A Logic-Based Model to Reduce IoT Security Risks

Luiz Otavio Botelho Lento[1,2][a], Pedro Patinho[1][b] and Salvador Abreu[1,2][c]
[1]*Department of Informatics, University of Évora, Portugal*
[2]*NOVA-LINCS, University of Évora, Portugal*

Keywords: Risk Management, IoT, Threat Mitigation, Probabilistic Logic, Fuzzy Logic.

Abstract: As the world becomes more and more dynamic and competitive, people live more and more connected, breathing a cybernetic reality in their lives. IoT systems also do not escape this reality, they are omnipresent, providing a wide range of services to their users, and increasing their quality of life, enabled by IoT devices. In parallel with this technology, information security problems are also part of this IoT evolution. A key issue with IoT environments is ensuring security across all services and devices. The diversity of threats, together with the lack of concern of most of its administrators and device designers, make the IoT network environment vulnerable. This article presents RTRMM, a logic-based security risk management model that can help protect IoT environments, with new strategies to detect, analyze and assess risks, making it possible to predict risks and aiming to manage them in real time, thereby improving the reliability and safety of the IoT environment. It makes use of a combination of probability, fuzzy logic, Markov Chains, Games Theory, and Logic Programming to specify, test and validate its functionalities.

1 INTRODUCTION

With the world becoming more dynamic and competitive, people are living increasingly connected lives, experiencing a cybernetic daily routine. This way, the vast majority of mankind is connected 24 hours a day, 7 days a week, 365 days a year, as the use of computational resources has streamlined and facilitated their daily lives. The Internet is seen as the "oxygen" for the survival of how we come about in the world, how we do business and exchange information. IoT systems also do not escape this reality, they are ubiquitous, providing a wide range of services to their users, and enhancing their quality of life by enabling smart devices, sensors and/or anything that is not generally considered a computer, to generate, exchange and use data with minimal human intervention (Sabry et al., 2019). This ability to incorporate and integrate all these smart devices meets the evolution of the Internet and wireless technology, causing a great impact on information and communication technologies (ICT), and in the Industry 4.0 (Lu, 2017).

In parallel with all this technology, information security problems are also part of this IoT evolution. Problems like the capillarity of communication, the IoT devices are vulnerable to cyberattacks, the lack of adequate security mechanisms for new threats, as well as the constant evolution of IoT technologies are elements that contribute to the growth of vulnerabilities and the increase of insecurity in the IoT environment. One of the biggest challenges is ensuring basic security properties such as confidentiality, integrity and availability of information exchanged between IoT devices is a major challenge (Ammara et al., 2018). Furthermore, the limitations in processing, storage, and even energy of each IoT device are factors that limit/prevent the implementation of more robust information security solutions (Rizvi et al., 2018). Therefore, perhaps the most appropriate way to make IoT systems more secure and reliable is to mitigate and maintain risks at an acceptable level, giving users more confidence to use its resources and services.

In this way, this article offers a new vision of managing security risks in IoT systems in real time. For this purpose, we created the RTRMM (Real Time Risk Management Model), a logic-based security risk management model that can help protect IoT environments. This model aims to manage and mitigate risks in real time, as simultaneous learning occurs. The model presents new strategies to detect, analyze

[a] https://orcid.org/0009-0002-5399-2659
[b] https://orcid.org/0000-0001-7906-6114
[c] https://orcid.org/0000-0002-1613-4631

Lento, L., Patinho, P. and Abreu, S.
A Logic-Based Model to Reduce IoT Security Risks.
DOI: 10.5220/0012455200003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 3*, pages 1197-1204
ISBN: 978-989-758-680-4; ISSN: 2184-433X

and assess these risks, using techniques that combine information security risk management strategies with Fuzzy Logic strategies, Markov Chains, Games Theory, logic and probabilistic programming. This article presents the RTRMM architecture, and the Threat Analyzer and Risk Management modules (risk analysis and assessment features).

2 IoT SECURITY - ISSUES AND CHALLENGES

Implementing security in an IoT system is also an arduous and ongoing process. IoT is one of the more rapidly and dynamically growing technology that handles protected information. The process is arduous because it consists of a wide range of steps, and continuous because its management (monitoring and control) must be carried out periodically due to the possible changes that the system may undergo, in addition to the constant and new threats that are presented in the cyber universe. To facilitate the security deployment task, understanding the organization's business processes is essential, as this understanding facilitates decision-making about which security controls will be applied, as well as the most appropriate way to implement them, in order to reduce the risks that an IoT device can suffer (Lento, 2018).

