Social Robots Privacy Enhancement Using Colored Petri Net (CPN) for Behavior Modeling: A Case Study of Asus Zenbo Robot

Benjamin Yankson[1], Farkhund Iqbal[2], and Fadya AlMaeeni[2]
[1]HackIoT Lab - College of Emergency Preparedness, Homeland Security and Cybersecurity University at Albany, Albany, USA
[2]College of Technological Innovation, Zayed University, Abu Dhabi, UAE

Byankson@albany.edu

Abstract: The interactions between a social robot and the user consist of continuous communication and behavior involving different data types that can be subject to cybersecurity attacks and prone to user privacy concerns. In this paper, we use Colored Petri Net (CPN) to develop two graphical models that illustrate different patterned behavior of a robot during such interaction. Using CPN, we model and analyze complicated robot system interactions considering synchronization and concurrency of events that can be subject to privacy violations. We focus on two specific scenarios involving user registration and medication reminder to provide an efficient illustration of the objects and events collaborated to carry out the intended tasks of the robot pertaining to privacy issues. The results show that CPN modeling simulation can capture and represent the robot's behavior, provide a better understanding of the task execution, and highlight users' privacy gaps requiring immediate controls.

Keywords: Privacy, Coloured Petri Nets, Modelling, Security, Robot

1. Introduction

For the last couple of years, one of the trending technologies has been social robots, which are widely used for numerous purposes and carry meaningful importance in social environments. Social robots are intelligent robots that interact socially with humans in their private environments, such as households or workplaces, and can mimic some human behavior. According to Taipale et al. (Taipale, 2020)], "Social robots are being designed to deal with human care, health, domestic tasks, entertainment, and various other immaterial and material tasks." These robots perform multiple activities based on their designs, making people feel comfortable and satisfied to interact with them. According to Tobe (Tobe, 2017), it is remarked that the robotics business is rising significantly, and it is predicted by the Boston Consulting Group (BCG) that the sales of robots "will reach $87 billion by 2025." (Tobe, 2017).

Similarly, Tractica predicted the sales of robots "will reach $237 billion by 2022." (Cole, 2020). Social robots are part of the Internet of Things (IoT) devices and comprise hardware and software modules with Internet connectivity, cloud services, and third-party applications for robot operations. Like any emerging computing technology, the amount of data residing in the robot constructions opens it to the same security and privacy concerns as other IoT devices. Moreover, the social robot's basic task is communicating with users visually and verbally; hereby, it continuously sends and receives data packets directly or over the network. The demand for Internet connectivity means that security holes would exist within the conceptual robot design because the fact is that anything connected to the Internet is prone to cyberattacks and subject to privacy violations (Poremba, 2016).

As such, the components of the robot system need to be understood to explore and identify all the possible security and privacy matters that a user may encounter during social robot interaction. Capturing the robot's behavior in a graphical representation to express the data flow between objects and describing the events would assist in identifying the vulnerable areas in the robot design and verifying the reliability of robot operations. To precisely capture the robot's behavior, formal models such as a Finite State Machine, automata, calculus, predicate logic, and Unified Modeling Language (UML) can be used. Accordingly, in this paper, we used Colored Petri Net (CPN) modeling mechanism to capture the behavior of ASUS' Zenbo robot during user interaction. CPN modeling is an extended version of the standard Petri Net (PN), which is a mathematical model applied to represent a Discrete Event System (DES), and it can be in two different structures: matrix representation or graphical representation (Da Mota, 2018). This model is widely used in computer science for software design, considering synchronization, sequence, and concurrency for complex system implementation (Reisig, 2013). CPN was introduced due to a significant limitation in the basic PN modeling: different data types cannot be referred to as a single state or object within the system. However, CPN allows users to define various data sets with different attributes. With CPN, the user can assign multiple attributes to one state depending on the requirements of the constructed scenario to assess privacy during user interaction simulation with the robot.

1.1 Security Implications

The logical examination of social robots' security problems demands understanding the robot's conceptual design and structure that integrates different components. Moreover, recognizing the services the social robot providential robot's services would help identify the features that the robot uses in communication and perform its functions, such as vision through gazing (gesture and motion), speech recognition, or screen display.

