the table 5.

Table 5. Attacks on communication layer

IP attacks	TCP/IP attacks
Denial of Service/Distributed Denial of Service	TCP SYN Flood
ICMP flood	TCP reset
Address Spoofing	TCP session hijacking
Man in the Middle	UDP flood
Session Hijacking	

a. IP attacks

- *Denial of Service/Distributed Denial of Service*, IP networks are affected by two major types of DoS attacks that occur: the transmission of maliciously formatted packets and the transmission of an overwhelming amount of traffic. DDoS attacks are similar to DoS attacks, except that they use multiple devices simultaneously for attack. Compromised IoT devices are often used as attack devices in a DDoS attack. Threat actors have been able to exploit password vulnerabilities to copy malicious software onto thousands of internet-connected devices. These devices were then used to attack websites. The sheer number of IoT devices, and the very serious security vulnerabilities in many of them, make IoT devices a very attractive target for botnet DDoS attacks.

- *ICMP flood*, the ICMP protocol is used to determine if a device can be accessed from outside the network but also to check for network errors. An echo request is used to check connectivity to the target device, operating system, and system firewall status. Attackers use ICMP to scan devices and gather information. ICMP is most often used to initiate DoS attacks, by transmitting an overwhelming number of ICMP requests, thus slowing down or even interrupting network services. Elements of IoT systems that communicate via IP address are susceptible to such attacks.

- *Address Spoofing* attacks include both IP address spoofing and MAC address spoofing. These attacks transmit packets over the network that contain false source addresses. The falsification of the MAC address is possible if the attacker is in the same network.
- *Man in the Middle*, IoT devices are prone to man-in-the-middle (MitM) attacks [18]. MiTM attacks are initiated to monitor, capture and control unauthorized communication between devices. Connecting an unauthorized IoT device to an IoT system can result in data theft or tampering.

A possible attack scenario would be in an instance where IoT device is communicating with the cloud for execution instructions, administrative decision making, or firmware updates [19]. An adversary could attempt to redirect network traffic with an attack conducted at the network level, to include Address Resolution Protocol (ARP) cache poisoning or Domain Name System (DNS) modification attacks [20].

- *Session Hijacking*, threat actors gain access to the physical network, and then use an MITM attack to sniff a valid token for access to a web server.

b. TCP/IP attacks

- *TCP SYN Flood*, attackers continually sends TCP SYN session request packets with a randomly spoofed source IP address to an intended target. The target device replies with a TCP SYN-ACK packet to the spoofed IP address and waits for a TCP ACK packet. Those responses never arrive. Eventually the target host is overwhelmed with half-open TCP connections and denies legitimate TCP traffic.
- *TCP reset* (RST) packet is used by a TCP sender to indicate that it will neither accept nor receive more data. Out-of-path network management devices may generate and inject TCP Reset packets in order to terminate undesired connections [21].
- *TCP session hijacking* is meant to intercept the already established TCP sessions between any two communicating parties and then pretending to be one of them, finally redirecting the TCP traffic to it by injecting spoofed IP packets so that your commands are processed on behalf of the authenticated host of the session. It desynchronizes the session between the actual communicating parties and by intruding itself in between [22].
- *UDP flood*, the attacker sends multiple UDP datagrams of different sizes at a time. This causes denial of service to the system and its resources [23].

3) Application layer

- *Vulnerabilities*, IoT devices have many different types of software such as firmware, operating systems, and applications. Any connected device is a vulnerable device. The applications that are installed can have many different types of vulnerabilities. IoT systems can be divided at the application level into:
  - Local IoT applications
  - IoT web and cloud applications.

OWASP [17] made the following classification of vulnerabilities related to the application layer, reflected in Table 6.

Table 6. Application Layer Vulnerabilities

Local IoT applications	**IoT web and cloud applications**
Username enumeration	Injection
Weak passwords	XML external entities (XXE)
Account lockout	Sensitive data exposure
Lack of multi-factor authentication	Broken access control
Insecure 3rd party components	Broken authentication
Insecure communication	
Insecure data storage	
Insecure authentication	
Improper platform usage	
Insufficient cryptography	

- *Attacks*, ensuring security at the application layer is very challenging [24]. Based on the vulnerability classifications performed above, the attacks that can take place at the application layer can be classified primarily into: attacks on local applications and attacks at the web/cloud application.

Cyber-attacks that can take place on local applications can be classified as: local attacks that take place within the network and remote attacks. The results identified from the research are reflected in the table below.