The challenges and problems in IoT systems are in part similar to most existing computational problems, differing in their specificities, such as memory limitation, processing, amongst others. IoT systems represent a diversity of interconnected technologies communicating and sharing data continuously. Therefore, security risks are created around this universe, which can cause serious problems for its users, such as those who use devices that store or inform data to patients. What can be said about IoT systems is that the confidentiality, integrity and authenticity of data exchanged, processed or even stored by devices in IoT systems is fundamental. Issues such as availability and response time are also crucial aspects for certain IoT systems, which must also be addressed. It is also worth mentioning that an IoT environment has a diversity of devices and communication technologies in its domain, which can bring discomfort to its designers and users, as it can make the search for an IoT security solution even more difficult (He et al., 2016). In (Rizvi et al., 2018), it is mentioned that the open architecture of IoT systems further increases the challenge of protecting devices, as it increases the diversity of functionalities and architectures.

Apart from the challenges already mentioned, (Malik and Singh, 2019) draws attention to yet more, such as user privacy, in which data protection is fundamental when exchanged over the Internet, ensuring confidentiality and integrity, in addition to the privacy of the user and/or devices that are handling this data. The challenge of identifying and authenticating devices and/or users can be solved by implementing cryptographic protocols and identity management strategies, as described in (Osmanoglu, 2014). However, it is worth emphasizing that the level of security of this data, for example, depends on the algorithm and the size of the key used.

3 RTRMM ARCHITECTURE

RTRMM is a new security risk management model for IoT environments, which aims to address the security challenges that IoT systems pose in real time. The logical structure of this model is based on ISO 27005 (ISO/IEC, 2022), as it is a robust approach, considering all stages of the risk management process in a clear and objective way, in addition to being an open architecture, enabling the inclusion of new functionalities. It is composed of a set of 4 (four) modules (figure 1): Threat Analyzer; Risk Management, Threat Category and Controls DB. All of these modules are integrated with each other, aiming to detect possible threats, analyse/evaluate risks and provide security measures in order to reduce the probability of incidents that may affect the functionality of IoT systems.

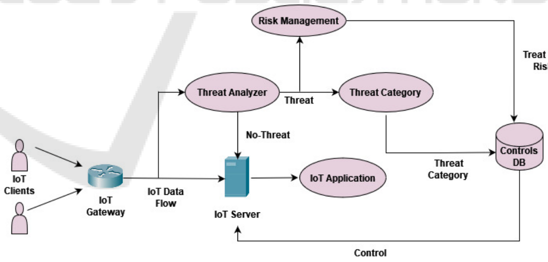


Figure 1: RTRMM Logic Model.

3.1 Threat Analyser Module

The Threat Analyzer is a module that aims to analyze an IoT data stream in order to verify if a threat exists. The detection technique applied by Threat Analyzer is based on the premise of uncertainty and probability theories, and fuzzy logic. This architecture was based on the fuzzy logic technique, as it works with a degree of uncertainty, but at the same time offers support for deciding whether a threat occurred or not (Sanjaa, 2007). The architecture of the Threat Analyzer module, shown in figure 2, was based on fuzzy logic control (FLC) (Iancu, 2012).

The choice to work with the fuzzy control architecture is due to the fact that it is applied to processes considered complex and poorly defined, especially those that can be controlled by a qualified human operator without knowledge of their evaluation dynamics. The developed architecture is composed of a set of functionalities responsible for analyzing the IoT data flow in order to check if there is a threat, presenting as a result to the probability of the data flow having a threat. How does Threat Analyzer work?

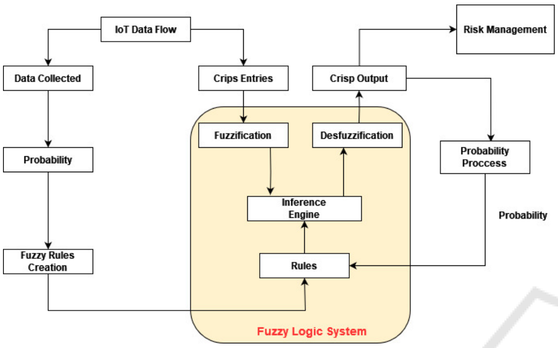


Figure 2: Threat Analyzer Architecture - Adapted from (Iancu, 2012).

When the IoT flow is analyzed for the first time, the process of inferring the probability value is carried out based on one or more fields (protocol, IP address, ports, ...) of the collected packets. The value to be inferred is determined by the IoT system administrator, based on external factors such as: history of attacks, type and volume of traffic, application context of the device in the IoT system, among others.