Many robotic systems are built on the ROS, which is used as middleware to provide several levels of software abstraction to hardware and robotics resources (i.e., sensor and actuators) in addition to reusing open-source project libraries (Koubaa, 2017). We surveyed different social robots to identify the platform used for robot development. It was observed that all of them are based on open-source platforms, including ROS and Linux. The vulnerabilities of Android and Linux-based operating systems are an ongoing research topic in the cyber security field because of their open-source feature and allowing for installing third-party applications on the system (Sankar, 2017).

Current research work and investigation have also identified a plethora of known vulnerabilities, including but not limited to robot interface authentication/authorization issues and bypasses and insecure transport of data (Zorz, 2017). Other avenues of vulnerabilities include a lack of firmware update mechanisms, improper and undocumented methods, hard-coded passwords, unencrypted storage, and easily disabled human safety protections (Zorz, 2017). Other than the identified vulnerabilities, some of these robots have exposed connectivity ports such as USB and Ethernet, with the possible remote access that a remote attacker can exploit to interfere with the robots' safety features. All such vulnerabilities can, in most cases, be exploited to violate users' privacy, hijack the robots, and in the worse cases, result in user safety in a case where the robot safety feature is compromised.

1.2 Privacy Implications

There are privacy issues associated with the use of social robots. Based on the services offered by social robots, users' sensitive data are collected and stored in their applications' databases or cloud storage. Since the user needs to be identified by the robot using different methods such as creating an account, face recognition, and voice recognition, personal details like name, email, and images are stored within the robot system. Moreover, the user interacts with different applications provided by the robot-like the reminder, calls, and information browsing. Therefore, users should consider their data protection critical while using robots for various activities. Such protection includes understanding all-important privacy issues such as robot manufacture used, robust cryptography methods to encrypt the local or cloud storage data, the ability of the user to identify and turn off location tracking and context data collection, the understanding of data ownership after transmission to the cloud, the availability of privacy policy and privacy compliance.

The contribution of this work is three-fold. First, We contribute to the literature by being one of the first to use Colored Petri Net (CPN) to model robot and user interaction to identify multiple data transition points that can be subject to a possible user privacy breach. In this work, CPN is used to develop two graphical models that illustrate different patterned behavior of a robot during such interaction. The presented model can be used to analyze complicated robot system interactions considering synchronization and concurrency of events that can be subject to privacy violations. The model capture and represent the robot's behavior, provide a better understanding of the task execution, and highlights users' privacy gaps requiring immediate controls. Second, this work highlights the privacy and security implications of humanoid robots. This work provides a better understanding of the task execution and the sequence of events. Third, CPN modeling is used to capture and represent the behavior of the Zenbo robot. We focus on two specific scenarios that demonstrate user registration on Zenbo and the capability of Zenbo to remind the user about his medication. The graphical representation of the situations would help simplify Zenbo's user interactions. The rest of the paper is organized as follows: Section 2 presents an overview of previous work. Section 3 details the methodology and implementation of a Zenbo scenario using CPN Tools. Section 4 concludes the paper and highlights future works.

2. Background and Related Work

This section provides an overview of previous work on social robots considering two sections: security and privacy and a Petri Net model.

2.1 Social Robot Security & Privacy

Chung et al. (Chung, 2017) discussed the complex operation of IoT devices, specifically the Amazon Alexa robot with Amazon Echo Wireless speaker, which they studied to identify the fundamental components that can be used as evidence in digital forensics crime cases. These components include cloud resources, user resources (mobile applications and web browsers), network resources, and hardware. Moreover, the researchers presented a systematic methodology called Cloud-based IoT Forensics Toolkit (CIFT), mainly built to obtain the inherent forensics objects from Alexa and analyze them using a Python script to extract forensics artifacts. The analysis includes different uncovered details, including the client files, Wi-Fi settings, Alexa-enabled skills, artifacts related to mobile apps, and a timetable for the results. Finally, Subramanian (2017) highlighted the significant implications of social home robots in several aspects, including security, privacy, and ethics. Additionally, he proposed some key points that can solve issues that several previous researchers raised.

Ray (Ray, 2016) provided an overview of the Internet of Robotic Things (IoRT) system. He described its design structure elements, consisting of different layers referred to as the Open System Interconnection (OSI) model. Besides, Ray examined the existing challenges in the system related to computation, optimization, security, and ethical issues. Do et al. (Do, 2018) developed a new robot called RiSH, Robot-integrated Smart Home, which aims to help older adults at home. The researchers presented the complete architectural design of the robot by describing all the hardware components used and the software components implemented. Besides, they built different services for the robot involving a variety of voice and motion sensors. The proposed RiSH system is believed to help other scientific researchers to conduct experiments on different robotic systems and highlight concerns related to the Information Technology field.