Table 7. Attacks on Application layer

Local IoT applications		**IoT web and cloud applications**
Local attacks	**Remote attacks**	
Firmware Replacement Cloning Denial of Service (DoS) Extraction of Security Parameters Malicious codes	Man-In-the-Middle Eavesdropping SQL Injection (SQLi) Routing Attack Malicious codes	Cross-site Scripting SQL Injection Broken Authentication Malicious codes

a. Local IoT applications

- *Firmware Replacement*, system updates and patch installation are usually done centrally from within the network, but if communications have been compromised. Attackers can modify the contents of the package containing the updates by installing their own programs through which they can later gain access.
- *Cloning* is possible by replacing an authorized network device with another device running the same operating system and applications.
- *Denial of Service (DoS)*, an attacker can execute DoS or distributed denial of service DDoS attacks on the affected IoT network through the application layer, affecting all users in the network. This kind of attack can also block the legitimate users from the application layer giving full application layer access to the attacker; databases and private sensitive data [25].
- *Extraction of Security Parameters*, when a device is not protected properly, the threat actor may be able to extract security parameters from it such as authentication information or security keys [26]
- *Malicious codes*, an adversary can infect the system with malicious software resulting in a variety of outcomes; stealing information, tampering data or even denial of service [27]. Malicious codes can be viruses, worms, spyware and adware, phishing, Trojan Horse.
- *Man-In-the-Middle*, when two users of an IoT system A and B, exchange keys during a challenge-response scenario, so as to establish a secure communication channel, an adversary positions himself between them on the communication line. The adversary then intercepts the signals that A and B send to each other and attempt to interfere by performing a key exchange with A and B separately. The adversary will then be able to decrypt/encrypt any data coming from A and B with the keys that he shares with both of them. Both A and B will think that they are talking with each other [12].
- *Eavesdropping attack* occurs by intercepting data, such as the security keys of a session.
- *SQL Injection (SQLi)* occurs when the attacker exploits vulnerabilities in Structured Query Language (SQL). Subsequently gain access to the database, where unauthorized access or retrieve data from the file system.
- *Routing Attack*, direct attacks that the adversary by spoofing, altering or replaying routing information can complicate the network and create routing loops, allowing or dropping traffic, sending false error messages, shortening or extending source routes or even partitioning the network [28].

b. IoT web and cloud applications

- *Cross-site Scripting (XSS)* specifically affect the scripts of a web application's code. JavaScript scripts are most affected. The attacker inserts in the legitimate code scripts that fulfil certain functions, such as stealing cookies, redirecting traffic to rogue web pages. These functions are performed whenever you hover over a link or click.
- *SQL Injection* allows the attacker to gain access to university databases. This way attackers can modify, insert false data or delete records. As with XSS, the attacker inserts unauthorized code that performs certain programmed functions. There are common attacks because the application does not sanitize the untrusted data that is entered in the web page fields.
- *Broken Authentication* occur through brute force attacks, dictionary attacks or when using credentials by default. The attacker hijacks a session by assuming the identity of the authorized user.

III. RESULTS AND DISCUSSION

A. Threat mitigation on device layer

To mitigate the risks related to the device layer, a defining role is played by controlling access to devices and the services provided by them, but also encrypting stored data, such as passwords, and encrypting the communication channels through which data is transmitted in an IoT system. Next will be given some recommendations.

- *Access Control Model*

In order to mitigate the security risks at device level in IoT systems, it is necessary first of all to consider the implementation of an optimal access control model. Most common models are: Mandatory access control (MAC), Discretionary access control (DAC), Non-Discretionary access control (RBAC) and Attribute-based access control (ABAC). Because of its simplicity, DAC remains the most used method for real-life IoT deployments, and notably in the case where an IoT Object, identified by its physical address (i.e., Media Access Control Address) or by credentials stored within the object [29]. The owner of an object sets access control policies on an object. The AC decision is based on the access rights of subjects, characterized by an identifier, e.g., IP or physical address. These rights are typically represented by an access matrix or Access Control Lists (ACLs) assigned to each object [29].

- *Authentication and authorization*

Authorizing allowed actions is a key factor that limits access to IoT devices, including access from third-party applications. The standardized protocol OAuth 2.0 Authorization Framework, presupposes the existence of a centralized server that authorizes the access to resources. The working algorithm is based on the authorization request and on its validation in the system.

- *Identity Resource Management (IRM)*

Is a new approach, it can help to manage a large number of identities and relationship between them. Is much more scalable than Identity and Access Management (IAM) platforms that cannot perform the link between many devices.

- *Data Security*

It is very challenges because many old IoT device, cannot support any type of encryption. Another problem is IoT devices often use wireless connections which makes easier for attackers to intercept data transmissions. Anyway, exists several IoT wireless standards that support some level of security: Zigbee, LoRa, LTE-M, White-Fi.