The crisp inputs will go through the fuzzification process to be used by the inference engine, responsible for comparing the values that underwent the fuzzification process with the created rules. The result of this process is sent to the defuzzification process, which can be forwarded to the Risk Management module or re-evaluated with the aim of better approximating the probability of a threat trail being a real threat to the system or not.

The system learns by updating the probability values of a detection, that is, all calculated probability values are inserted into the composition of new rules to be evaluated by the system. These new rules will be applied to a new set of data flows, coming from the same IoT device, seeking to bring threat detection closer to reality.

Some technologies were analyzed to determine probability values andThe design decision for Markov chain technology (Behrends, 2000). The decision is due to the fact that for each IoT data flow, a probability value is calculated in order to verify the chance of this flow being a threat. There is no concern with the previous state, but rather with the result of the current state for decision making.

3.1.1 Fuzzification and Desfuzzification Process

The fuzzification process will treat input values such as: port number, IP address, protocol (Crisps inputs), for relevance values. Four (4) groups were specified to represent the terms in the fuzzification process, enabling the calculation of the value of the fuzzy variable. The groups were established based on the methodology of a qualitative risk analysis (ISO/IEC, 2022). The choice of four (4) levels is because it makes the analysis simpler and more practical, optimizing time in the process.

1. Really - there is a high probability that it could be a threat. The degree of relevance is: $\mu S \geq 0.9$
2. Almost - there is a probability of being a threat. The degree of relevance is: $\mu S \geq 0.7 \wedge \mu S < 0.9$
3. Sometimes - there is a medium probability of being a threat. The degree of relevance is: $\mu S \geq 0.4$ $\wedge\ \mu S < 0.7$
4. Impossible - the probability of being a threat is minimal. The degree of relevance is: $\mu S < 0.4$

Inference Engine and Inference Rules - - The Threat Analyzer inference engine is responsible for applying the inference rules to the fuzzy input to generate the fuzzy output. The rules are defined together with the fuzzy inputs according to the membership functions (reflects the knowledge we have regarding the intensity with which the object belongs to the fuzzy set (S. and Izquierdo, 2018). To determine the resulting region, the inference process used the Mamdani (Mamdani and Assilian, 1975) technique, as it is intuitive, more suitable for human input, as it has a more interpretable rule base (IF-ELSE: IF TERM is Y) and a wide acceptance.

The Defuzzification process is necessary when the system is expected to return a number and not the fuzzy set created for each event. The defuzzification technique adopted was the centroid, where its calculation varies depending on the output, and can assume discrete or continuous values. In the case of Threat Analyzer, we work with discrete values, where the probability value of an anomaly determines whether or not it is a threat (Jantzen, 2007).

3.2 Risk Management Module

The Risk Management module aims to analyze and evaluate the threats detected by the Threat Analyzer directed to IoT systems, dynamically in real time, resulting in the calculated, analyzed and evaluated risk, informing the DB Controls module which evaluated risks will be treated. This module is divided into 3 (three) basic functionalities, as shown in figure 3: Calculate Impact, which aims to calculate the impact value for the IoT system in the event that an IoT device is compromised by one or more threats ; risk analysis (risk analyzer) which aims to analyze and calculate the risks existing in traffic between components of the IoT system; and evaluate risks (risk assessment), which aims to evaluate the risks determined by the risk analysis function, in order to determine those that will be treated.

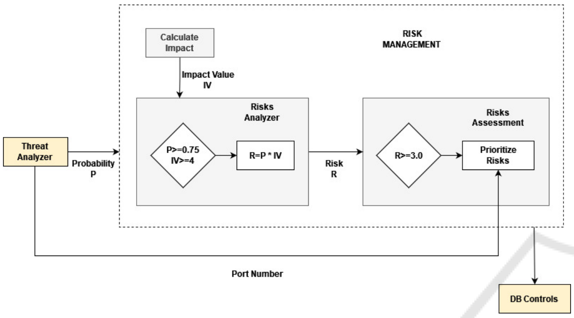


Figure 3: Risk Management Module.