Portugal et al. (Portugal, 2017) inspected several home robots which use the ROS to analyze the most significant security concerns affecting the users, including authentication, authorization, and confidentiality issues. Furthermore, a case study is introduced to implement the possible security measures in each layer involved in a robotics ROS framework, including a wireless network, PHP Graphical User Interface (GUI), cloud server, ROS Multimaster, Operating System, disk data, and hardware level. The proposed protection system was called STOP Research and Development (R&D), mainly inherited from the actual STOP R&D Project.

Lee et al. (Lee, 2013) addressed data privacy associated with humanoid robots by conducting interviews with people to observe their understanding of data collection and protection within the robotic systems. The researchers highlighted the robotic features that pose privacy concerns, such as the various sensors built within the system, the motion capability, the verbal interaction between the user and the robot that reveals a lot of personal data, and sharing of social robots among different users. Fernandes et al. (Fernandes, 2016) also highlighted the risk of privacy issues of social robots since the camera is undoubtedly a part of the robot's systems. Thus, visual records can be obtained, revealing the users' unacceptable conditions. They discussed the privacy concerns specifically for older adults and disabled individuals because they are some of the primary users of social robots engaged within smart homes for their helpful capabilities. To enhance the privacy perspective in social robots and protect users' sensitive data, the authors presented a mechanism involving Convolutional Neural Networks (CNN): a concept used to examine images considering different features. In their work, CNN was used to recognize the private situations of the users where their privacy could be invaded. The model was built on the Advanced Sensing, Computation, and Control (ASCC) home service robot to navigate the social robots away from embarrassing situations based on its recognition using CNN. Moreover, they conducted a survey highlighting that people claimed their privacy and supported the implemented model.

2.2 Petri Net Model for Robots

Erden and Araz (Erden & Araz, 2016) presented a PN model developed to capture the walking behavior of a dog in real-time. It was built using the Artifex graphical modeling and simulation tool. The behavior in the model is represented by different functions, which can be used as transitions in the model. The dog's body parts (brain and four legs) involved in the behavior are referred to as places in the model. Andrews et al. (Andrew, 2016) represented Urban Search and Rescue Robots, where different disaster behavior models are described in the PN model. UML classes were used as a first step to describe each function and describe the functions' relations. Then, the PN model was built in reference to the created classes. Yang et al. (Yang, 2015) introduced a PN model integrated with Fuzzy logic, where the robot navigation functionality is used as a case to represent the PN model. It also explained the algorithm used to represent the PN model for the proposed scenario alongside the graphical representation. Two tools were used to build the Feature Pyramid Network (FPN) model: the HPSim Petri Net simulation tool and the YASPER tool. Chao and Thomaz (Chao & Thomaz , 2016) described implementing the PN

model for controlling real-time turn-taking decisions of the Simon Robot considering the time, called Timed PN. The paper guided the model through a building, mainly considering resources such as objects, spatial regions, the speaking floor, and the robot Degree of Freedom (DoF) represented by tokens and behavior (speech, gaze, gesture, and manipulation). Da Mota et al. (Da Mote, 2018) presented the PN model for a robot navigating behavior using RFID sensors by building four different paths. The model captures different actions: “stopping, moving forward, and turning right and left.” The authors' used matrix and graphical representations of the PN model. Another work that is instrumental in this area is the work of Stephenson (Stephenson, 2004). The author (Stephenson, 2004) proposes an optional method of risk analysis in Cybersecurity and quantification using statistical methods. Using proven formalisms, the authors used CPN for sophisticated modeling, simulation, and analysis of complex information security system behavior. The authors pointed out the advantage of CPNet's graphical representation, which helps construct, modify, and present complex models. Based authors' work, we build on modeling privacy spots in data transition for robot interaction.

3. Methodology - Petri Net Model Implementation

We analyze the two implemented CPN models and express the models in algorithm forms to clearly understand the sequence of events. The classical PN is composed of four fundamental components, where the fourth component is used only with CPN modeling, and they are (De La Mota, 2017):

1. *Places (P)*: denoted by circles, representing the system’s different states/objects/resources carrying out the events.
2. *Transitions (T)*: They are denoted by bars or rectangles, representing the events/interactions.
3. *Arcs (A)*: denoted by arrows, and they are used to represent the connections between places and transitions. It is impossible to draw an arc between the same type of nodes (i.e., between two places or between two transitions).
4. *Tokens* are denoted by dots and used to represent the data values inside each place based on the type of color set assigned to the place. These tokens are used as input data for the transition to be enabled and verify the action carried out. The tokens are added by marking the place.