B. Threat mitigation on communication layer

To mitigate the risk of the level of communications it is necessary to check the following: the use of encryption and authentication, the integrity of session key management, how to maintain the ACL, the version of protocols used, whether the firewall segregated traffic in different areas. After verification, it is recommended to use the STRIDE model to identify the threats and the DREAD model to score the risk of each threat. OWASP divides the risk mitigation measures at the communication layer, as follows: unsecured network services and problems with transported data.

- *Insecure network services*

A major challenge is logical ports, to mitigate attacks the number of open ports should be limited and should not be accessible from the Internet.

- *Lack of transport encryption*

It is necessary to encrypt not only the communications between the system components but also the communications between the system / device and the internet. Proper configuration and implementation of TLS and SSL protocols is also very important. Using a firewall to limit incoming/outcoming data.

B. Threat mitigation on application layer

Secure development and deployment of IoT systems is the most effective way to mitigate risk. This process should be a requirement for the deployment of all devices, communication channels, and applications. To mitigate risks at the application layer is very challenging [24]. Vulnerabilities that occur with the use of third-party software that cannot be controlled by the system administrator, such as disabling updates by the manufacturer to reduce costs, or vulnerabilities that allow exploitation by malicious code, make this layer the security of IoT systems to be one of the most difficult to ensure. In addition to vulnerabilities related to third-party applications, there are many vulnerabilities related to messaging protocols, such as: MQTT or CoAP. Finally, vulnerabilities in weak or inefficient passwords that are susceptible to multiple attacks.

- *Securing messaging protocols*

It is necessary to use Client ID authentication, usernames and passwords or even the use of client certificates, but when there are many clients this becomes inconvenient because the certificates must be configured and managed on each device. Also, as in the case of securing the HTTP protocol, in this case too, the configuration with the TLS protocol that will encrypt the connection between the nodes would be a good solution. It's just that this solution is limited by devices that have little processing power or memory.

- *Strengthen and protect passwords*

Using a pass phrase instead of a password might be a good solution. Or use a password that is hard to guess, and in this sense, it is quite useful to use a password manager, which will allow you to create a strong password. For more sensitive systems it is advisable to use multi-factor authentication.

- *Harden Administrative Interfaces*

Interfaces exposed to the internet are constantly under attack by threat actors. Limiting access can be an effective measure. Using encrypted transport protocol and validating certificates are good practice. Strong credentials are in place before deployment because many users will not change the configuration.

IV. CONCLUSION

The security analysis of IoT systems performed in this article showed several security issues, various attacks and how challenging the data security process is. The governance of data managed by IoT systems is susceptible to several attacks aimed at exploiting vulnerabilities at the IoT device level, violating the security of communication protocols and of course exploiting web interfaces.

Restricting access at each level: device, communications and application; remains the biggest challenge. Imposing strict access rules solves several problems, so:

- access to the IoT device and unauthorized authentication, obtaining administrative access rights, escalating privileges or creating backdoor accounts on systems.
- access to internet traffic and exploration of transport protocol vulnerabilities.

- unauthorized access to web applications and data theft, account blocking, failure of multi-factor authentication and improper use of web platforms.

The attacks that IoT systems are susceptible, common to all layers are: DoS / DDoS, Man attacks in the Middle and of course malicious code infiltration.

Mitigation process and strengthening IoT systems is complex and still need to be researched and explored.
However, the number of IoT devices used in HEIs is constantly growing, due to mobility, scalability and convenience in use.