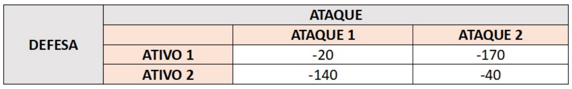
3.2.1 Impact Calculation

Impact is one of the criteria that makes it possible to know and calculate risk in the risk management process. In the RTRMM project, the parameter selected to determine the importance value was the classification level of the IoT device, that is, how important it is for maintaining the system's functionalities. The Model adopted 4 (four) impact levels, "critical", "high", "medium" and "low", in which each level receives a numerical value from 2 to 5, with 5 being "critical" and the other values being subsequent. in descending order. See, for example, how the impact can be calculated taking into account the number of IoT devices connected to the same functionality;

1. o Critical - $\exists d_i | i \geq 2$, the system must have at least 2 devices available to meet the system's needs. This means that if a device is affected in a security incident, there will be at least 2 other devices that will supply the data transfer demand in the system.
2. High - $\exists d_i | i \geq 1$, the system must have at least 1 or more devices available to meet the system's needs.
3. Medium - $\exists d_i$ — if only if $i = 1$, the system must have at least 1 device available to meet the system's needs.
4. Low – the device is not important to the system.

3.2.2 Risk Analysis

The risk analysis process (Risks Analyzer) aims to know, analyze and calculate the risks, based on all threats provided by the Threat Analyzer. To analyze and evaluate the risks, the game theory technique was adopted. The figure 4 presents a generic payoff table, with two attacking and defending players, in which the defender has two possible defense strategies - defending Asset 1 or defending Asset 2, taking into account that the attacker can attack both. There is a cost of 20 euros to defend Asset 1 and 40 euros to defend Asset 2. At the same time, it is known that there is a failure cost to defend the Asset of 150 euros and 100 euros for Asset 2.



DEFESA	ATAQUE	
	ATAQUE 1	ATAQUE 2
ATIVO 1	-20	-170
ATIVO 2	-140	-40

Figure 4: Payoff Matrix, (Cox, 2009).

With this information, you can:

1 - Know the probability of the attacker maximizing damage to the defender by attacking asset 1.

$$\frac{(40-170)}{(40-170)+(20-140)} = \frac{130}{130+120} = \frac{130}{250} = 0.52$$

2 - Know the probability of the defender minimizing the loss by defending the asset1.

$$\frac{(40-140)}{(40-140)+(20-170)} = \frac{100}{100+150} = \frac{100}{250} = 0.4$$

Now we have an assessment of the situation, the probability of damage that can be caused to an asset, 50%, and the probability of minimizing the success of an attack, which is lower, which demonstrates a possible loss if treatment is not taken adequate. In this way, the asset must be treated. Therefore, all risks that have a probability *pgeq*7 must be treated. The probability distribution for the threat is known based on the payoff matrix, in which the probability value is the value obtained by the attacker being successful in causing damage to an IoT device. The value of the impact is related to the importance the device has on the IoT system (critical, high, medium). The value of vulnerability is attributed by risk managers, as they are the ones who have full knowledge of the IoT system, its context within the organization and its environment. The risk calculation is carried out based on threat (A), vulnerability (V) and impact (I). The risk calculation formula is represented by the expression:

Risk = A x V x I

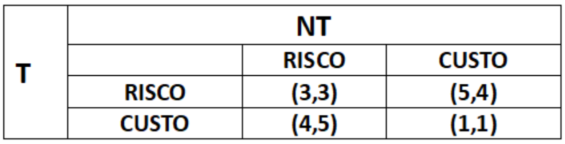
3.2.3 Risk Assessment

The risk assessment process of the risk management module has the functions of determining which risks will be treated and their order of priority. To resolve what will be discussed, Game Theory techniques were applied (Sartini et al., 2004; Pim, 2021).

The problem in the risk assessment phase is to define what will be treated (Treat - T) and what will not be treated (Do Not Treat - NT), called players, in game theory. In parallel, two strategies are established, RISK and COST, which belong to a set of pure strategies to define what to treat and what not to treat. The risk strategy to be addressed is based on the risk classification, that is, on the risk value calculated in the risk analysis phase. 4 (four) levels were specified to classify the risk. However, as a rule, the RTRMM project decided to only treat risks that have values $r \geq 3$ ("Critical" and "High" risks).

The cost strategy, which was called operational cost, is directly related to the impact that an IoT device may cause to the IoT system when it is affected by a threat. To analyze the operational cost for an IoT system when one or more devices are affected by one or more threats, factors such as impact analysis, system capillarity, and interconnected devices are taken into account in the assessment. Therefore, the cost was classified into 2 (two) categories, with the premise that only critical and high risks will be treated in real time by RTRMM.