3.1 Tool Setup & Scenarios

3.1.1 CPN Tools

Since We installed CPN Tools 3.4.0 on a Windows 7 VM running on VMWare Workstation to implement the CPN model (CPN, 2018). Although we use the ASUS Zenbo robot for this work, other researchers can use the methodologies to apply to other humanoid robots.

3.1.2 ASUS Zenbo Scenarios

The Zenbo robot is known to carry out multiple functions concurrent with different behavior interactions with the user. Our target in this work is to focus on a scenario to capture the robot’s interactions while executing a specific task involving a set of objects and events. We present the scenario captured through the CPN model, and for the scenario, we explain the developed model along with the involved events and objects. In addition, focusing on the security and general safety aspects, we highlight areas in the developed model that require end-user attention and awareness when using this social humanoid robot. Our presented scenario is about using the reminders App of the robot to add a medication reminder. The scenario was developed based on actual experiments on the Zenbo robot that involved studying some of the installed applications, and the way data is stored. Therefore, this approach was considered to create realistic CPN models based on factual data rather than assumed scenarios.

3.1.3 User Registration

Our first CPN model describes the process when users register themselves with Zenbo, as illustrated in Figure 1. Figure 1. shows that the process starts with P1, expressed by the STATUS data set to include the “OFF” token, which exemplifies that Zenbo is turned off. We have the declarations mentioned in Table 1. The “ZDATA” color set is composed of all the data types in the model, and the “STATUS” color set describes the different status types that an object can be at a given time. The initial User registration model comprises eleven places and ten transitions showing the execution of events sequentially. In addition, several tokens were defined based on the required input and output data for the states needed to fulfill the robot system interaction with the user. This model comprises eleven places and ten transitions showing the execution of events sequentially

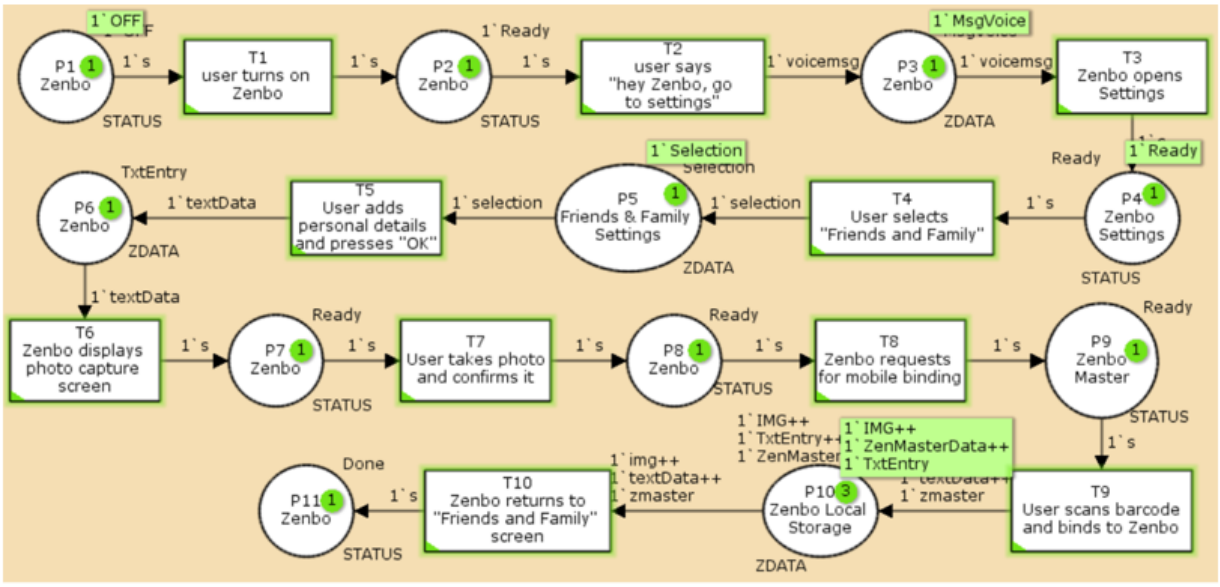

Figure 1. User Initial Registration

In this first model, we have the declarations mentioned in Table 1. The "ZDATA" color set is composed of all the data types in the model, and the "STATUS" color set describes the different status types that an object can be at a given time. We declared one variable referring to a STATUS color set, while five variables refer to the ZDATA color set.