REFERENCES

1. Mamadou Gueye and Ernesto Exposito, "University 4.0: The Industry4.0 paradigm applied to Education," Oct. 2020.
2. Irina Neaga, "APPLYING INDUSTRY 4.0 AND EDUCATION 4.0 TO ENGINEERING EDUCATION," in Proceedings 2019 Canadian Engineering Education Association (CEEA-ACEG19) Conference CEEA19, Jun. 2019, pp. 1–6.
3. D. Mourtzis, E. Vlachou, G. Dimitrakopoulos, and V. Zogopoulos, "Cyber- Physical Systems and Education 4.0 –The Teaching Factory 4.0 Concept," Procedia Manufacturing, vol. 23, 2018, doi: 10.1016/j.promfg.2018.04.005.
4. T. Pereira, L. Barreto, and A. Amaral, "Network and information security challenges within Industry 4.0 paradigm," Procedia Manufacturing, vol. 13, 2017, doi: 10.1016/j.promfg.2017.09.047.
5. Tim Hearn, "VMWare. VMware Cyber Security Report." https://www.qassociates.co.uk/wp-content/uploads/2016/06/36300-VMware-Cyber-Security-Report.pdf (accessed Feb. 15, 2021).
6. Christy Pettey, "How IoT Impacts Data and Analytics," Gartner, 2018. https://www.gartner.com/smarterwithgartner/how-iot-impacts-data-and-analytics/ (accessed Feb. 23, 2021).
7. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions," Jul. 2012, Accessed: Feb. 28, 2021. [Online]. Available: https://arxiv.org/abs/1207.0203.
8. M. Poongothai, A. L., and R. Priyadharshini, "Implementation of IoT based Smart Laboratory," International Journal of Computer Applications, vol. 182, no. 15, Sep. 2018, doi: 10.5120/ijca2018917853.
9. Madhura Rao, Neetha, Rao Swathi, Sneha M, Shannon Kotian, and Nagaraja Rao, "An IoT Based Smart Campus System ," International Journal of Scientific & Engineering Research, vol. 9, no. 4, Apr. 2018.
10. "Machine-to-Machine Communications (M2M); Functional Architecture," ETSI TS 102 690 v2.1.1, 2013.
11. David Hanes and Gonzalo Salgueiro, Networking Technologies, Protocols, And Use Cases For The Internet Of Things, 1st ed., vol. 2017.
12. I. Andrea, C. Chrysostomou, and G. Hadjichristofi, "Internet of Things: Security vulnerabilities and challenges," Jul. 2015, doi: 10.1109/ISCC.2015.7405513.
13. A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," Communications of the ACM, vol. 47, no. 6, Jun. 2004, doi: 10.1145/990680.990707.
14. Li, "Study on security architecture in the Internet of Things," in In Measurement, Information and Control (MIC), 2012 International Conference, 2012, pp. 374–377.
15. A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou, "A survey on jamming attacks and countermeasures in WSNs," IEEE Communications Surveys & Tutorials, vol. 11, no. 4, 2009, doi: 10.1109/SURV.2009.090404.
16. "Security On The Industrial Internet of Things: The essentials of a secure network," Deutsche Telekom AG Corporate Communications, 2016.
17. "Internet of Things," OWASP. https://owasp.org/www-project- internet-of-things/ (accessed Feb. 20, 2021).
18. Navas RE, le Bouder H, Cuppens N, Cuppens F, and Papadopoulos GZ, "Do not trust your neighbors! a small IoT platform illustrating a man-in-the-middle attack," in International conference on ad-hoc networks and wireless, 2018, pp. 120–125.
19. A., T. Niculcea, P. Ranaweera, and N.-A. Le-Khac, "Security Considerations for Internet of Things: A Survey," SN Computer Science, vol. 1, no. 4, Jul. 2020, doi: 10.1007/s42979-020-00201-3.
20. M. A. Hussain, H. Jin, Z. A. Hussien, Z. A. Abduljabbar, S. H. Abbdal, and A. Ibrahim, "Enc-DNS-HTTP: Utilising DNS Infrastructure to Secure Web Browsing," Security and Communication Networks, vol. 2017, 2017, doi: 10.1155/2017/9479476.
21. N. C. Weaver, "TCP Reset Injection," in Encyclopedia of Cryptography and Security, Boston, MA: Springer US, 2011.
22. Shray Kapoor, "Session Hijacking Exploiting TCP, UDP and HTTP Sessions", 2006.
23. A. Bijalwan, M. Wazid, E. S. Pilli, and R. C. Joshi, "Forensics of Random-UDP Flooding Attacks," Journal of Networks, vol. 10, no. 5, May 2015, doi: 10.4304/jnw.10.5.287-293.

24. G. Nebbione and M. C. Calzarossa, “Security of IoT Application Layer Protocols: Challenges and Findings,” Future Internet, vol. 12, no. 3, Mar. 2020, doi: 10.3390/fi12030055.
25. C. M. Medaglia and A. Serbanati, “An Overview of Privacy and Security Issues in the Internet of Things,” in The Internet of Things, New York, NY: Springer New York, 2010.
26. CISCO, “IoT security fundamentals,” CISCO. https://static-course-assets.s3.amazonaws.com/IoTFIoTSec10/en/index.html#5.1.1.2 (accessed Feb. 14, 2021).
27. T. Heer, O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. S. Kumar, and K. Wehrle, “Security Challenges in the IP-based Internet of Things,” Wireless Personal Communications, vol. 61, no. 3, Dec. 2011, doi: 10.1007/s11277-011-0385-5.
28. D. Wu and G. Hu, “Research and improve on secure routing protocols in wireless sensor networks.,” in Circuits and Systems for Communications, 2008. ICCSC 2008. 4th IEEE International Conference, 2008, pp. 853–856.
29. E. Bertin, D. Hussein, C. Sengul, and V. Frey, “Access control in the Internet of Things: a survey of existing approaches and open research questions,” Annals of Telecommunications, vol. 74, no. 7–8, Aug. 2019, doi: 10.1007/s12243-019-00709-7.