The two functions RISK and COST are given by numeric values, represent the payoffs of N and NT. The assigned numerical values were obtained from calculating the risk and operational cost for the IoT system calculated based on the importance of the IoT device to the system. The functions are mapped into the payoffs matrix (figure 5), in which each cell of this matrix represents the payoffs of treating (T) and not treating (NT).



		NT	
T		RISCO	CUSTO
	RISCO	(3,3)	(5,4)
	CUSTO	(4,5)	(1,1)

Figure 5: Payoff Matrix.

Solution
The solution is to predict the outcome of the game. When analyzing risk assessment from the perspective of treatment, it can be carried out both depending on the risk and the operational cost. In both situations, the strategy of dominance was applied, instead of balance, as the values are dominant compared to others. Therefore, applying the dominance strategy, the result for the treatment prioritizes the risk and then the cost. **(RISK,COST)**

3.3 Threat Category Module

The Threat Category module aims to classify threats into categories so that they can be easily searched when selecting the best security mechanism. The strategy adopted was to link categories of similar threats, aiming to reduce the number of security mechanisms to be applied in the IoT system. Despite being a somewhat generic strategy, it is useful due to the need for RTRMM to manage risks in real time. The similarity strategy groups a set of threats into the same category, such as DoS (Deny of Service), invasion, data privacy, among others. Therefore, to categorize a threat, a set of parameters were established for each of these categories.

Upon receiving the threat parameters from the Threat Analyzer, the Threat Category analyzes the parameters, pre-established by the risk manager, and classifies it into a category (figure 6). However, there is still a question, how to analyze these parameters and classify the threat.

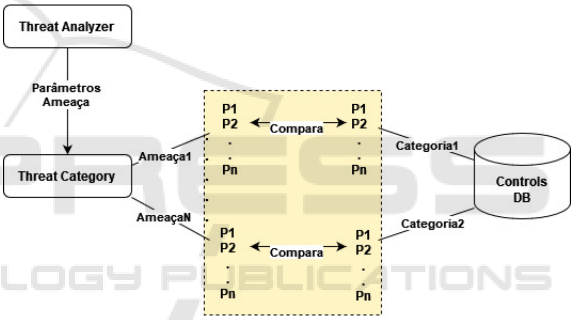


Figure 6: Threat Category Dynamics.

Therefore, a solution was adopted that makes use of a discrete-time stochastic process, where an acyclic graph G = (V,E) has a finite, non-empty set of vertices V and a set of edges A. Each vertex of this graph represents a parameter that makes up the threat detected by the Threat Analyzer, and the path between two or more sequential vertices defines a threat category.
Definition: for each pair of sequential vertices (u,v), in an acyclic graph G = (V,E), where u and v are IoT traffic parameters, form an edge of a path that represents the characteristic threat.

There is an edge only if: $e_v = \{e_1, e_2, ..., e_n\}$, where e_v is a set of security events.

If $(e_i \wedge e_n)$ ε $e_v \rightarrow \forall e_i, e_{i+1} \ \exists \ e_d$, where e_d is an edge composed of (e_i, e_{e+1})

Definition. There is a path in an acyclic graph $G = (V, E)$ if only if there is a set of edges (u,v).

$e_d = \{e_{d1}, e_{d2}, ..., e_{dn}\}$, where e_d is a set of edges e_{di}.

If $(e_{di} \wedge e_{di..n})$ ε e_d → ∃ PATH | PATH = $\{e_{d1}, e_{d2}, ..., e_{dn}\}$

The construction of the graph is based on the information collected in the IoT data flow, that is, on the detected threat parameters. For each IoT traffic threat parameter, a vertex of the graph G = (V,E) corresponds. The edges will be created by interconnecting these vertices forming several paths. Each path will correspond to a threat category. All threat categories will be defined in advance, stored in a database or defined in real time (zero-day attacks may create a new category).

For example, suppose an acyclic graph, like figure 7, with $n = |N|$ and $m = |E|$, where n is the number of vertices and m the number of edges.

Begin $t\varepsilon T = \{1,2,3,4,5,...\}$.

$$X_t = \begin{cases} Threat(Cat), & \text{if s and t in V(G) there is a path} \\ No-threat, & \text{not exist a path} \end{cases}$$

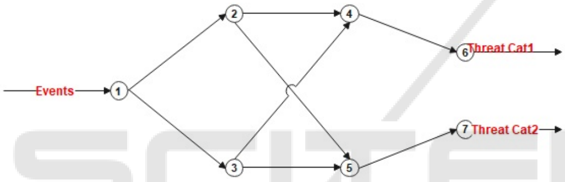


Figure 7: Threat Category.