Table 1. CPN Model 1 Declarations

Name	Type	Tokens/Values
ZDATA	colset	IMG, ZenMasterDate, TxtEntry, MsgVoice, MsgText, Selection
STATUS	colset	Done, ON, OFF, Ready
s	var	STATUS
voicemsg	var	ZDATA
selection	var	ZDATA
textData	var	ZDATA
img	var	ZDATA
zmaster	var	ZDATA

3.1.4 Reminder App

The reminder is a built-in app in Zenbo that provides valuable functionality for reminding users of different tasks through two options; a Calendar event or a To-Do-List item. This feature is vital for an older adult because it can remind the user to take the required medications at a specified time. For example, the CPN model illustrated in Figure 2 tells the scenario for adding a medication reminder through a To-Do-List item and then getting reminded by Zenbo at the specified time.

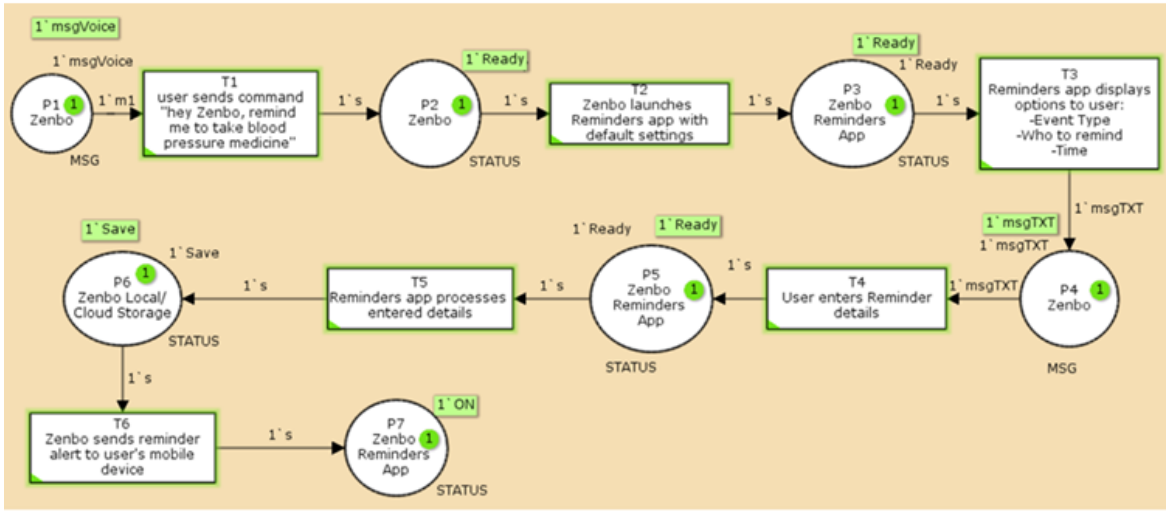

Figure 2. : CPN model of User Interaction with Zenbo Reminder app

3.1.5 Declaration

Table 2 shows all the color sets and variables declared for this model. This model declared three color sets with required tokens and two variables referencing some of these declared color sets.

Table 2: CPN Model 2 Declarations

Name	Type	Tokens/Values
ZDATA	colset	Msg, Voice
STATUS	colset	Save, ON, OFF, Ready
MSG	colset	MsgVoioce, MsgTXT
s	var	STATUS
m1	var	MSG

3.1.6 Model Description

Seven places represent the objects/resources involved in the process's execution of the Zenbo Reminders app used by the user. The significant entities of the structure include Zenbo, Reminders App, and the User. These entities are defined in several places characterized by one token or more having the same data types based on the case running to achieve the required task. However, six transitions capture the events executed sequentially between the places. They indicate the changes occurring between the states and the success of the data flow operation. For example, P1 represents Zenbo as it is ready to receive a voice message from the user. Zenbo gets alerted when the user initiates verbal communication with "Hey Zenbo" and posts these two words. The user continues by requesting the robot remind him to take blood pressure medication, represented by transition T1. There are different ways to create the required reminder on Zenbo; for example, the user may raise his request in one sentence to create the required reminder for a specific time and person, but in our model, we represent the process step-wise. Finally, P2 represents Zenbo again, but now it is ready to launch the Reminders App with the default settings, represented by transition T2, based on the initial voice commands sent by the user in an earlier transition.

Zenbo Reminders app opens and is ready for further action. This state is presented by P3, which triggers transition T3, which displays a list of other options to edit the reminder entry before saving it. Some fields the user needs to attend are the event type (calendar event or to-do-item), whom to remind among the members configured on Zenbo, and the time for the reminder to trigger. Based on our experiment, the to-do-item is a better option to select than calendar events in scenarios for medication reminders because such alerts get triggered on the Zenbo Masters app on the user's mobile device. While the user can configure the rest of the reminder entry options by voice commands, in our model, we present the configuration with a text/selection option since this is easier for the user based on our experiment. In transition T5, the Reminders app processes the reminder entry stored in P6, representing local or cloud storage. When it is time for the reminder to trigger based on the configured time, Zenbo launches the Reminders app that triggers an alert only on the mobile device linked to the robot via Zenbo's Masters' App. It is to be noted that though the originally published videos about Zenbo on the internet highlight that an alert for a to-do-item can be presented on Zenbo's screen, our

experiment did not find this possibility. The Reminders tutorial configured on Zenbo mentions that reminder alerts will be sent on a linked mobile device without mentioning the on-screen alert.

3.1.7 Observation

Table 3 lists the five observations that we have on this process.

Table 3: CPN Model 2 Observations

#	Description	Ref
1	Since it is difficult to set up voice recognition, as mentioned in the earlier model, this will mean that anybody nearby Zenbo can add a reminder to it. However, this may cause a medical issue for the elderly if two reminders were added for the type of medicine that is supposed to be taken once a day.	T1
2	The user cannot upload his medical record on Zenbo, so any entry for medicine reminders can be validated against the record. We recommend ASUS implement this enhancement on Zenbo since the robot can be used for elderly care as one of its stated purposes.	T1
3	It is challenging to have all the entries done through voice commands. So, the user may necessarily use selection instead of issuing voice commands. This makes it difficult for the elderly to set a reminder. ASUS could have made this process much easier by breaking the required entries into separate screens, with each screen displaying when the user can make direct voice commands to select or even choose to use manual entries.	T3
4	Reminder records are getting stored on the robot as well as syncing online. This could be proven by locating the extracted App's records that state the last syncing option and next syncing token. This is not a vulnerability, but it is worth highlighting since some users may be concerned about having their data on the cloud. Additionally, there is no encryption for the reminder data stored locally on the robot.	P6
5	The reminders App does not show the alert on the display screen of Zenbo. The ideal expected situation is that the alert can be displayed with a voice message to take his medicine. However, that does not happen. Instead, the user can get a reminder on the Zenbo Master mobile app, which may make it inconvenient for the elderly because not all of them are familiar with mobile devices, and they may not be as close to mobile devices as often as younger people.	T6

4. Analysis, Result, and Conclusions

The implemented models of Zenbo behavior may apply to any other social robot since any robot would have similar processes to execute its tasks. Using CPN modeling to capture social robot interactions with the user has several advantages from different perspectives. The user can design the complex system of the robot hierarchically, considering the synchronization of the workflow among the objects in detail. The graphical representation is highly comprehensive in that the user can easily assign captions to each model component, making the diagram readable. This feature allows the user to identify the needed resources for each task to be carried out and realize any system failures encountered while executing the intended task. In other words, the user can refer to the CPN model to follow the sequence of events and ensure that the robot is acting normally as expected based on the outcome of the robot. People need to consider the successful operation of the robots if they decide to depend on them for serious tasks.

Furthermore, CPN models may spot insight into the robot system's data privacy and security concerns. For example, the CPN model visualizes the data flow, representing the incoming and outgoing traffic through the system. This raises attention to exploring the expected security vulnerabilities that can exploit the operation of the robot processes. Likewise, the data collection and storage, which usually contain user data, must be ensured with high protection.

Social robots pose security and privacy concerns similar to any other computing smart device. This paper captured user registration, initial interaction with Zenbo, and Zenbo's reminders app for medication notification. The behaviors were represented using the CPN modeling tool, considering the tool's capability to deal with complex systems. Furthermore, the models were constructed to show the workflow of the intended behavior. As a result, CPN representations clearly understood the robot's interaction with the user and its internal systems, showing the process status with the flow of different data types.