Therefore, the strategy used in the Threat Category is a way to improve the performance of risk management in RTRMM, as it makes it easier to determine which security mechanism to use.

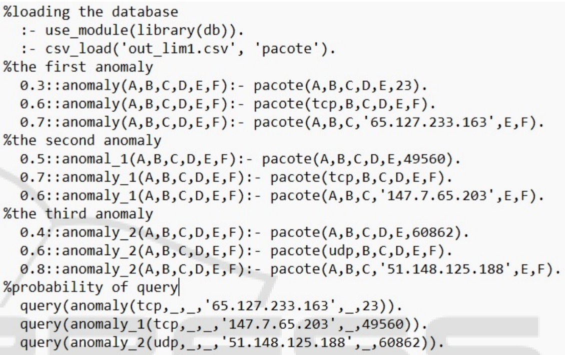
4 RTRMM PROTOTYPE

The section aims to present how the RTRMM was implemented, based on the logical model shown in figure 3. Not all modules were implemented and tested, as the project is still in the implementation and validation phase. The functionalities of the implemented modules were carried out based on logic programming and probabilistic logics, resorting to Prolog and Problog (de Raedt, 2007).

4.1 Threat Analyzer Prototype Implementation

The Threat Analyzer module was implemented in Problog (de Raedt, 2007) and used IoT-23, a network traffic dataset of Internet of Things (IoT) devices, generated by "Stratosphere Laboratory, AIC group, FEL, CTU University, Czech Republic" (Laboratory, 2020), that aims to offer a large dataset, infected by malware from real IoT devices.

The developed code made use of 6 fields (IoT traffic identification parameters) from the IoT-23 database, simulating Crisp entries, as in the fuzification process: protocol; Date of the event; source and destination IP address; and source port and destination port. Initially. 3 anomalies were specified, based on the fuzzification rules strategy, and on the malware parameters presented in IoT-23. For each of the anomalies, three (3) IoT traffic identification parameters were selected for their composition: protocol, destination IP address and destination port (figure 8).



```
%loading the database
  :- use_module(library(db)).
  :- csv_load('out_lim1.csv', 'pacote').
%the first anomaly
  0.3::anomaly(A,B,C,D,E,F):- pacote(A,B,C,D,E,23).
  0.6::anomaly(A,B,C,D,E,F):- pacote(tcp,B,C,D,E,F).
  0.7::anomaly(A,B,C,D,E,F):- pacote(A,B,C,'65.127.233.163',E,F).
%the second anomaly
  0.5::anomal_1(A,B,C,D,E,F):- pacote(A,B,C,D,E,49560).
  0.7::anomaly_1(A,B,C,D,E,F):- pacote(tcp,B,C,D,E,F).
  0.6::anomaly_1(A,B,C,D,E,F):- pacote(A,B,C,'147.7.65.203',E,F).
%the third anomaly
  0.4::anomaly_2(A,B,C,D,E,F):- pacote(A,B,C,D,E,60862).
  0.6::anomaly_2(A,B,C,D,E,F):- pacote(udp,B,C,D,E,F).
  0.8::anomaly_2(A,B,C,D,E,F):- pacote(A,B,C,'51.148.125.188',E,F).
%probability of query
  query(anomaly(tcp,_,_,'65.127.233.163',_,23)).
  query(anomaly_1(tcp,_,_,'147.7.65.203',_,49560)).
  query(anomaly_2(udp,_,_,'51.148.125.188',_,60862)).
```

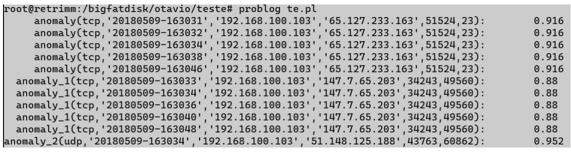
Figure 8: Codificação do Threat Analyzer.

The strategy of inferring the probability value for each parameter in the composition of each anomaly was initially based on the analysis/evaluation of the events presented in IoT-23. The strategy analyzed the number of occurrences of each parameter in the IoT-23 database in the N/M ratio. The N value corresponds to the number of occurrences of a parameter in the IoT-23 database, and M to the total number of data flows in the IoT-23 database. These values were inserted in one of the nodes of the Markov network, which allows the probability value for a packet to be a threat or not.