Furthermore, the models allowed us to infer some hints about the robotic system's possible security and privacy issues. Finally, different recommendations were provided to overcome the robot forensic legal challenge and improve the security of robot systems. This work has provided insight into social robots' security and privacy implications.

References

Andrews, A., Abdelgawad, M., and Gario, A., (2016) "World Model for Testing Urban Search and Rescue (USAR) Robots using Petri Nets," Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development

Chung, H., Park, J., and Lee, S., (2017) "Digital forensic approaches for Amazon Alexa ecosystem," Digital Investigation, vol. 22, 2017, pp. S15-S25.

Chao C., and Thomaz, A., (2016) "Timed Petri nets for fluent turn-taking over multimodal interaction resources in human-robot collaboration," The International Journal of Robotics Research, vol. 35, no. 11, pp. 1330.

CPN, T, (2018) "CPN Tools – A tool for editing, simulating, and analyzing Colored Petri nets",(2018) Cpntools.org,. [Online]. Available: http://cpntools.org. [Accessed: 10- Nov- 2018].[24]

Cole, E. (2019) "What is New in Robotics." Robotiq, [Online]. Available: https://blog.robotiq.com/whats-new-in-robotics-this-week-jul-14 [Accessed: 3-June-2022]

Da Mota, F., Rocha, M., Rodrigues, J., De Albuquerque, V., and De Alexandria, A., (2018) "Localization and Navigation for Autonomous Mobile Robots Using Petri Nets in Indoor Environments," IEEE Access, vol. 6, pp. 31665-31676.

De La Mota, I., Guasch, A., Mujica Mota M., and Angel Piera, M., (2017) "Robust Modelling and Simulation". Cham: Springer, 2017, pp. 49-52.

Do, H., Pham, M., Sheng, W., Yang, D., and Liu, M., (2018) "RiSH: A robot-integrated smart home for elderly care," Robotics and Autonomous Systems, vol. 101, pp. 74-92.

Erden Z., and Araz, M., (2016) "Petri Net Modeling and Simulation of Walking Behaviour for Design of a Bioinspired Robot Dog," Proceedings of the 6th International Conference on Simulation and Modeling Methodologies, Technologies and Applications.

Fernandes, F., Guanci Y., Do, H., and Sheng, W., (2016) "Detection of privacy-sensitive situations for social robots in smart homes," 2016 IEEE International Conference on Automation Science and Engineering (CASE).

M. Lee, K. Tang, J. Forlizzi and S. Kiesler, "Understanding users' perception of privacy in human-robot interaction," Proceedings of the 6th international conference on Human-robot interaction - HRI '11, 2011.

Portugal, D., Pereira, S., and Couceiro, M., (2017) "The role of security in human-robot shared environments: A case study in ROS-based surveillance robots," 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN).

Petrosyan G., and Ter-Vardanyan, L., (2016) "Modelling of identification and secret-key generation system with Colored Petri Net," 2016 International Conference on Control, Decision and Information Technologies (CoDIT), 2016.

Poremba S., (2016) "The Internet Of Things Has A Growing Number Of Cyber Security Problems,"," Retrieved from http://www.forbes.com/sites/sungardas/2015/01/29/the-internet-of-things-has-a-growing-number-of-cyber-security-problems

Ray, P., (2016) "Internet of Robotic Things: Concept, Technologies, and Challenges," IEEE Access, vol. 4, pp. 9489-9500.

Reisig, W. (2013) "Understanding Petri Nets". Springer, 2013.

Stephenson, P. R. (2004). "A formal model for information risk analysis using colored Petri nets." in Proceedings of the Fifth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark, October 8-11, 2004, DAIMI PB - 570 / Kurt Jensen (Ed.), Oct, 2004, pp. 167–184.

Taipale, S., Vincent J., Sapio, B., Lugano, G., and Fortunati, L. (2015) " Social robots from a human perspective." Switzerland: Springer, 2015, pp. 11-15.

Tobe, F., (2017) "Robotics industry growing faster than expected - The Robot Report," The Robot Report, 2017. [Online]. Available: https://www.therobotreport.com/robotics-industry-growing-faster-than-expected/. [Accessed: 16- Sep- 2018].

Yang, Y., Dresher, S., and Kim, S., (2015) "Modeling, Simulation and Analysis of Autonomous Robot Navigation Algorithms using Fuzzy Petri Nets," International Conference on Computers And Their Applications.