4.2 Risk Management Implementation

The implementation of the Risk Management module was carried out in Problog, using as input a database with anomalies and their respective probabilities detected by the Threat Analyzer. This database contained eleven (11) entries (anomalies and probabilities), where five (5) were of one type of anomaly, another five (5) of another type and, finally, a single occurrence of the third type of anomaly, for each type of anomaly, the system calculated the probability of these being a threat or not, as can be seen in figure 7.

The implementation of the Risk Management module was carried out in Prolog, having as input a database with the anomalies and their respective probabilities detected by the Threat Analyzer. This database contained eleven (11) entries with the anomalies and their respective probabilities, as seen in figure 9.



```
root@retrimm:/bigfatdisk/otavio/teste# problog te.pl
     anomaly(tcp,'20180509-163031','192.168.100.103','65.127.233.163',51524,23):        0.916
     anomaly(tcp,'20180509-163032','192.168.100.103','65.127.233.163',51524,23):        0.916
     anomaly(tcp,'20180509-163034','192.168.100.103','65.127.233.163',51524,23):        0.916
     anomaly(tcp,'20180509-163038','192.168.100.103','65.127.233.163',51524,23):        0.916
     anomaly(tcp,'20180509-163046','192.168.100.103','65.127.233.163',51524,23):        0.916
  anomaly_1(tcp,'20180509-163033','192.168.100.103','147.7.65.203',34243,49560):        0.88
  anomaly_1(tcp,'20180509-163034','192.168.100.103','147.7.65.203',34243,49560):        0.88
  anomaly_1(tcp,'20180509-163036','192.168.100.103','147.7.65.203',34243,49560):        0.88
  anomaly_1(tcp,'20180509-163040','192.168.100.103','147.7.65.203',34243,49560):        0.88
  anomaly_1(tcp,'20180509-163048','192.168.100.103','147.7.65.203',34243,49560):        0.88
anomaly_2(udp,'20180509-163034','192.168.100.103','51.148.125.188',43763,60862):        0.952
```

Figure 9: Result Anomalies Detected.

The risk management module created a database in Prolog, consisting of the type of anomaly and its probability, in addition to the threat levels proposed by the RTRMM (Really, Almost, Sometimes, Impossible). The following results were obtained, and whether they should be adjusted:

- Anomaly - this anomaly scored 0.916 as a result and is considered “Really”, a high possibility of being a threat. DEAL WITH THE THREAT - PRIORITY 2;
- Anomaly_1 - this anomaly scored 0.88 as a result, being considered “Almost” there is a possibility of being a threat. DEAL WITH THE THREAT - PRIORITY 3;
- Anomaly_2 - This anomaly scored 0.952 as a result and is considered “Really”, a high possibility of being a threat. TREAT - PRIORITY 1.

Considering that these anomalies are a threat, the next step of the RTRMM Risk Management is to evaluate which of these threats will be treated and their degree of priority. RTRMM assumed that for any threat that has the degree of relevance (probability) $\mu S \geq 0.7$ it should be treated. The reason for this decision is to reduce the rate of false positives/negatives, improving the probability of treating detected threats. The initial rate of false positives/negatives, without applying the value of the relevance degree $\mu S \geq 0.7$ was around 20% occurrences, considered very high. With the application of the relevance value, this rate was reduced by approximately 60%, excluding threats with little probability of being true, improving the reliability of threat validation.

As for the priority of treatment of threats, RTRMM initially adopted that the ordering of probabilities calculated by the system will determine the priority of treatment. This means that the treatment sequence will be: Anomaly_2; Anamoly; and Anomaly_1.

4.3 Performance

To assess whether RTRMM is viable, a brief performance analysis was carried out with two different situations, as shown in figure 10.

- Situation 1 - each anomaly had 3 parameters (protocol, destination address and destination port), and the system used 3, 5 and 7 anomalies for time analysis.
- Situation 2 - each anomaly had 5 parameters (protocol, source and destination address and source and destination port), and the system used 3, 5 and 7 anomalies for time analysis.

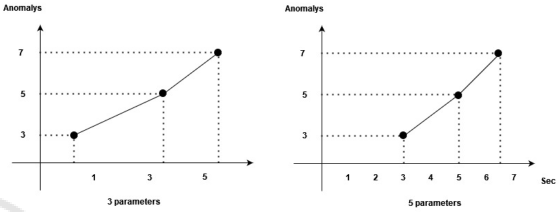


Figure 10: Run-time Comparison.

What can be observed is that the system presented a performance behavior considered adequate, with its time growth rate practically proportional (i.e. linear) to the growth of the number of anomalies, both with the use of 3 or 5 parameters in the composition of an anomaly.

5 CONCLUSION

This article presented a new security risk management model based on probabilistic logic for IoT environments. The model is able to recognize threats and the probability of them happening, in addition to analyzing and evaluating the risks, as well as treating them. The Threat Analyzer and Risk Management modules were implemented, showing how RTRMM will be able to perform its functionalities.

Not all modules and functionalities have been completely implemented and tested, such as the DB Controls module, the learning strategy in the Threat Analyzer and the choice of security measure that is supposed to be applied. Currently, the process is still in its testing phase regarding its efficiency using Bayesian networks and it will still be subjected to final analysis. Regarding the application of security measures, the interruption of communication with the IoT device that is suffering the threat/attack, is being applied. However, all of these elements are being worked on in order to be added to the functionality of RTRMM. The next step of this project is to benchmark

RTRMM in IoT Healthcare environments and extensively compare it to competing systems.

ACKNOWLEDGEMENTS

This work was partially supported by Funda̧ão para a Ciência e a Tecnologia, Portugal, through the NOVA LINCS research center (UIDB/04516/2020).

REFERENCES

Ammara, M., Russellob, G., and Crispo, B. (2018). Internet of things: A survey on the security of iot frameworks. In *Journal of Information Security and Applications*, pages 8–27. Elsevier.

Behrends, E. (2000). Introduction to markov chains with special emphasis on rapid mixing. In *Advanced Lectures in Mathematics*. Spinger.

Cox, L. (2009). Game theory and risk analysis. In *Risk Analysis, Vol.29*, pages 1062–1068. Society for Risk Analysis.

de Raedt, L. (2007). A probabilistic prolog and its application in link discovery. In *IJCAI-07*, pages 2468–2472. IJCAI.

He, H., Maple, C., Watson, T., Tiwari, A., Mehnen, J., Jin, Y., and Gabrys, B. (2016). The security challenges in the iot enabled cyber-physical systems and opportunities for evolutionary computing and other computational intelligence. In *IEEE Congress on Evolutionary Computation (CEC)*, pages 1015–1021.

Iancu, I. (2012). A mamdani type fuzzy logic controller. In *Fuzzy Logic – Controls, Concepts, Theories and Applications*, pages 325–349. Intechopen.

ISO/IEC (2022). Iso/iec 27005:2022 information security, cybersecurity and privacy protection — guidance on managing information security risks. In *ISO/IEC*. ISO/IEC.

Jantzen, J. (2007). Tutorial on fuzzy logic. In *Tech. report no 98-E 868*. Technical University of Denmark, Oersted-DTU.

Laboratory, S. (2020). Aposemat iot-23. In *https://www.stratosphereips.org/datasets-iot23*. Stratosphere Laboratory.

Lento, L. O. (2018). *Segurança da Informação*. UNISUL, Brasil, 1st edition.

Lu, Y. (2017). Industry 4.0:a survey on technologies, applications and open research issues. In *Journal of Industrial Information Integration*, pages 1–10. Elsevier.

Malik, V. and Singh, S. (2019). Security risk management in iot environment. In *Journal of Discrete Mathematical Sciences and Cryptography vol. 22, no 4*, pages 697–709. Taru Publications.

Mamdani, E. and Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. In *Int. J. Man-mach. Studies*, pages 1–13. Elsevier.

Osmanoglu, E. (2014). Identity and access management - business performance through connected intelligence. In *Syngress*. Elsevier.

Pim, B. (2021). Uma breve introdução à teoria dos jogos. In *Revista Eletrônica Paulista de Matemática, vol. 1*, pages 69–80. C.Q.D.

Rizvi, S., Pfeffer III, J., Kurtz, A., and Rizvi, M. (2018). Securing the internet of things (iot): A security taxonomy for iot. In *17th IEEE Int. Conf. On Trust, Security And Privacy In Computing And Communications*, pages 163–168. IEEE.

S., I. and Izquierdo, L. R. (2018). Mamdani fuzzy systems for modelling and simulation: A critical assessment. In *Journal of Artificial Societies and Social Simulation*. JASSS.

Sabry, S. S., Qarabash, N. A., and Obaid, H. S. (2019). The road to the internet of things: a survey. In *9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)*, pages 290–296. IEEE.

Sanjaa, B. (2007). Fuzzy and probability. In *IEEE Xplore*, pages 141–143. IEEE.

Sartini, B., Garbugio, G., Bortolossi, H., Santos, P., and Barreto, L. (2004). Uma introdução a teoria dos jogos. In *II Bienal da SBM*, pages 1–62. UFB.